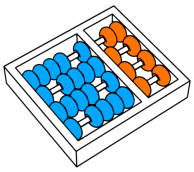
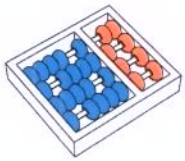


Fabio Augusto Faria

"A Framework for Pattern Classifier Selection and Fusion"

"*Um Arcabouço para Seleção e Fusão de Classificadores de Padrão*"

CAMPINAS
2014



University of Campinas
Institute of Computing

Universidade Estadual de Campinas
Instituto de Computação

Fabio Augusto Faria

"A Framework for Pattern Classifier Selection and Fusion"

Supervisor: *Orientador(a):* Prof. Dr. Ricardo da Silva Torres

Co-Supervisor: *Co-orientador(a):* Prof. Dr. Anderson Rocha

"Um Arcabouço para Seleção e Fusão de Classificadores de Padrão"

DEGREE Thesis presented to the Post Graduate Program of the Institute of Computing of the University of Campinas to obtain a TITLE degree in Computer Science.

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Computação da Universidade Estadual de Campinas para obtenção do título de Doutor em Ciência da Computação.

THIS VOLUME CORRESPONDS TO THE FINAL VERSION OF THE THESIS DEFENDED BY FABIO AUGUSTO FARIA, UNDER THE SUPERVISION OF PROF. DR. RICARDO DA SILVA TORRES.

ESTE EXEMPLAR CORRESPONDE À VERSÃO FINAL DA TESE DEFENDIDA POR FABIO AUGUSTO FARIA, SOB ORIENTAÇÃO DE PROF. DR. RICARDO DA SILVA TORRES.

Ricardo Torres

Supervisor's signature / *Assinatura do Orientador(a)*

CAMPINAS
2014

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca do Instituto de Matemática, Estatística e Computação Científica
Maria Fabiana Bezerra Muller - CRB 8/6162

Faria, Fabio Augusto, 1983-
F225f A framework for pattern classifier selection and fusion / Fabio Augusto Faria. – Campinas, SP : [s.n.], 2014.

Orientador: Ricardo da Silva Torres.
Coorientador: Anderson Rocha.
Tese (doutorado) – Universidade Estadual de Campinas, Instituto de Computação.

1. Categorização de imagens. 2. Reconhecimento de padrões. 3. Processamento de imagens. 4. Sensoriamento remoto. I. Torres, Ricardo da Silva,1977-. II. Rocha, Anderson de Rezende,1980-. III. Universidade Estadual de Campinas. Instituto de Computação. IV. Título.

Informações para Biblioteca Digital

Título em outro idioma: Um arcabouço para seleção e fusão de classificadores de padrão
Palavras-chave em inglês:
Image categorization
Pattern recognition
Image processing
Remote sensing
Área de concentração: Ciência da Computação
Titulação: Doutor em Ciência da Computação
Banca examinadora:
Ricardo da Silva Torres [Orientador]
William Robson Schwartz
Renata de Matos Galante
Hélio Pedrini
Letícia Rittner
Data de defesa: 07-03-2014
Programa de Pós-Graduação: Ciência da Computação

TERMO DE APROVAÇÃO

Defesa de Tese de Doutorado em Ciência da Computação, apresentada pelo Doutorando Fabio Augusto Faria, aprovada em 07 de março de 2014 pela Banca examinadora composta pelos professores doutores:

Prof. Dr. William Robson Schwartz
DCC / UFMG

Profª. Drª. Renata de Matos Galante
Inf / UFRGS

Prof. Dr. Hélio Pedrini
IC / UNICAMP

Profª. Drª. Letícia Rittner
FEEC / UNICAMP

Prof. Dr. Ricardo da Silva Torres
IC / UNICAMP

Institute of Computing /*Instituto de Computação*
University of Campinas /*Universidade Estadual de Campinas*

A Framework for Pattern Classifier Selection and Fusion

Fabio Augusto Faria[1]

March 07, 2014

Examiner Board/*Banca Examinadora*:

- Prof. Dr. Ricardo da Silva Torres (Supervisor)
- Prof. Dr. Hélio Pedrini
  Institute of Computing - UNICAMP
- Prof. Dr. Leticia Rittner
  School of Electrical and Computer Engineering - UNICAMP
- Prof. Dr. Renata de Matos Galante
  Institute of Informatics - UFRGS
- Prof. Dr. William Robson Schwartz
  Department of Computer Science - UFMG
- Prof. Dr. Fábio Augusto Cappabianco
  Institute of Science and Technology - UNIFESP (Substitute)
- Prof. Dr. Leandro Aparecido Villas &
  Prof. Dr. Marco Antonio Garcia de Carvalho
  Institute of Computing & Faculty of Technology - UNICAMP (Substitute)

[1]Financial support: São Paulo Research Foundation – FAPESP (grant 2010/14910-0) 2010-2014 and CAPES (grant 1260-12-0).

Abstract

The frequent growth of visual data, either by countless available monitoring video cameras or the popularization of mobile devices that allow each person to create, edit, and share their own images and videos have contributed enormously to the so-called "big-data revolution". This shear amount of visual data gives rise to a Pandora box of new visual classification problems never imagined before. Image and video classification tasks have been inserted in different and complex applications and the use of machine learning-based solutions has become the most popular approach to several applications. Notwithstanding, there is no silver bullet that solves all the problems, i.e., it is not possible to characterize all images of different domains with the same description method nor is it possible to use the same learning method to achieve good results in any kind of application. In this thesis, we aim at proposing a framework for classifier selection and fusion. Our method seeks to combine image characterization and learning methods by means of a meta-learning approach responsible for assessing which methods contribute more towards the solution of a given problem. The framework uses three different strategies of classifier selection which pinpoints the less correlated, yet effective, classifiers through a series of diversity measure analysis. The experiments show that the proposed approaches yield comparable results to well-known algorithms from the literature on many different applications but using less learning and description methods as well as not incurring in the curse of dimensionality and normalization problems common to some fusion techniques. Furthermore, our approach is able to achieve effective classification results using very reduced training sets.

Resumo

O crescente aumento de dados visuais, seja pelo uso de inúmeras câmeras de vídeo monitoramento disponíveis ou pela popularização de dispositivos móveis que permitem pessoas criar, editar e compartilhar suas próprias imagens/vídeos, tem contribuído enormemente para a chamada "*big data revolution.*"Esta grande quantidade de dados visuais dá origem a uma caixa de Pandora de novos problemas de classificação visuais nunca antes imaginados. Tarefas de classificação de imagens e vídeos foram inseridos em diferentes e complexas aplicações e o uso de soluções baseadas em aprendizagem de máquina tornou-se mais popular para diversas aplicações. Entretanto, por outro lado, não existe uma "bala de prata"que resolva todos os problemas, ou seja, não é possível caracterizar todas as imagens de diferentes domínios com o mesmo método de descrição e nem utilizar o mesmo método de aprendizagem para alcançar bons resultados em qualquer tipo de aplicação. Nesta tese, propomos um arcabouço para seleção e fusão de classificadores. Nosso arcabouço busca combinar métodos de caracterização de imagem e aprendizagem por meio de uma abordagem de meta-aprendizagem que avalia quais deles contribuem melhor para solução de um determinado problema. O arcabouço utiliza três diferentes estratégias de seleção de classificadores para apontar o menos correlacionados e eficazes, por meio de análises de medidas de diversidade. Os experimentos mostram que as abordagens propostas produzem resultados comparáveis aos famosos métodos da literatura para diferentes aplicações, utilizando menos classificadores e não sofrendo com problemas que afetam outras técnicas como a maldição da dimensionalidade e normalização. Além disso, a nossa abordagem é capaz de alcançar resultados eficazes de classificação usando conjuntos de treinamento muito reduzidos.

Dedication
To my family and friends.

Dedicatória
Para minha família e amigos.

Agradecimentos/Acknowledgements

Eu gostaria de agradecer ao meu pai pelo suporte, pela confiança depositada em todos os momentos, por ser essa pessoa tão importante em minha vida e que sou grato todos os dias (um grande amigo); aos meus irmãos Sérgio e Vitor pelo apoio e amizade verdadeira que temos; à minha princesa e quase esposa Ariadne pelo carinho, paciência e cumplicidade em todos os momentos; aos meus colegas de Laboratório (Recod) do Instituto de Computação da Unicamp pelo ambiente descontraído e pelas colaborações que tivemos durante o período; a todos os amigos por existirem, pois sem eles eu nada seria.

Agradeço ao orientador Prof. Dr. Ricardo da Silva Torres pela paciência, compreensão, ensinamentos e principalmente por se lembrar que pessoas são diferentes e por isso, necessitam de tratamentos diferentes. Esta capacidade de gerenciar pessoas é fundamental para seu sucesso e não tenho dúvidas que só aumentará no futuro. Deixo aqui o meu sincero MUITO OBRIGADO!

Agradeço ao co-orientador Prof. Dr. Anderson Rocha pelos ensinamentos e por todas as discussões "acaloradas" que tivemos durante o período. Tenho certeza que cada reunião de pesquisa me fez crescer ainda mais como pessoa, professor e pesquisador; à Prof. Dra. Claudia Medeiros pelo apoio, sabedoria e pela oportunidade de fazer parte de seu grupo de pesquisas (LIS); a todos colaboradores de diferentes universidades do país e fora dele, pela parceria nos trabalhos realizados e nos que ainda virão.

Agradeço aos funcionários do Instituto de Computação, principalmente Daniel, Flavio, Wilson e Dona Rosa que foram sempre muito dedicados aos alunos de pós-graduação.

Agradeço a Fundação de Amparo à Pesquisa do Estado de São Paulo - FAPESP (processo 2010/14910-0), Conselho Nacional de Desenvolvimento Científico e Tecnológico - CNPq e Coordenação de Aperfeiçoamento de Pessoal de Nivel Superior - CAPES (proceso 1260-12-0) pelo apoio financeiro.

I would like to thank Prof. Dr. Sudeep Sarkar for all the wonderful things he did for me while I was at University of South Florida (USF) and by treatment that I have received there. I have no words to describe how I am grateful to him for accepting me in his research group. The experience I gained will be extremely important to my personal and professional life.

Epigraph

"The more I see of the World, the more I find out the extent of my ignorance."

Just paraphrasing Socrates...

Epígrafe

"Quanto mais conheço o Mundo, mais vejo o tamanho da minha ignorância."

Apenas parafraseando Socrates...

Contents

Abstract **ix**

Resumo **xi**

Dedication **xiii**

Agradecimentos/Acknowledgements **xv**

Epigraph **xvii**

1 Introduction **1**
1.1 Motivation 1
1.2 Objective and Contributions 3
1.3 Thesis Organization 4

2 Related Work and Background **5**
2.1 Image Categorization and Information Fusion 5
2.2 Learning Methods 8
2.3 Ensemble Techniques 12
2.4 Diversity Measures 14
2.5 Image Descriptors 16
2.6 Applications 19
2.6.1 Object Recognition 20
2.6.2 Produce Recognition 20
2.6.3 Remote Sensing Image Recognition 21
2.6.4 Natural Scene Classification 23

3 Classifier Selection and Fusion Framework **25**
3.1 Formalization of Selection and Fusion Framework 25
3.2 Classifier Selection Approaches based on Diversity Measures 26

3.2.1 Selection based on Consensus . . . 26
3.2.2 Selection based on Kendall Correlation . . . 29
3.2.3 Selection based on Rank Aggregation . . . 32
3.3 Meta-learning Approach . . . 36
3.4 Cost-effectiveness Analysis . . . 38

4 Experimental Methodology **39**
4.1 Datasets . . . 39
4.2 Image Descriptors . . . 39
4.3 Learning Methods and Baselines . . . 41
4.4 Evaluation Measures . . . 43
4.5 Validation Protocol . . . 43

5 Experimental Results **45**
5.1 Framework for Classifier Fusion . . . 45
5.1.1 Effectiveness Analysis . . . 45
5.1.2 Tests of Statistical Significance . . . 47
5.1.3 Training Set Size Impact . . . 48
5.2 Correlation Analysis between Diversity Measures . . . 51
5.3 Classifier Selection Approaches . . . 51
5.3.1 Behavioral Analysis . . . 52
5.3.2 Fine-grained Analysis of the Selection Process . . . 54
5.3.3 Effectiveness Analysis considering Early, Late, and Hybrid Fusion . . . 57
5.3.4 Statistical Test of Significance (t-test) . . . 60

6 Conclusions **63**
6.1 Contributions . . . 63
6.2 Future Work . . . 65
6.3 List of Publications . . . 67

Bibliography **69**

List of Tables

2.1 Relationship matrix $\mathcal{M}$ between two classifiers c_i and c_j. 14

4.1 Five datasets used in this thesis. 39
4.2 Four datasets and seven image descriptors used in our experiments. 40
4.3 Five datasets and six image descriptors used in these experiments. 40
4.4 Two datasets and six image descriptors used in these experiments. 40
4.5 Learning methods used in our experiments with their default parameters. . 42
4.6 Simple learning methods combined by our meta-learning technique for each experiment. 42
4.7 Confusion Matrix. TP, TN, FP, and FN stand for true positive, true negative, false positive, and false negative, respectively. 43

5.1 Classification effectiveness of the proposed framework and baselines, with their respective standard deviations. 46
5.2 Significance tests for FSVM-KERNEL-$|C^*|$, where $|C^*|$ is the number of classifiers used by SVM in each test. 48
5.3 Kendall tau between five diversity measures using four different datasets. In blue, highest correlation coefficients and in red, lowest correlation coefficients. 52
5.4 Kappa indices computed for all diversity measures using the 5-fold cross-validation protocol for different number of classifiers ($|\mathcal{C}^*|$) in the Coffee dataset. Similar effectiveness performances were observed for other evaluation measures (accuracy and Tau index). 55
5.5 Kappa indices computed for all diversity measures using the 5-fold cross-validation protocol for different number of classifiers ($|\mathcal{C}^*|$) in the Urban dataset. Similar effectiveness performances were observed for other evaluation measures (accuracy and Tau index). 56
5.6 Classification effectiveness of different classifier fusion techniques, with their respective standard deviations. In red are rank aggregation approach, which achieves better results. 61
5.7 Significance tests comparing our approach with all baselines. 62

List of Figures

2.1 Three different fusion schemes for remote sensing image classification. . . . 8

2.2 A decision tree created for binary classification task (blue and red classes). Notice that a decision tree might be composed of different attributes (e.g., binary, ordinal, and continuous). . . . 9

2.3 (a) Samples of two classes (square and circle) in the features space. (b) Given a new object, its $k = 5$ nearest neighbors will define its class. In this case, the green point is labeled by the blue class. . . . 10

2.4 The SVM classifier builds a maximum margin decision hyperplane to separate two classes (squares and circles). Filled squares and circle are support vectors. . . . 11

2.5 Initially, weight each training example equally. (a) Weak Classifier 1 is applied; (b) Weights are increased; (c) Weak Classifier 2 is applied; (d) Weights are increased; (e) Weak Classifier 3 is applied; (f) Final classifier is a linear combination of all weak classifiers. . . . 13

2.6 Training and classification steps using a Bagging approach. . . . 14

2.7 Example of ranked lists for two different diversity measures used in our work. 16

2.8 *Accordion class from the Caltech101 dataset.* . . . 20

2.9 *Four different images from the Supermarket Produce dataset.* . . . 21

2.10 Coffee data with (a) original RSI and (b) ground truth that indicates the regions that correspond to coffee crop. Urban data with (c) original RSI and (d) ground truth that indicates the regions that correspond to urban areas. In (b) and (d), white and black regions are coffee/non-coffee crops and uban/non-urban area, respectively. . . . 22

2.11 Five different images from the Freefoto dataset. . . . 23

3.1 Proposed framework for classifier fusion and selection. (a) Given a classification problem with training examples, we train different classifiers. (b) By means of diversity measures, we select the most discriminatives classifiers. (c) Combine classifiers in a meta level using any other classifier. Notice that, in this particular example, both the SVM and Majority Voting (MV) techniques can be used for classifier fusion. 27
3.2 The five steps for classifier selection are: (a) Computation of diversity measures from the validation matrix M_V; (b) Ranking of pairs of classifiers by their diversity measure scores; (c) Selection of the top $t = 100$ ranked pairs of classifiers; (d) Computation of a histogram $\mathcal{H}$ that counts the number of occurrences of each classifier; (e) Selection the most appropriate classifiers $|\mathcal{C}^*|$ based on their occurrence in $\mathcal{H}$ and on a defined threshold $\mathcal{T}$. 28
3.3 Example of three computed ranked lists ($\mathcal{R}_{d_1}$, $\mathcal{R}_{d_2}$, and $\mathcal{R}_{d_3}$) and *Kendall* scores between them. Both ranked lists ($\mathcal{R}$) and *Kendall* are computed by using the validation matrix M_V (see Section 3.2). 31
3.4 The six steps for new classifier selection are: (a) Compute diversity measures from the validation matrix M_V; (b) Sort $\mathcal{R}$ lists by diversity measure scores; (c) Compute *Kendall* correlation coefficients among all ranked lists of classifiers $\mathcal{R}$; (d) Select $\mathcal{R}_{d_{H_1}}$ and $\mathcal{R}_{d_{H_2}}$ or $\mathcal{R}_{d_{L_1}}$ and $\mathcal{R}_{d_{L_2}}$ ranked lists to be used in the next step; (e) $\mathcal{R}^t$ lists with top $t = 100$; (f) Compute a histogram $\mathcal{H}$ that counts the number of occurrences of each classifier; (g) Select the most appropriate classifiers $|\mathcal{C}^*|$ based on their occurrence in $\mathcal{H}$ and a defined threshold $\mathcal{T}$. 32
3.5 The six steps of the new classifier selection are: (a) Compute diversity measures from the validation matrix M_V; (b) Sort $\mathcal{R}$ lists according to scores of diversity measures; (c) Compute rank aggregation using all ranked lists of classifiers ($\mathcal{R}$) and evaluation measures (E); (d) Create a single list $\mathcal{R}_c^t$, which list has the top $t = 100$; (e) Compute a histogram $\mathcal{H}$ that counts the number of occurrences of each classifier; (f) Select the most appropriate classifiers $|\mathcal{C}^*|$ that satisfy a defined threshold $\mathcal{T}$. 35
3.6 Rank aggregation approach for combining ranked lists of classifiers defined by both diversity and evaluation measures. 37
3.7 Example of matrix M_V. 37

5.1 The best results using different learning methods in the Caltech, Coffee, Freefoto, and Fruits datasets. 47
5.2 Confidence intervals for all techniques compared in Table 5.1. 49
5.3 Accuracy scores of all classifiers using training sets with different sizes. . . 50

5.4 Histograms related to the occurrence of classifiers in the selection process for the Coffee dataset. 57
5.5 Histograms related to the occurrence of classifiers in the selection process for the Urban dataset. 58
5.6 Average accuracy performances of all non-complex classifiers used in our experiments for the Coffee dataset. The blue line defines the employed threshold ($\mathcal{T}$) values, as described in Section 3.2. 58
5.7 Average accuracy performances of all non-complex classifiers used in our experiments for the Urban dataset. The blue line defines the employed threshold ($\mathcal{T}$) values, as described in Section 3.2. 59
5.8 Confidence intervals for all late fusion techniques compared in Table 5.6. . 62

6.1 Steps/modules of the proposed framework that could be extended/improved in future work. 66

Chapter 1

Introduction

1.1 Motivation

The ever growing presence of sensors in our daily lives led us to the so called big-data revolution and within this shear amount of data, visual data is of particular interest. Citing a recent The New York Times article [11], "The philosophy of data," surely the rising philosophy of the day is *data-ism* in which everyone wants to take advantage of data as it is the holy grail of contemporaneity. But, amidst such a massive amount of data, the question is how to process such data to actually come out with useful conclusions?

Visual data is of particular interest in this revolution. The explosion of visual data makes us face many new challenges unthinkable two decades ago. Image and video classification tasks have been inserted in different and complex applications (e.g., data categorization in search, biometric recognition, and document indexing through visual content, object recognition, etc.) and the use of machine learning-based solutions has become the most popular approach to several applications. However, there is no silver bullet that solves all the problems which means that it is not possible to characterize all images of different domains with the same description method nor is it possible to use the same learning method to achieve good results in any kind of application ("No Free Lunch" theorems) [111]. Depending on the extraction and learning methods used might create different classifiers that provide complementary information.

One common strategy that has been used to take advantage of these complementary information and improve classification results is the Multiple Classifier Systems (MCS). In MCS, the diversity of classifiers is an essential factor to reach better effectiveness results [61, 95]. Diversity measures assess the degree of agreement/disagreement between classifiers and might identify potential classifiers for fusion. In this sense, Kuncheva and Whitaker [62] studied different diversity measures as well as discussed their impacts on the final accuracy of ensemble systems.

Different works have been using diversity measures to select appropriate high-performance classifiers, but the challenge of finding the optimal number of classifiers for a target task has not been properly addressed yet. In general, the proposed solutions rely on the a priori use of ad hoc strategies for selecting classifiers, followed by the evaluation of their effectiveness results during training. Searching by the optimal number of classifiers, however, makes the selection process more expensive.

Currently, some of the most important challenges in MCS involve:

- choosing the best diversity measure to be used;
- combining different available measures for classifier selection in an ensemble system;
- finding out whether or not the existing measures describe the "real" diversity within the ensemble systems [12, 20].

Typically, works in the literature have adopted a single diversity measure or combined different measures using simple strategies (e.g., based on average of the measures scores [26, 112]). However, the aforementioned methods might not take full advantage of the different opinions provided by all of the available diversity measures. Moreover, another persistent problem in MCS approaches is how to combine different and non-correlated extraction and learning methods automatically.

In the literature, many works have been proposed to try sorting out problems cited previously as for example, the well-known AdaBoost [38] and Bagging [8] approaches. AdaBoost and Bagging ensemble approaches (see Section 2.3) have been used in several works in the literature due to their good results achieved in diverse applications. However, previous work has also shown their limitations in terms of efficiency, normalization, overfitting, and feature dimensionality problems. In [108], for example, training time has been a concern when more features were used to train an AdaBoost algorithm for face localization. The same problem has been reported in [72], which trained an AdaBoost algorithm for tracking indoor soccer players using videos. In [57, 97, 98], the authors discuss about another problem: the sensitivity of the classical AdaBoost algorithm to noisy datasets. They have proposed different solutions to reduce the overfitting effect caused in those cases. In [89, 90], the authors discuss the problems of feature normalization in the context of combining classifiers. More detail about tracking down fusion and classification problems can be found in [28].

The combination of multiple feature vectors defined by different image descriptors in AdaBoost and Bagging approaches is usually based on their concatenation (feature binding). Usually, when performing feature binding of different nature/domain, normalization techniques should be applied to standardize all feature values in the same range. For example, BIC feature vectors [96] have values in the range $0 - 9$, while LAS feature

vectors [103] have values in the range $0 - 1$. If we combine BIC and LAS features into a single feature vector, the BIC values might dominate. In fact, the normalization problem has been a great and difficult challenge for the machine learning community [90].

Another common problem faced when features are concatenated refers to the "curse of dimensionality" [109]. The curse of dimensionality problem is related to the fact that the dimension of the feature space increases in such a way that the available training instances become indistinguishable and not enough for allowing the definition of a good decision hyperplane [4].

1.2 Objective and Contributions

In this work, we seek an alternative to AdaBoost and Bagging ensembles. Our objective is to propose a stacking framework, able to perform automatic fusion of different visual properties and learning methods in existence in the literature for different multimedia recognition tasks.

In principle, the proposed framework has no concern about normalization issues nor has it with regard to feature dimensionality problems. The method assesses several descriptors and learning methods performing fusion in a final stage (late fusion) using a low-dimension feature vector and simple (fast) classifiers. Furthermore, the framework uses independent classifiers, being amenable for parallelization through Graphics Processing Unit (GPU), cluster architectures, or threads. Another difference of the proposed method, when compared to AdaBoost and Bagging techniques, is that the proposed framework seeks greater diversity between the simple classifiers being able to choose only the ones that effectively contribute to the solution of the classification problem of interest.

Diversity may be obtained in different ways such as using:

1. different learning methods and the same training set;
2. the same learning method and different training samples;
3. different methods using different types of classifier outcomes during the combination;
4. predictions as new attributes to train some learning method (meta-learning).

In this work, we use two out of four ways (1 and 4). We also use different visual properties (color, texture, and shape) to each of the learning methods chosen to be simple classifiers. We follow the concept that two instances of the same class have similar classification outputs for the same set of classifiers [54].

In this regard, in this thesis, we investigate the combination of several learning methods and image descriptors aiming at creating more effective classifiers. We propose a

framework for automatically combining the most discriminative classifiers using the support vector machine (SVM) technique, as well as exploring the use of diversity measures to select the less-correlated, yet effective, classifiers in three different selection strategies. We have performed experiments that demonstrate that the proposed framework for classifier fusion yields comparable results to the traditional fusion approaches but using less learning and description methods as well as not incurring in the curse of dimensionality problems, which are common to some fusion techniques. Another major advantage of the proposed method is that it yields good classification results using small training examples being more robust to the small sample size problem common in many classification techniques [4].

Our research hypothesis is that

> *appropriate classifier selection approaches can take advantage of classifier diversity to improve the accuracy performance of multiple classifier systems.*

The thesis contributions are:

- a framework for classifier selection and fusion through a meta-learning approach using Support Vector Machines techniques [31];
- a new classifier selection approach based on diversity measures consensus [32, 33];
- a new classifier selection approach based on *Kendall* correlation analysis [34]; and
- a new classifier selection approach based on rank aggregation techniques [35].

1.3 Thesis Organization

We organized the remainder of this thesis in five chapters. Chapter 2 presents related work and background concepts necessary for the understanding of this thesis. Chapter 3 describes the steps of the proposed framework for classifier fusion and three different strategies for selecting the most appropriate classifiers based on diversity measures. Chapter 4 shows the experimental protocol we devised to validate our work, while Chapter 5 discusses the results. Finally, Chapter 6 concludes the thesis and points out future research directions.

Chapter 2

Related Work and Background

This chapter is organized to five sections and aims to present related work and background concepts which are essential for a self-contained understanding of this thesis. Section 2.1 presents related work on image categorization and information fusion. Section 2.1 presents the fusion schemes used in our experiments. Section 2.2 describes each simple classifier used in the proposed framework. Section 2.3 comments on two well-known ensemble techniques that served as the baseline in the conducted experiments. Section 2.4 presents the diversity concept and five different diversity measures that have been employed in the classifier selection process. Section 2.5 presents the image descriptors used to extract visual properties from target datasets. Finally, Section 2.6 presents five different applications that have been considered in the validation of our proposed framework.

2.1 Image Categorization and Information Fusion

The importance and difficulty of visual data categorization have been discussed in several studies in the literature. In [65], large-scale classification has been addressed using 1.2 million images divided into 1,000 classes. In [19], experiments on classification with more than 10,000 image classes have been performed and allowed several observations about dataset scale, category density, and image hierarchy. Also, the study described in [114] addresses the scene categorization task, a fundamental problem in computer vision in a large dataset comprising 899 categories and 130,519 images. In [44], the authors performed experiments using a dataset with more than 500 classes and 500,000 images.

Image categorization is present in different areas of applications (e.g., medicine, remote sensing, and security) and likely the most popular approach towards the solution of the image classification problem consists in the use of machine learning methods. In medicine, Antonie et al. [1] have used learning methods for tumor detection/classification in digital mammography. In [100], a learning method was used for automatic medical

diagnosis based on parasite microscopy images. Also, the work presented in [58] has shown that it is possible to automate white blood cell (WBC) image classification and segmentation tasks. Machine learning is also extensively used in remote sensing applications to automate mapping and land cover inventories. It is applied, for instance, for crop recognition [22, 31] and classification of urban areas [22, 117]. Security issues such as face, iris, and fingerprinting recognition usually apply machine learning techniques to deal with specific challenges (e.g., large amount of data, large number of classes, and high dimensionality) [47, 94].

Dealing with such complex problems by using just one image characterization method (also known as feature descriptor) may produce ineffective results, demanding new solutions. In that scenario, information fusion may become mandatory, since different features may provide different, but complementary, information about the target data. Nakamura et al. [74] have defined information fusion as the combination of different sources to achieve improved information (cost, quality, or relevance). Furthermore, according to Ross et al. [92], information fusion may be performed in four levels: sensor, feature, rank, and decision.

Sensor or pixel level is the early stage of feature extraction, in which raw data are used to compose other richer data. This strategy has been widely used in fingerprint identification in which multiple fingerprint images are combined to compose a fingerprint image with more information, as in a mosaicing scheme [113].

Feature level fusion or early fusion is a strategy to handle coded data into a feature vector through some kind of description algorithm. This fusion can be as simple as a binding of different visual properties (e.g., color, texture, and shape) or more complex when using artificial intelligence (e.g., evolutionary algorithms [104] and support vector machine [4]). Therefore, several works have been proposed in the literature for feature fusion with good results. In [30, 69, 88], several evolutionary-based techniques (genetic programming, particle swarm optimization, and harmony search) were used to seek an optimum similarity function that combines different visual properties (e.g., color, texture, and shape) in image classification tasks. Also some works have used machine learning methods to complement feature fusion. For example, [40, 48] used multiple kernel learning (MKL) and LPBoost methods to optimize the weights in SVM function in object recognition applications. In [37], the authors have proposed a new logistic regression-based fusion method (LRFF) that explores a set of diverse and complementary visual words (color, shape, and texture) for image classification problems.

Rank level fusion is a technique that tries to combine different ranked lists of possible candidates sorted in decreasing order of confidence. In biometric systems, the idea is to create a consensus between different ranks finding the best match to new data [46]. In Content-based Image Retrieval (CBIR) systems, rank-level fusion can be used to combine

ranked lists from different kinds of image descriptors (e.g., color and texture) and then produce a final rank or rank aggregation [82].

Finally, the fourth approach for fusion is known as decision-level fusion or late fusion. Here, the averaging model can be considered the simplest and most popular way of combining classifiers, among which there are the majority and weighted voting methods widely used in works involving neural networks [3, 43, 84].

Amidst the different fusion methods, in [75], Tumer et al. have discussed the concept of ensemble systems. The basic idea of an ensemble system is to use a set of weak classifiers, usually of the same type, to obtain a powerful classifier. An ensemble system can be dependent or independent [91]. On one hand, dependent ensembles are those which use the output of a classifier to build the next classifier, taking advantage of previous experience and applying it in the next iterations (e.g., AdaBoost [38, 93]). On the other hand, the independent ensembles are those that combine different outputs of independent classifiers in the final stage of classification such as Bagging [8] and Random Forest [9] techniques. In addition, there are stacking systems that were introduced by Wolpert [110] and use the output of different learning methods as a basis to teach a new learning method and thus generate a new classification model (meta-learning) [29, 54].

Fusion Schemes

This section presents three fusion schemes (early, late, and hybrid), which we have used in our experiments.

Early Fusion

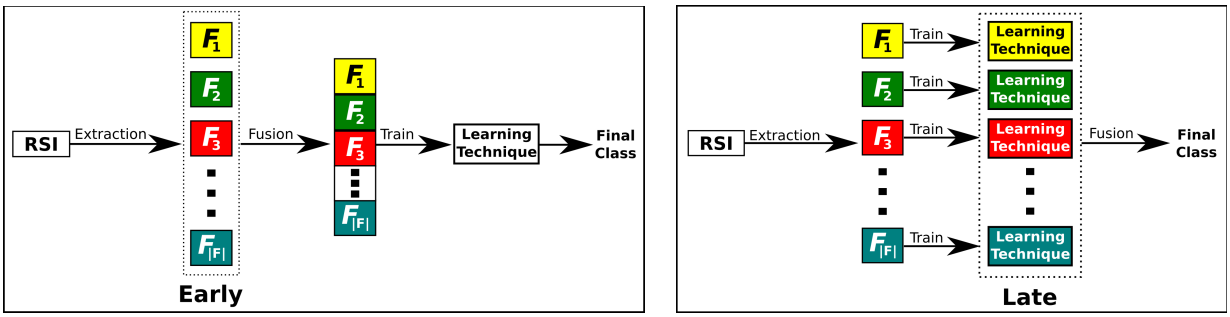
Early fusion, also known as feature-level fusion, refers to the fusion process that takes place before learning. Figure 2.1-(a) shows an example of the early fusion process from the feature extraction stage to the final classification stage. Given a remote sensing image (RSI), it first performs feature extraction with different image descriptors. Then it combines the different feature vectors in a single and augmented feature vector through feature binding (concatenation).

Late Fusion

Late fusion or decision-level fusion refers to the fusion process that takes place after the learning step (e.g., AdaBoost [93] and Bagging [8]). Figure 2.1-(b) shows an example of the late fusion process. Given a remote sensing image (RSI), if first uses different learning techniques obtaining the individual classification results. Then it combines the different decisions somehow (e.g., majority voting).

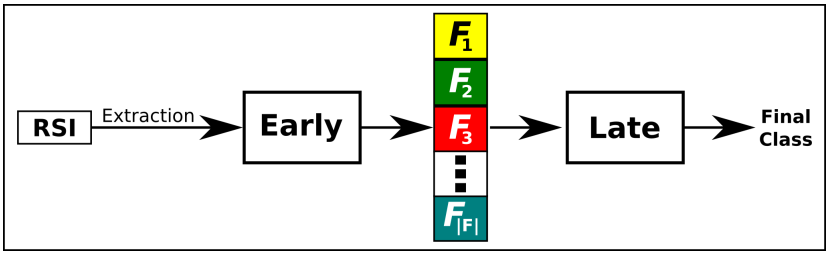
Hybrid Fusion

Refers to the fusion processes involving characteristics of both early and late fusion processes. Figure 2.1-(c) shows a hybrid fusion scheme.



(a) Early Fusion.

(b) Late Fusion.



(c) Hybrid Fusion.

Figure 2.1: Three different fusion schemes for remote sensing image classification.

2.2 Learning Methods

This section presents six simple learning techniques which have been combined by our meta-learning framework.

Decision Tree (DT)

Decision tree is one of the learning techniques most intuitive that exists in the literature. It presents a simple and easy way to understand the classification process [66].

DT is composed of three kinds of nodes: root, internal, and leaf or terminal. A root node is the initial node that has zero or more outgoing edges (square in Figure 2.2). Internal nodes are those that contain attributes (circles in Figure 2.2). Finally, leaf nodes are the ones at the end of branches and define a class (triangles in Figure 2.2) of a given input sample.

In this technique, two issues must be addressed [102]:

1. **The split policy for training records:** Typically, we use the entropy, impurity measure, (I) and the information gain of each attribute (Δ_{info}) to decide which attribute must be selected in the next recursive algorithm call;

2. **The stopping criteria for the splitting procedure:** One strategy could be the natural tree growing until each attribute is allocated in a single class. However, this strategy might result in trees that are too large to handle and lead to overfitting problems. Tree-pruning strategies are usually adopted to avoid these issues.

Equation 2.1 shows the gain, where $I(.)$ is the impurity measure from the given node, N is the number of sample at the parent node, k is the number of attributes, and $N(a_j)$ is the number of samples associated with the node a_j. Equation 2.2 shows the entropy, $p(i|a_j)$ denotes the fraction of samples from a class i at a given node a_j [102].

$$\Delta_{info} = I(parent) - \sum_{j=1}^{k} \frac{N(a_j)}{N} I(a_j) \tag{2.1}$$

$$Entropy(a_j) = -\sum_{i=0}^{c-1} p(i|a_j) \log_2 p(i|a_j) \tag{2.2}$$

Figure 2.2 shows an example of decision tree, where attributes of samples from the used dataset are a_1, a_2, a_3, a_4, and a_5. The branches or edges are possible values for each attribute.

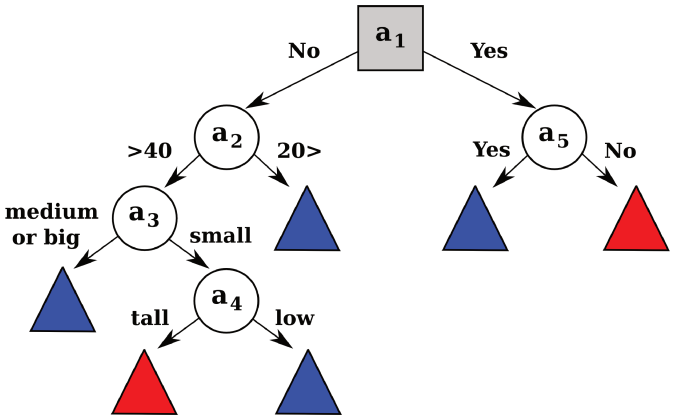


Figure 2.2: A decision tree created for binary classification task (blue and red classes). Notice that a decision tree might be composed of different attributes (e.g., binary, ordinal, and continuous).

The most used decision tree algorithms in the literature are ID3 [86] e C4.5 [87].

k-nearest neighbors (kNN)

The k-nearest neighbor classifier is a technique based on the closest training examples in the feature space [39]. Equation 2.3 illustrates an adjustment of kNN defined for x.

$$kNN(x) = \sum_{x_i \in \mathcal{N}_k(x)} y_i \tag{2.3}$$

where $\mathcal{N}_k(x)$ is the k-nearest neighbors from x in the training set, and y_i is a distance value among x and the current neighbor x_i (e.g., Euclidean distance). One common way to perform classification tasks might be deciding by majority voting of nearest neighbor.

Figure 2.3 illustrates an example of classification using kNN.

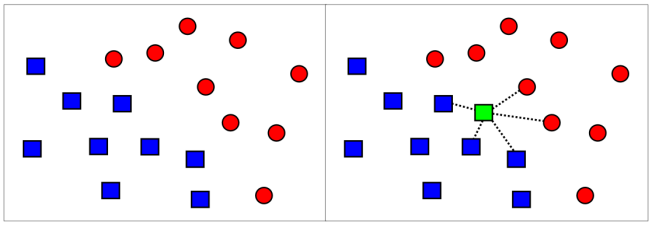


Figure 2.3: (a) Samples of two classes (square and circle) in the features space. (b) Given a new object, its $k = 5$ nearest neighbors will define its class. In this case, the green point is labeled by the blue class.

Naïve Bayes (NB)

Naïve Bayes is a simple probabilistic technique based on Bayes theorem to the problem of pattern classification. This technique assumes that the probability of each relevant attribute a_j is known and independent.

Equations 2.4 and 2.5 show the Bayes' formula, where $P(c_i)$ is the prior probability of the class c_i and $p(a_j|c_i)$ is a class-conditional probability densities function, $p(a_j)$ is the probability density function for a_j given that the state of nature is c_i [27].

$$P(c_i|a_j) = \frac{p(a_j|c_i) \times P(c_i)}{p(a_j)} \tag{2.4}$$

$$p(a_j) = \sum_{i=1}^{k} p(a_j|c_i)P(c_i) \tag{2.5}$$

Equation 2.6 shows an informal Bayes' formula from the Equation 2.4.

$$Posterior = \frac{likelihood \times prior}{evidence} \tag{2.6}$$

Naïve Bayes Tree (NBT)

Naïve Bayes Tree is a hybrid technique that induces a decision tree and Naïve bayes classifier. This technique has almost the same properties than decision trees (DT) with the additional Naïve Bayes (NB) classifier in the leaves for better deciding the class to which an input belongs. According to [59], NBT retains clean understanding of the techniques DT and NB and achieves better results in large databases.

Support Vector Machines (SVM)

Support Vector Machine is a machine learning method introduced in [6]. The goal is to construct an optimum hyperplane or set of hyperplanes, which can be used to separate an n-dimensional feature space. The hyperplane is calculated such that it maximizes the margin between two classes (the standard SVM is a two-class classifier). The margin can be seen as the minimum distance of one point of one class to the other. It can be interpreted as a separation measure between two classes and represents the separability degree between them (quality measure of classification). The points on borders between the classes are called support vectors. When it is not possible to find a linear separator for the classes, the data are mapped on-the-fly onto higher dimensional spaces through a non-linear mapping using the kernel trick [16]. The important detail here is that SVMs can efficiently perform non-linear classification. The reason for choosing SVM in this work is that by using the kernel, SVMs gain flexibility in the choice of the form of the threshold separating the classes of interest, which do not need to be linear and even do not need to have the same functional form for all data. Also, SVMs deliver a unique solution, since the optimality problem is convex.

Figure 2.4 illustrates the use of SVM to separate two classes. More details about this technique can be found in [6].

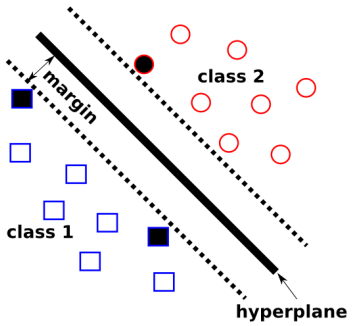


Figure 2.4: The SVM classifier builds a maximum margin decision hyperplane to separate two classes (squares and circles). Filled squares and circle are support vectors.

We use SVMs to combine multiple classifiers in our fusion framework (see Chapter 3).

2.3 Ensemble Techniques

This section introduces the widely used ensemble techniques from the literature: AdaBoost (BOOST) and Bagging (BAGG). Both techniques are used as baselines in our experimental protocol.

AdaBoost (BOOST)

The AdaBoost algorithm was proposed by Schapire [93] and is also an ensemble technique. It constructs an ensemble system (strong classifier) by repetitive evaluation of *weak classifiers*[1] in a series of rounds ($r = 1, \ldots, R$). In this section, we briefly describe the binary AdaBoost (algorithm AdaBoost.M1) proposed in [93] and implemented on Weka[2] data mining library. The multiclass AdaBoost [38] is a variation of this strategy.

Let T be the training set with m instances $T = ((\vec{x_1}, y_1), \ldots, (\vec{x_m}, y_m))$, let $\vec{x_i}$ be a feature vector associated with instance (image) i from some space X and $\vec{y_i} \in Y$ is the class label associated with $\vec{x_i}$. The strategy consists in keeping a set of weights $W_r(\vec{x_i})$ over R, where r is the current round. These weights can be interpreted as a measure of the difficulty level to classify each training sample. At the beginning, all the samples have the same weight ($W_1(i) = \frac{1}{m} \forall i$), but in each round, the weights of the misclassified samples are increased. Thus, in subsequent rounds the weak classifiers are forced to classify the harder samples.

For each round, the algorithm selects the best weak classifier or hypothesis $h_r(\vec{x_i})$ and computes a coefficient α_r that indicates the degree of importance of $h_r(\vec{x_i})$ in the final strong classifier or final hypothesis:

$$\alpha_r = \frac{\epsilon_r}{1 - \epsilon_r} \tag{2.7}$$

The classification error ϵ_r of a classifier h_r is given by:

$$h_r : \epsilon_r = \sum_{i:h_r(\vec{x_i}) \neq y_i} W_r(\vec{x_i}) \tag{2.8}$$

In the Weka implementation, the weak classifier is trained and selected based on the error on the training set T. The weights W_{r+1} are computed for set T based on the current

[1] A weak classifier is one that produces classification results slightly better than chance.

[2] `http://www.cs.waikato.ac.nz/~ml/weka` (As of May 2014).

weights W_r:

$$W_{r+1}(\vec{x_i}) = \frac{W_r(\vec{x_i})}{Z_r} \times \begin{cases} \alpha_r & \text{if } h_r(\vec{x_i}) = y_i \\ 1 & \text{otherwise} \end{cases} \tag{2.9}$$

where Z_r is a normalization constant.

At the end of R rounds, the strong classifier $f_{boost}(\vec{x})$ is given by a linear combination of R weak classifiers $h_r(\vec{x_i})$ and its coefficient α_r:

$$f_{boost}(\vec{x}) = \arg\max_{y \in Y} \sum_{r:h_r(\vec{x})=y} \log \frac{1}{\alpha_r}. \tag{2.10}$$

Figure 2.5 illustrates the training steps of the AdaBoost approach.

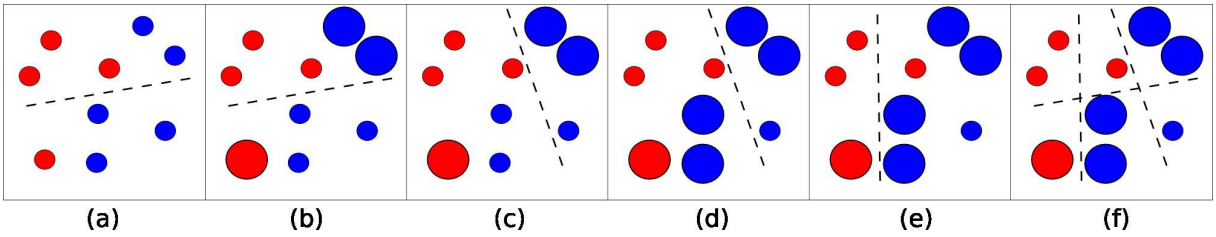


Figure 2.5: Initially, weight each training example equally. (a) Weak Classifier 1 is applied; (b) Weights are increased; (c) Weak Classifier 2 is applied; (d) Weights are increased; (e) Weak Classifier 3 is applied; (f) Final classifier is a linear combination of all weak classifiers.

Bootstrap Aggregation (BAGG)

Bootstrap aggregation (Bagging) approach is an ensemble technique which aims at evaluating the predictions on an image collection (bootstrap samples) [8]. Formally, let T be an initial training set which is divided into B equal parts or subsets $Z^i, i = 1, 2, ..., B$ [39]. Each subset is used for training B weak classifiers. After training, each weak classifier obtains a model α_i that will be used in the classification step. Given a new instance (testing image), each model α_i is used to determine its class. The class that will be assigned to a testing image is defined by the majority voting between the B weak classifiers [4]. For each image represented by a feature vector $\vec{x}$, a prediction $\vec{f^i}(\vec{x})$ from each classifier is obtained and the result is calculated:

$$f_{bag}(\vec{x}) = \frac{1}{B} \sum_{i=1}^{B} \vec{f^i}(x) \tag{2.11}$$

Figure 2.6 illustrates the training and the classification steps of the Bagging approach.

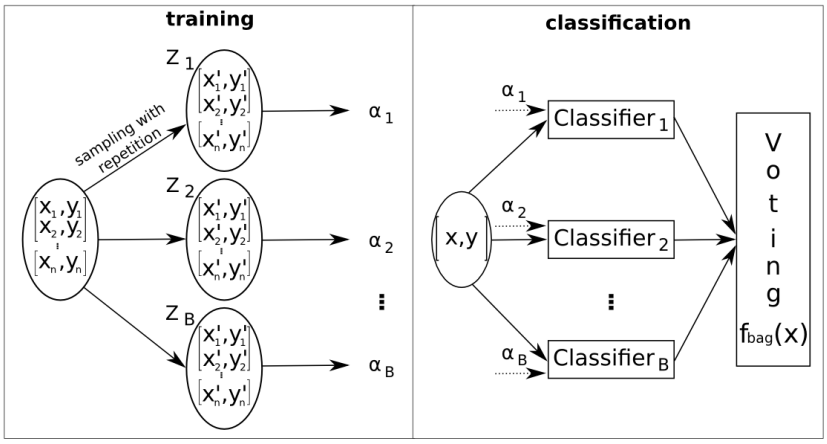


Figure 2.6: Training and classification steps using a Bagging approach.

2.4 Diversity Measures

Diversity is the degree of agreement/disagreement between involved classifiers pointing out the most interesting ones to be further used in a combination scheme. To achieve this diversity score or quantitative value inside ensemble systems, diversity measures have been used. In [61, 62], Kuncheva et al. presented several measures to assess diversity, considering pairs of classifiers.

Let $\mathcal{M}$ be a matrix containing the relationship between a pair of classifiers with percentage of concordance. Table 2.1 shows a relationship matrix $\mathcal{M}$ with percentage of hit and miss for two exemplifying classifiers c_i and c_j. The value a is the percentage of regions that both classifiers c_i and c_j classified correctly in a validation set. Values b and c are the percentage of regions that c_j hit and c_i missed and vice-versa. The value d is the percentage of regions that both classifiers missed.

Table 2.1: Relationship matrix $\mathcal{M}$ between two classifiers c_i and c_j.

	Hit c_i	Miss c_i
Hit c_j	a	b
Miss c_j	c	d

In our experiments, we have used *Double-Fault Measure* (DFM), *Q-Statistic* ($QSTAT$), *Interrater Agreement k* (IA), *Correlation Coefficient* ρ (COR), and *Disagreement Measure* (DM). Those measures are defined as follows.

$$COR(c_i, c_j) = \frac{ad - bc}{\sqrt{(a+b)(c+d)(a+c)(b+d)}}, \tag{2.12}$$

$$DFM(c_i, c_j) = d, \tag{2.13}$$

$$DM(c_i, c_j) = \frac{b+c}{a+b+c+d}. \tag{2.14}$$

$$QSTAT(c_i, c_j) = \frac{ad - bc}{ad + bc}, \tag{2.15}$$

$$IA(c_i, c_j) = \frac{2(ac - bd)}{(a+b)(c+d) + (a+c)(b+d)}, \tag{2.16}$$

The diversity is greater if the measures *Double-Fault Measure*, *Q-Statistic*, *Interrater Agreement* k, and *Correlation Coefficient* p are lower among pairs of classifiers c_i and c_j. In the case of the *Disagreement Measure*, the greater the measure, the greater the diversity [61, 62]. Ranges of COR, $QSTAT$, and IA are in $[-1, 1]$ while DFM and DM are in $[0, 1]$.

Figure 2.7 depicts a toy example of computed diversity measures associated with pairs of classifiers. In Figure 2.7(a), the first row shows the ground truth (GT) data (oracle response for each testing example) and the three additional rows show the output of three different classifiers (C_1, C_2, and C_3). The columns are instances that have been predicted by the classifiers. Red outputs denote misclassifications. Figure 2.7(b) shows the relationship between all possible combinations of pairs of classifiers, a total of three ($C_1 \times C_2$, $C_1 \times C_3$, and $C_2 \times C_3$). In addition, the figure also presents the score values of two diversity measures (DFM and QSTAT) for each pair of classifier. Figure 2.7(c) shows the ranked lists, one for each diversity measure. Notice that according to both lists, the best pair of classifiers to be combined is C_1 and C_2. However, it is not always possible to come up with a single solution for the best set of classifiers to combine and conflicts may be possible as can be seen in the second and third positions in this example highlighted in red in Figure 2.7(c)). This shows that the measures may generate different lists. This issue is also of interest and is discussed in more details in Section 3.2.2.

GT	A	A	B	B	C	C	D	D	E	E	100%
C_1	A	A	B	B	A	C	D	D	E	E	90%
C_2	A	A	C	B	C	C	D	B	E	E	80%
C_3	A	A	A	B	A	C	D	D	E	E	80%

(a) Ground-truth (GT) and classifiers outcomes

Relationship Matrix

C_1 x C_2		C_1 x C_3		C_2 x C_3	
7	1	8	0	7	1
2	0	1	1	1	1

Diversity Measures

	C_1 x C_2	C_1 x C_3	C_2 x C_3
DFM:	0.00	1.00	1.00
QSTAT:	-1.00	1.00	0.75

(b) Diversity measures calculated

DFM

C_i	C_j	SCORE
C_1	C_2	0.00
C_1	C_3	1.00
C_2	C_3	1.00

QSTAT

C_i	C_j	SCORE
C_1	C_2	-1.00
C_2	C_3	0.75
C_1	C_3	1.00

(c) Ranked lists

Figure 2.7: Example of ranked lists for two different diversity measures used in our work.

2.5 Image Descriptors

As we stated in Chapter 1, there is no silver bullet to solve all image classification problems with just one machine learning classifier or even with just one image characterization technique. To choose the most appropriate descriptors is also a difficult task.

This is where this work's contribution shines. Our framework can consider a diverse set of classifiers and descriptors and point out the most interesting ones to solve a problem. In this sense, here we have used several image descriptors comprising color-, texture-, and shape-based methods. The used color descriptors include Color Autocorrelogram (ACC) [50], Border/Interior Pixel Classification (BIC) [96], Color Coherence Vector (CCV) [78], Global Color Histogram (GCH) [101], and Local Color Histogram (LCH) [101]. The used texture descriptors include Local Activity Spectrum (LAS) [103], Quantized Compound Change Histogram (QCCH) [49], Statistical Analysis of Structural Information (SASI) [13], Steerable Pyramid Decomposition (SID) [116], and Unser [106]. The used shape descriptors include Edge Orientation Autocorrelogram (EOAC) [68], and Spherical Pyramid-Technique (SPYTEC) [63].

The criteria for choosing the image descriptors, for each dataset, are based on extensive experiments performed in [18, 24, 83] pointing out some of the most interesting image descriptors in the current computer vision literature.

Color Autocorrelogram (ACC)

The role of this descriptor is to map the spatial information of colors by pixel correlations at different distances. It computes the probability of finding in the image two pixels with color C at distance d from each other. For each distance d, m probabilities are computed, where m represents the number of colors in the quantized space. The implemented version quantized the color space into 64 bins and considered 4 distance values (1, 3, 5, and 7) [50].

Border/Interior Pixel Classification (BIC)

BIC has been successful in many applications [21, 45, 70, 99]. The first step of the feature vector extraction process relies on the classification of image pixels into *border* or *interior* ones. When a pixel has the same spectral value in the quantized space as its four neighbors (the ones which are above, below, on the right, and on the left), it is classified as *interior*. Otherwise, the pixel is classified as *border*. Two histograms are computed after the classification: one for the interior pixels and another for the border ones. Both histograms are merged to compose the feature vector. The implemented version quantized the color space into 64 bins [96].

Color Coherence Vector (CCV)

This descriptor uses an extraction algorithm that classifies the image pixels as "coherent" or "incoherent" pixels. This classification takes into consideration whether the pixel belongs or not to a region with similar colors, that is, coherent regions. Two color histograms are computed after quantization: one for coherent pixels and another for incoherent ones. Both histograms are merged to compose the feature vector. In our experiments, the color space was quantized into 64 bins [78].

Global Color Histogram (GCH)

GCH is one of the most commonly used descriptors. It uses an extraction algorithm which quantizes the color space in a uniform way and it scans the image computing the number of pixels belonging to each bin. The size of the feature vector depends on the quantization used. In the present work, the color space was split into 64 bins, thus, the feature vector has 64 values [101].

Local Color Histogram (LCH)

LCH is an extension of GCH descriptor, which makes the concatenation of color histograms (GCH) of 64 bins, obtained from an image divided by a grid 4×4. The imple-

mented version has a feature vector with 1,024 dimensions [101].

Local Activity Spectrum (LAS)

LAS descriptor captures textures spatial activity in four different directions separately: horizontal, vertical, diagonal, and anti-diagonal. The four activity measures are computed for a pixel (i, j) by considering the values of neighboring in the four directions. The values obtained are used to compute a histogram that is called *local activity spectrum*. Each component g_i is quantized independently. In our experiments, each component was non-uniformly quantized into 4 bins, leading to a histogram with 256 bins [103].

Quantized Compound Change Histogram (QCCH)

QCCH uses the relation between pixels and their neighbors to encode texture information. This descriptor generates a representation invariant to rotation and translation. Its extraction algorithm scans the image with a square window. For each position in the image, the average gray value of the window is computed. Four variation rates are then computed by taking into consideration the average gray values in four directions: horizontal, vertical, diagonal, and anti-diagonal directions. The average of these four variations is calculated for each window position, they are grouped into 40 bins and a histogram of these values is computed [49].

Statistical Analysis of Structural Information (SASI)

SASI descriptor encodes texture properties based on structural properties from textures image. Feature extraction algorithm scans an image with different windows resolutions and orientations. The used implementation has three different windows sizes (3×3, 5×5 and 7×7 pixels) and used four directions (0°, 45°, 90°, and 135°). SASI encodes spectral information from each window in different direction by calculating auto-correlation values. The final vector is composed of 64 values [13].

Edge Orientation Autocorrelogram(EOAC)

EOAC is a shape descriptor. We chose this descriptor because it does not depend on segmentation to extract features. Its strategy is to classify the image edges according to two aspects: the edge orientation and the correlation between neighbor edges. The first step is to compute the image gradient from the input image. Then, the algorithm computes an edge orientation auto-correlogram. The feature vector is composed of the values from this auto-correlogram. In this implementation, we use angle quantization in

72 segments of 5° degrees each one; four distance values (1, 3, 5, and 7); the Sobel operator to compute the image gradient; and a gradient threshold equal to 25, as suggested in [68]. The final vector is comprised by 288 values.

Spherical Pyramid-Technique (SPYTEC)

Spytech descriptor handles images in gray scales and extracts edges information of the image through Discrete Wavelet Transform (DWT) and Sobel. Normalized coefficients of DWT with greater magnitude are stored resulting in a feature vector with 16 dimensions [63].

Steerable Pyramid Decomposition (SID)

SID descriptor employs a set of filters sensitive to different scales and orientations to process the image. The image is first decomposed into two sub-bands using a high-pass and a low-pass filter. After that, the low-pass sub-band is decomposed recursively into K sub-bands by band-pass filters and into one sub-band by a low-pass filter. Various directional information about each scale is captured by each recursive step. The mean and standard deviation of each sub-band are used as feature values. To obtain the invariance to scale and orientation, circular shifts in the feature vector are applied. The implemented version uses two scales and four orientations, which gives a final feature vector with 16 values [116].

Unser

Unser descriptor is based on co-occurrence matrices, still one of the most widely used descriptors to encode texture in remote sensing applications. Its extraction algorithm computes a histogram of sums H_{sum} and a histogram of differences H_{dif}. The histogram of sums is incremented considering the sum, while the histogram of differences is incremented by taking into account the difference between the values of two neighbor pixels. As well as gray level co-occurrence matrices, measures such as energy, contrast, and entropy can be extracted from the histograms. In our experiments, eight different measures were extracted from histograms and four angles were used (0°, 45°, 90°, and 135°). The final feature vector is composed of 32 values [106].

2.6 Applications

This section describes applications considered in the validation of the proposed approaches.

2.6.1 Object Recognition

One of the major challenges in computer vision and machine learning is object recognition task [36]. This task involves several challenges such as variation of visual properties (e.g., shapes, colors, and textures), different vantage points (e.g., front, side or back), in many places (e.g., indoor or outdoor) and sizes. Objects might be partially clogged and are semantically dependent.

To address this application, we adopted the Caltech101 dataset [36] in this thesis. Challenges of using this dataset are based on large number of classes (101) and images (9,145). Furthermore, the number of images per class varies from 40 to 800 (unbalanced classes). Figure 2.8 shows some images of the Caltech101 dataset.

Figure 2.8: *Accordion class from the Caltech101 dataset.*

2.6.2 Produce Recognition

Fruit and vegetable recognition is a recurrent task in supermarkets [5, 89]. One common application is concerned with the definition of the price of a produce, given its identification. This is a challenging problem as it deals with both different species of fruits and vegetables (e.g., apple, orange, potatoes) and many varieties of a single produce species (for example, Golden Delicious, Akane, Gala, and Fuji are different varieties of apples) [89].

Usually, existing recognition approaches are not automatic and demand long-term and laborious previous training sessions. One attempt to address that problem concerns with the use of barcodes that are assigned to packages of fruits/vegetables. A drawback of this solution relies on the lack of freedom on choosing the produce of interest. Another solution consists in using booklets containing photos of fruits/vegetables that are browsed to properly determine their price. That solution, however, poses new challenges related to the memorization and the subjectivity in the recognition process.

To address this application, we adopted the supermarket produce dataset used [89] in this thesis. Challenges of using this dataset are based on different pose, the number of elements within an image, and illumination that represent a more realistic scenario.

The dataset comprises 15 different categories: Plum (264), Agata Potato (201), Asterix Potato (182), Cashew (210), Onion (75), Orange (103), Taiti Lime (106), Kiwi (171),

Fuji Apple (212), Granny-Smith Apple (155), Watermelon (192), Honeydew Melon (145), Nectarine (247), Williams Pear (159), and Diamond Peach (211); totalizing 2633 images. The number of images per category varies from 75 to 247 images (unbalanced classes). Figure 2.9 shows some images of the supermarket produce dataset used.

Figure 2.9: *Four different images from the Supermarket Produce dataset.*

2.6.3 Remote Sensing Image Recognition

New challenges and opportunities in Remote Sensing Image (RSI) classification have emerged due to the recent advances in sensor technologies [2]. The increasing availability of spectral, temporal, and spatial resolution imagery produces a large amount of data and enables deeper and more detailed image analysis for different applications [64,85,105,115]. In this scenario, novel machine learning and image processing strategies are demanded to handle those huge and complex image sets and convert them into more useful information.

In crop recognition applications, for example, different factors can influence in the spectral response of the crops as kind of region (e.g., mountainous or flat), age of tree (e.g., younger or older) and even used spacing between plants. In mountainous regions, the spectral patterns tend to be affected by the topographical differences and interference generated by shadows. In seasonal crops, there are different growing stages of crops which result in another difficult challenge in terms of classification. In this case, each stage might has a different spectral pattern, but all of them represent the same culture or class.

To address this application, we adopted two remote sensing images (Coffee and Urban) that have been used in different works in our research group [21–23, 25].

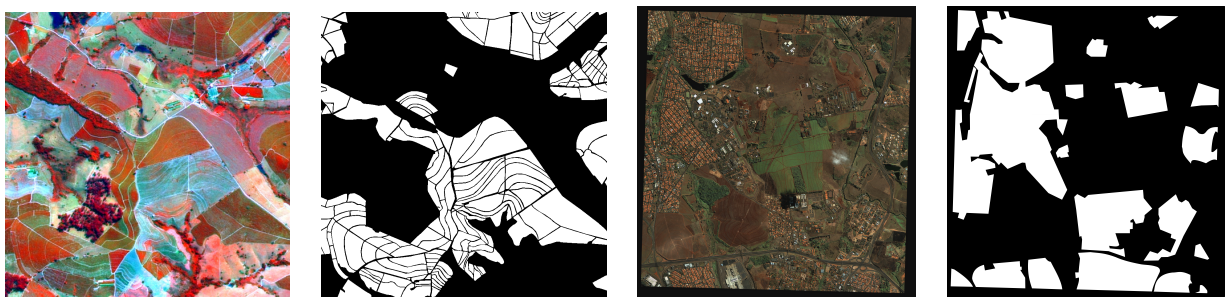
Coffee Dataset

In this work, we consider $4,885$ regions ($1,006$ coffee and $3,879$ non-coffee) created via the method for multi-scale segmentation proposed by Guigues et al. [41], which separated into regions a SPOT satellite image of $1,000 \times 1,000$ pixels (Figure 2.10-(a)). The SPOT satellite image corresponds to the Monte Santo de Minas county, in the State of Minas Gerais, Brazil, a traditional place of coffee cultivation. The region where this image was captured is mountainous. Therefore, the spectral patterns tend to be affected by the

topographical differences and interference generated by the shadows. Another problem is that coffee is not a seasonal crop. Thus, in the same area, there may be crops of different ages. Concerning classification aspects, we have several completely different patterns representing the same class while some of these patterns are much closer to other classes. To evaluate the accuracy, we use a ground truth that indicates all coffee regions in the image. As the experiments were performed with region level image and the ground truth (Figure 2.10-(b)) is in pixel level, it was necessary to define a rule to label each region: if more than 80% of a region contain pixels of coffee, that region was labeled as "coffee"; otherwise it is a non-coffee region. Figure 2.10 (a) illustrates the Coffee image, while Figure 2.10 (b) indicates the coffee crop in the Coffee image.

Urban Dataset

This dataset is a Quickbird scene taken in 2003 from Campinas region, Brazil. This scene is composed of 1 million pixels ($1,000 \times 1,000$) with spatial resolution equal to 0.62 meters. It is composed of three bands that correspond to the visible spectrum (red, green, and blue). We have empirically created the ground truth based on our knowledge about the region. We considered as urban the places which correspond to residential, commercial, or industrial regions. Highways, roads, native vegetation, crops, and rural buildings are considered non-urban areas. The experimental protocol is the same of the Coffee dataset and $5,362$ regions ($1,698$ urban and $3,664$ non-urban) have been created. Figure 2.10-(c) illustrates the Urban image, while Figure 2.10-(d) indicates the urban areas in the Urban image.



(a) (b) (c) (d)

Figure 2.10: Coffee data with (a) original RSI and (b) ground truth that indicates the regions that correspond to coffee crop. Urban data with (c) original RSI and (d) ground truth that indicates the regions that correspond to urban areas. In (b) and (d), white and black regions are coffee/non-coffee crops and uban/non-urban area, respectively.

2.6.4 Natural Scene Classification

Natural scene classification is not a trivial task due to the high dependence on semantics employed by users. A problem present in images of natural scene datasets is multi-labels instance, which a target object can belongs to different classes. For example, the classes "Ocean" and "River" can have the same target object "water" in both classes and resulting in a major challenge for machine learning techniques [7].

To address this application, we adopted the FreeFoto dataset extracted from a larger website FreeFoto.com composed of 171 sections and $3,542$ categories totalizing $129,559$ images. The FreeFoto dataset used in this thesis is the same previously used in other work in the literature [69], and comprises $3,462$ scene images and 9 categories. The number of images per class varies from 70 to 854. Figure 2.11 shows some images of the Freefoto dataset.

Figure 2.11: Five different images from the Freefoto dataset.

Chapter 3

Classifier Selection and Fusion Framework

Given a visual classification problem, we have a set of characterization or description techniques (descriptors) and a set of learning methods that will be used to learn patterns from available instances for training in order to classify new and unseen instances.

Once we train all necessary classifiers along with different image descriptors, the learned knowledge undergoes a selection process of the most relevant learning methods and descriptors to be combined by another learning method (meta-learning approach) aiming at selecting the most discriminative methods as well as boosting the classification performance at test time by selecting less, but more effective classifiers.

The classifiers (herein one classifier is a tuple learning/descriptor) are selected in a selection process that uses diversity measures calculated at training time to show the degree of agreement/disagreement between involved classifiers pointing out the most interesting ones to be further used in a combination scheme.

This chapter presents our framework for classifier selection and fusion. Sections 3.1, 3.2, and 3.3 present a formal description of our framework for classifier fusion and three selection approaches along with examples when necessary. Finally, Section 3.4 presents several advantages and some research directions that might be explored in future work.

3.1 Formalization of Selection and Fusion Framework

Let $\mathcal{L}$ be any set of learning methods (e.g., Decision Tree, Naïve Bayes, and kNN) and $\mathcal{F}$ be a set of image descriptors (e.g., Color Histogram). Suppose that classifiers are created by combining each available learning method with each image descriptor. For example, three classifiers could be created by combining the learning methods Decision Tree, Naïve Bayes, and kNN with the Color Histogram descriptor. Let $\mathcal{C}$ be the set of classifiers

created by that combination, where $|\mathcal{C}| = |\mathcal{L}| \times |\mathcal{F}|$.

Let $\mathcal{S}$ be a set of images, where the class of $s_i \in \mathcal{S}$ $(1 < i \leq |\mathcal{S}|)$ is known. The set $\mathcal{S}$ is used to construct both the training (T) and validation (V) sets, where $T \cup V = \mathcal{S}$ and $T \cap V = \emptyset$. As we consider a supervised learning scenario, the actual classes for training and validation data points are known *a priori*.

Initially, all classifiers $c_j \in \mathcal{C}$ $(1 < j \leq |\mathcal{C}|)$ are trained on the elements of the set T. Next, the outcome of each classifier on the validation set V is computed and stored into a matrix M_V, where $|M_V| = |V| \times |\mathcal{C}|$ and $|V|$ is the number of image in a validation set V. The actual classes of training and validation data points are known *a priori*.

In the following, M_V is used as input to select a set $\mathcal{C}^* \subset \mathcal{C}$ of classifiers that are good candidates to be combined. In our approach, diversity measures are employed to determine $\mathcal{C}^*$ (see Section 3.2). Note that a new matrix $M_V^* \subset M_V$ is created by using the selected classifiers in $\mathcal{C}^*$.

Given a new image I, we use each classifier $c_k \in \mathcal{C}^*$ $(1 < k \leq |\mathcal{C}^*|)$ to determine the class of I, producing k outcomes. The k outcomes are used as input of a fusion technique (e.g., majority voting and SVM) that takes the final decision regarding the definition of the class of I. In the case of a fusion technique that requires prior training (e.g., SVM), M_V^* is used.

Figure 3.1 illustrates the proposed framework for combining classifiers.

3.2 Classifier Selection Approaches based on Diversity Measures

In this section, we present three different classifier selection approaches that have been proposed in this work [33–35]. In Section 3.2.1, we describe a selection and fusion framework, which uses a consensus approach to combining five different diversity measures to select classifiers [33]. In Section 3.2.2, we introduce a novel strategy for selecting classifiers to be combined based on the correlation of different diversity measures [34]. Different from previous contributions, in Section 3.2.3, we introduce a new strategy for guiding the selection of classifiers based on the combination/fusion of multiple diversity and evaluation measures, using rank aggregation approach [35].

The three proposed selecting and combination approaches are general enough to be used in a diverse set of problems in the literature.

3.2.1 Selection based on Consensus

Consider the previously defined $\mathcal{C}$ (set of classifiers) and M_V (a matrix such that $|M_V| = |V| \times |\mathcal{C}|$), containing the outcomes of all $\mathcal{C}$ classifiers from the validation set

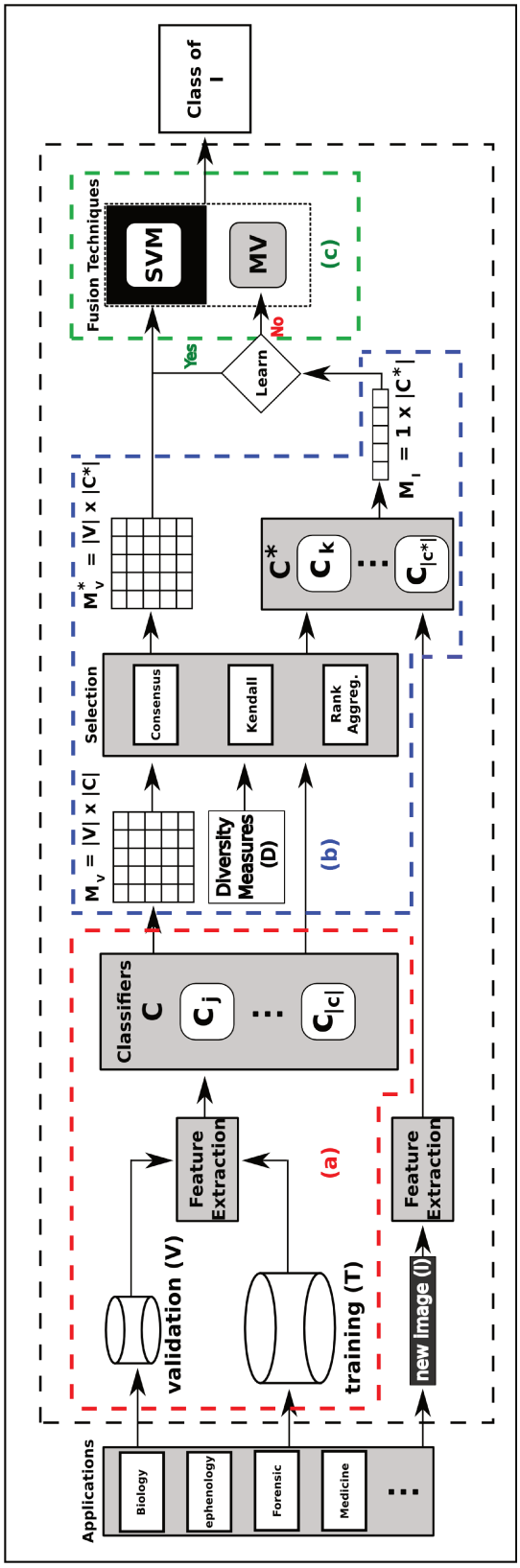


Figure 3.1: Proposed framework for classifier fusion and selection. (a) Given a classification problem with training examples, we train different classifiers. (b) By means of diversity measures, we select the most discriminatives classifiers. (c) Combine classifiers in a meta level using any other classifier. Notice that, in this particular example, both the SVM and Majority Voting (MV) techniques can be used for classifier fusion.

V.

Let $\mathcal{D}$ be a set of diversity measures. Each diversity measure $d_\ell \in \mathcal{D}$ is used to compute the agreement and disagreement between two classifiers $c_i, c_j \in \mathcal{C}$, considering all possible combinations of classifiers (arrow (a) in Figure 3.2).

Let $\mathcal{R}_{d_\ell} = \{(c_i, c_j), score_{d_\ell}(c_i, c_j), acc(c_i), acc(c_j)\}$ be a ranked list of pairs of classifiers defined by the score of the diversity measure d_ℓ and the accuracy values of each classifier (c_i and c_j) computed from the validation set.

Let $\mathcal{R} = \{\mathcal{R}_{d_1}, \mathcal{R}_{d_2} \ldots \mathcal{R}_{d_{|\mathcal{D}|}}\}$ be the set of ranked lists defined for each available diversity measure. This process is illustrated by arrow (b). Let $\mathcal{R}^t$ be a set of ranked lists, where each ranked list contains the top t pairs of classifiers (t pairs of classifiers that are good candidates to be combined) – arrow (c), and $\mathcal{H}$ be a histogram that counts the number of occurrences of a classifier in all ranked lists of $\mathcal{R}^t$ – arrow (d). Finally, the most frequent classifiers in $\mathcal{H}$, whose accuracy is greater that a given threshold $\mathcal{T}$, are combined by a fusion approach – arrow (e). $\mathcal{T}$ is a threshold defined in terms of the average accuracy among all classifiers using validation set V.

The top t value has been found through empirical search. From now on, all the experiments reported in this thesis consider $t = 100$ in selection approaches. Perhaps an additional and deeper study upon top values t might be conducted in future work.

Figure 3.2 illustrates the adopted five-step approach for selecting classifiers based on diversity measures.

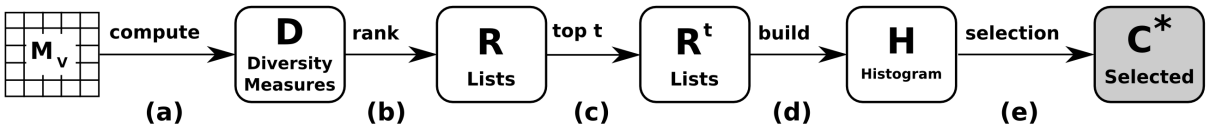


Figure 3.2: The five steps for classifier selection are: (a) Computation of diversity measures from the validation matrix M_V; (b) Ranking of pairs of classifiers by their diversity measure scores; (c) Selection of the top $t = 100$ ranked pairs of classifiers; (d) Computation of a histogram $\mathcal{H}$ that counts the number of occurrences of each classifier; (e) Selection the most appropriate classifiers $|\mathcal{C}^*|$ based on their occurrence in $\mathcal{H}$ and on a defined threshold $\mathcal{T}$.

Algorithm 1 outlines the proposed steps for selecting classifiers, by taking into account diversity measures. Lines 1-10 refer to the use of diversity measures for defining ranked lists containing pairs of classifiers (arrows (a) and (b) of Figure 3.2). Next, the top-ranked pairs of classifiers are selected in Line 11 (arrow (c)) and the number of occurrences of each classifier is determined in Lines 12-20 (arrow (d)). Finally, the most suitable classifiers to be used in the fusion step are defined in Line 21 (arrow (e)).

Algorithm 1 *Selection of classifiers based on diversity measure consensus*

Input: set $\mathcal{D}$ of diversity measures, set $\mathcal{C}$ of classifiers, and the outcomes of classifiers on validation set V encoded in M_V.

$\mathcal{R} \leftarrow \emptyset$
for each $d_\ell \in \mathcal{D}$ **do**
  $\mathcal{R}_{d_l} \leftarrow \emptyset$
  for each pair $(c_i, c_j) \in \mathcal{C} \times \mathcal{C}$ **do**
    $score_{d_l}(c_i, c_j) \leftarrow d_l(c_i, c_j)$
    $\mathcal{R}_{d_l} \leftarrow \mathcal{R}_{d_l} \cup \{((c_i, c_j), score_{d_l}(c_i, c_j))\}$
  end for
  Sort $\mathcal{R}_{d_l}$ with regard to $score_{d_l}$
  $\mathcal{R} \leftarrow \mathcal{R} \cup \mathcal{R}_{d_l}$
end for
$\mathcal{R}^t \leftarrow$ select the top t ranked pairs of classifiers for each ranked list in $\mathcal{R}$
for each $c_j \in \mathcal{C}$ **do**
  $\mathcal{H}(c_j) \leftarrow 0$
end for
for each $d_\ell \in \mathcal{D}$ **do**
  for each $((c_i, c_j), score_{d_l}(c_i, c_j)) \in \mathcal{R}^t_{d_\ell}$ **do**
    $\mathcal{H}(c_i)_{++}$
    $\mathcal{H}(c_j)_{++}$
  end for
end for
$\mathcal{C}^* \leftarrow \{c_i \in \mathcal{C}$, such as $|\mathcal{C}^*| = h$ and $\forall c_j \in \mathcal{C} \setminus \mathcal{C}^*$, $\mathcal{H}(c_i) > \mathcal{H}(c_j), Accuracy(c_j) > \mathcal{T}, Accuracy(c_i) > \mathcal{T}\}$

3.2.2 Selection based on Kendall Correlation

In this section, we expand upon previous work in the literature [33] and introduce a new strategy for guiding the selection of classifiers based on the opinion of multiple selected diversity measures.

We propose to use multiple diversity measures to determine which classifiers should be combined. Our hypothesis is that by exploring complementary information provided by different diversity measures, more appropriate classifiers are selected to be combined.

Recall from Section 2.4 that a diversity measure indicates the agreement of pairs of classifiers. In that sense, different diversity measures would rank pairs of classifiers differently. Therefore, we propose to explore different strategies to select classifiers based on *correlation* scores among ranked lists of pairs of classifiers. Ranked lists are defined by different diversity measures.

Defining ranked lists of pairs of classifiers

As mentioned before, let $\mathcal{C}$ be the set of classifiers created by the combination of learning methods and image descriptors. Let $\mathcal{P} = \{p_1, p_2, \ldots, p_{|\mathcal{C}\times\mathcal{C}|}\}$ be a set of all possible pairs of classifiers, i.e., $p_l = (c_i, c_j)$, where $(c_i, c_j) \in \mathcal{C} \times \mathcal{C}$.

Let $\mathcal{D} = \{d_1, d_2, \ldots, d_{|\mathcal{D}|}\}$ be a set of diversity measures, such that each diversity measure $d_k \in \mathcal{D}$ defines a distance function $\rho : \mathcal{P} \rightarrow \mathbb{R}$, where $\mathbb{R}$ denotes real numbers. Equations described in Section 2.4 that define different criteria for implementing the function ρ. Consider $\rho(p_l) \geq 0$ for all $p_l \in \mathcal{P}$ and $\rho(p_l) = 0$, with $p_l = (c_i, c_j)$, if $c_i = c_j$. The distance $\rho(p_l)$ among all pairs of classifiers $p_l = (c_i, c_j) \in \mathcal{C} \times \mathcal{C}$ can be computed to obtain a $|\mathcal{C}| \times |\mathcal{C}|$ distance matrix A.

Given a diversity measure $d_k \in \mathcal{D}$, we can compute a ranked list $\mathcal{R}_{d_l}$ by taking into account the distance matrix A. The ranked list $\mathcal{R}_{d_l}$=$\{p_1,\ p_2,\ \ldots,\ p_{|\mathcal{C}\times\mathcal{C}|}\}$ (where $p_l = (c_i, c_j)$ is a pair of classifiers) can be defined as a permutation of the collection $\mathcal{P}$, such that, if p_l is ranked at lower positions than p_m, i.e., p_l is ranked before p_m, then $\rho(p_l)$ $< \rho(p_m)$. In this way, pairs of classifiers are ranked according to their agreement score defined in terms of a diversity measure.

Measuring the correlation of ranked lists

We propose to exploit the correlation of ranked lists of pairs of classifiers to select the more appropriate ones to be combined. In this thesis, we use the *Kendall* tau rank correlation coefficient (τ) [56] to measure the degree of concordance between two different ranked lists of the same set of observed samples. We will use only the term '*Kendall*' for *Kendall* tau rank, thus avoiding possible confusion with 'tau' index (evaluation measure).

The *Kendall* correlation $\tau(\mathcal{R}_{d_i}, \mathcal{R}_{d_j})$ between two ranked lists $\mathcal{R}_{d_i}$ and $\mathcal{R}_{d_j}$ is defined in terms of the number of concordant pairs NC in $\mathcal{R}_{d_i}$ and $\mathcal{R}_{d_j}$, the number of discordant pairs ND, and the number of positions n in the ranked lists.

Equation 3.1 defines the *Kendall* correlation:

$$\tau(\mathcal{R}_{d_i}, \mathcal{R}_{d_j}) = \frac{NC - ND}{\frac{1}{2}n(n-1)}, \tag{3.1}$$

Notice that n positions in the ranked list, results in $\binom{n}{2} = \frac{1}{2}n(n-1)$ pairs of items among them. For each pair of items in the lists $(\mathcal{R}_{d_i}, \mathcal{R}_{d_j})$, $NC = NC + 1$ if the pair is ranked in the same order in both lists; otherwise, $ND = ND + 1$.

Figure 3.3 shows an example to illustrate the use of the *Kendall* correlation. In this example, we consider four classifiers c_1, c_2, c_3, and c_4 whose agreement is measured by means of three diversity measures (d_1, d_2, and d_3). Each diversity measure defines three ranked lists ($\mathcal{R}_{d_1}$, $\mathcal{R}_{d_2}$, and $\mathcal{R}_{d_3}$). We highlight in red the differences of $\mathcal{R}_{d_2}$, and $\mathcal{R}_{d_3}$

when compared to $\mathcal{R}_{d_1}$. Note that, in $\mathcal{R}_{d_2}$, just two pairs of classifiers are inverted. Pairs of classifiers in $\mathcal{R}_{d_3}$, in turn, are ranked in the inverse order, when compared to $\mathcal{R}_{d_1}$.

Figure 3.3 also shows in the table on the right side, the τ correlation scores among the three ranked lists. The correlation coefficient value $\tau(\mathcal{R}_{d_1}, \mathcal{R}_{d_2})$, as expected, is high, which means that ranked lists $\mathcal{R}_{d_1}$ and $\mathcal{R}_{d_2}$ have high degree of concordance. However, the correlation between ranked lists $\mathcal{R}_{d_1}$ and $\mathcal{R}_{d_3}$ is low (-1.0 stands for the lowest possible correlation score).

Ranked Lists (R)

d_1		
C_1	C_2	1
C_1	C_3	2
C_1	C_4	3
C_2	C_3	4
C_2	C_4	5
C_3	C_4	6

d_2		
C_1	C_2	1
C_1	C_4	3
C_1	C_3	2
C_2	C_3	4
C_2	C_4	5
C_3	C_4	6

d_3		
C_3	C_4	6
C_2	C_4	5
C_2	C_3	4
C_1	C_4	3
C_1	C_3	2
C_1	C_2	1

Kendall Tau

	d_1	d_2	d_3
d_1	**1.0**	0.8	-1.0
d_2	0.8	**1.0**	-0.8
d_3	-1.0	-0.8	**1.0**

Figure 3.3: Example of three computed ranked lists ($\mathcal{R}_{d_1}$, $\mathcal{R}_{d_2}$, and $\mathcal{R}_{d_3}$) and *Kendall* scores between them. Both ranked lists ($\mathcal{R}$) and *Kendall* are computed by using the validation matrix M_V (see Section 3.2).

Using the Kendall correlation measures for Selecting Classifiers

We propose a novel strategy, named *Kendall* classifier selection (KCS), to define appropriate classifiers to be used in the classification framework presented in [33]. KCS makes use of the degree of agreement of different diversity measures. This agreement is measured in terms of the *Kendall* correlation among ranked lists of classifiers, as presented in Section 3.2.2.

Let d_{H_1} and d_{H_2} be the diversity measures with the highest correlation scores, which are defined by the *Kendall* correlation. Let $\mathcal{R}_{d_{H_1}}$ and $\mathcal{R}_{d_{H_2}}$ be the ranked lists of pairs of classifies defined by d_{H_1} and d_{H_2}, respectively. KCS defines the top-ranked pairs of classifiers in $\mathcal{R}_{d_{H_1}}$ and $\mathcal{R}_{d_{H_2}}$ as the most appropriate ones to be used in the classification framework presented in [33].

We also tested in our experiments selected classifiers defined in terms of the lowest correlated diversity measures (d_{L_1} and d_{L_2}). In this case, we use classifiers defined in the top-ranked positions of $\mathcal{R}_{d_{L_1}}$ and $\mathcal{R}_{d_{L_2}}$.

Figure 3.4 summarizes in six steps the new approach for selecting classifiers based on *Kendall* correlation. It is important to highlight that all steps regarding the selection of classifiers for fusion are performed during the training phase of the decision-making framework. Using a validation set separated during training allows us to evaluate different

descriptors and learning techniques, assess their outcomes when classifying the validation examples, and properly selecting, by means of the proposed *Kendall*-based methodology, the most suitable classifiers for deployment during testing.

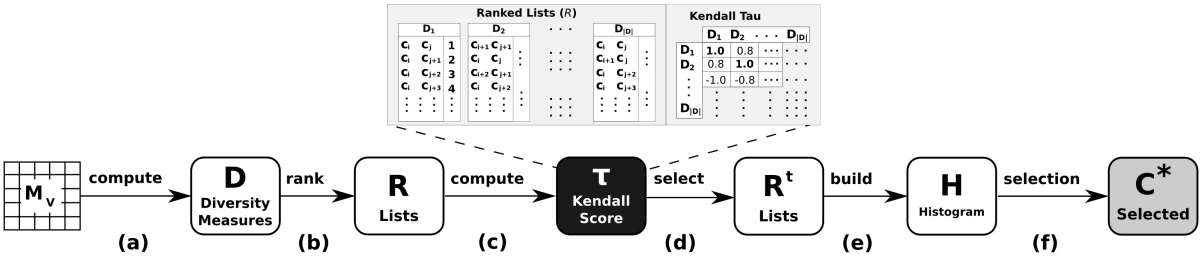


Figure 3.4: The six steps for new classifier selection are: (a) Compute diversity measures from the validation matrix M_V; (b) Sort $\mathcal{R}$ lists by diversity measure scores; (c) Compute *Kendall* correlation coefficients among all ranked lists of classifiers $\mathcal{R}$; (d) Select $\mathcal{R}_{d_{H_1}}$ and $\mathcal{R}_{d_{H_2}}$ or $\mathcal{R}_{d_{L_1}}$ and $\mathcal{R}_{d_{L_2}}$ ranked lists to be used in the next step; (e) $\mathcal{R}^t$ lists with top $t = 100$; (f) Compute a histogram $\mathcal{H}$ that counts the number of occurrences of each classifier; (g) Select the most appropriate classifiers $|\mathcal{C}^*|$ based on their occurrence in $\mathcal{H}$ and a defined threshold $\mathcal{T}$.

Algorithm 2 outlines the proposed steps for selecting classifiers, by taking into account diversity measures selected through *Kendall* correlation. Lines 1-12 refer to the use of diversity measures for defining ranked lists containing pairs of classifiers (arrows (a) and (b) of Figure 3.4). Lines 13-15 refer to the calculation of correlation coefficients as well as the selection of the most and less correlated diversity measures (arrow (c)). Next, the top-ranked pairs of classifiers are selected in Line 16 (arrow (d)) and the number of occurrences of each classifier is determined in Lines 17-25 (arrow (e)). Finally, the most suitable classifiers to be used in the fusion step are defined in Line 26 (arrow (f)).

3.2.3 Selection based on Rank Aggregation

We propose to use multiple diversity and evaluation measures (Kappa, Tau, and accuracy) to determine which classifiers should be combined to improve the classification results in a given problem. Recall that a classifier in this work is the tuple composed of a learning technique and an image descriptor. Our hypothesis is that by exploring complementary information provided by different measures, more appropriate classifiers can be selected to be combined.

As previously discussed in Section 2.4, a diversity measure indicates the agreement of pairs of classifiers. Different diversity measures would rank pairs of classifiers differently. In many situations, *rank aggregation* methods have been used as a way of obtaining a consensus ranking when multiple ranked lists are computed by different approaches.

Algorithm 2 *Selection of classifiers based on Kendall correlation*

Input: set $\mathcal{D}$ of diversity measures, set $\mathcal{C}$ of classifiers, and the outcomes of classifiers on validation set V encoded in M_V.

$\mathcal{R} \leftarrow \emptyset$
$\mathcal{R}_{aux} \leftarrow \emptyset$
$\mathcal{D}_{aux} \leftarrow \emptyset$
for each $d_\ell \in \mathcal{D}$ **do**
  $\mathcal{R}_{d_l} \leftarrow \emptyset$
  for each pair $(c_i, c_j) \in \mathcal{C} \times \mathcal{C}$ **do**
    $score_{d_l}(c_i, c_j) \leftarrow d_l(c_i, c_j)$
    $\mathcal{R}_{d_l} \leftarrow \mathcal{R}_{d_l} \cup \{((c_i, c_j), score_{d_l}(c_i, c_j))\}$
  end for
  Sort $\mathcal{R}_{d_l}$ with regard to $score_{d_l}$
  $\mathcal{R}_{aux} \leftarrow \mathcal{R}_{aux} \cup \mathcal{R}_{d_l}$
end for
Compute *Kendall* correlation coefficients from set $\mathcal{R}_{aux}$
$\mathcal{R} \leftarrow$ select the ranked lists $\mathcal{R}_{d_{H_1}}$, $\mathcal{R}_{d_{H_2}}$, $\mathcal{R}_{d_{L_1}}$, and $\mathcal{R}_{d_{L_2}} \in \mathcal{R}_{aux}$
$\mathcal{D}_{aux} \leftarrow$ select the diversity measures d_{H_1}, d_{H_2}, d_{L_1}, and $d_{L_2} \in \mathcal{D}$
$\mathcal{R}^t \leftarrow$ select the top t ranked pairs of classifiers for each ranked list in $\mathcal{R}$
for each $c_j \in \mathcal{C}$ **do**
  $\mathcal{H}(c_j) \leftarrow 0$
end for
for each $d_\ell \in \mathcal{D}_{aux}$ **do**
  for each $((c_i, c_j), score_{d_l}(c_i, c_j)) \in \mathcal{R}^t_{d_\ell}$ **do**
    $\mathcal{H}(c_i)_{++}$
    $\mathcal{H}(c_j)_{++}$
  end for
end for
$\mathcal{C}^* \leftarrow \{c_i \in \mathcal{C}$, such as $|\mathcal{C}^*| = h$ and $\forall c_j \in \mathcal{C} \setminus \mathcal{C}^*$, $\mathcal{H}(c_i) > \mathcal{H}(c_j), Accuracy(c_j) > \mathcal{T}, Accuracy(c_i) > \mathcal{T}\}$

Rank aggregation has also been treated as the task of combining different ranked lists (or scores) in order to obtain a single, and more accurate, ranked list. For classification tasks, the combination with the lowest error occurs when the classifiers being combined are non-correlated (high diversity) and yields high accuracy rate [17].

In our approach, each considered measure (both diversity and evaluation measures) produces a ranked list of pairs of classifiers. A rank aggregation method combines all ranked lists, producing a single combined ranked list, which is used to identify pairs of classifiers with good classification performance and high diversity. In the next section, we formally define the proposed rank aggregation approach.

Combination of Diversity Measures

Let $\mathcal{R}_{d_l} = \{(c_i, c_j),\ score_{d_l}(c_i, c_j)\}$ be a ranked list of pairs of classifiers defined by the score of the diversity measure $d_l \in \mathcal{D}$ of pairs of classifiers (c_i, c_j). Consider that low values of $score_{d_l}(c_i, c_j)$ indicate high diversity between the pair (c_i, c_j), and therefore, the most suitable pairs of classifiers to be combined are at the top positions of ranked list $\mathcal{R}_{d_l}$. The case of diversity measure for which high score values indicate high diversity, we use the inverse of this measure (e.g., $\frac{1}{DM}$, where DM stands for the Disagreement Measure defined in Equation 2.14).

Let $\mathcal{R}_d = \{\mathcal{R}_{d_1}, \mathcal{R}_{d_2} \ldots \mathcal{R}_{d_{|\mathcal{D}|}}\}$ be the set of ranked lists defined for each available diversity measure. Our objective is to compute a ranked list $\mathcal{R}_c$ that combines all ranked lists $\mathcal{R}_{d_l} \in \mathcal{R}$. We use a multiplication approach [81] for combining the scores of different diversity measures on a single score as follows:

$$score_{d_c}(c_i, c_j) = \prod_{l=1}^{|\mathcal{D}|}(1 + score_{d_l}(c_i, c_j)) \tag{3.2}$$

The ranked list combining the diversity measures is defined as $\mathcal{R}_{d_c} = \{(c_i, c_j),\ score_{d_c}(c_i, c_j)\}$. This ranked list is defined according to the diversity scores.

Evaluation Measure Combination

Let E be a set of evaluation measures. Each evaluation measure $e_\ell \in E$ is used to compute the evaluation of a classifier $c_i \in \mathcal{C}$, based on the validation set. Let $e_\ell(c_i)$ be the evaluation measure for the classifier c_i, an evaluation score for a pair of classifiers can be computed as follows:

$$score_{e_\ell}(c_i, c_j) = (1 + e_\ell(c_i)) \times (1 + e_\ell(c_j)) \tag{3.3}$$

Similarly to diversity measures, we can define a ranked list of pair of classifiers according to each diversity measures. Let $\mathcal{R}_{e_\ell} = \{(c_i, c_j),\ score_{e_\ell}(c_i, c_j)\}$ be a ranked list of pairs of classifiers defined by the score of the evaluation measure $e_\ell \in E$ of pairs of classifiers (c_i, c_j), where high values of $score_{e_\ell}(c_i, c_j)$ indicate that the pair of classifiers (c_i, c_j) is well evaluated.

Let $\mathcal{R}_e = \{\mathcal{R}_{e_1}, \mathcal{R}_{e_2} \ldots \mathcal{R}_{e_{|E|}}\}$ be the set of ranked lists defined for each available evaluation measure. We aim at computing a single ranked list $\mathcal{R}_e$ which combines all ranked lists $\mathcal{R}_{d_l} \in \mathcal{R}$. A multiplication approach similar to the one used for diversity measures is also used for combining evaluation measures. However, as the ranked lists are sorted in ascending order, we multiply the inverse of evaluation scores as follows:

$$score_{e_c}(c_i, c_j) = \prod_{l=1}^{|E|} \frac{1}{score_{e_\ell}(c_i, c_j)} \tag{3.4}$$

The ranked list combining all evaluation measures is defined as $\mathcal{R}_{e_c} = \{(c_i, c_j), score_{e_c}(c_i, c_j)\}$, and is computed using the $score_{e_c}$ score.

Final Ranking of Pairs of Classifiers

As previously stated, our objective is to compute a ranked list that sorts pairs of classifiers with high diversity and high evaluation at top positions. In this way, we compute a final ranked list combining scores of diversity and evaluation measures, as follows:

$$score_c(c_i, c_j) = score_{e_c}(c_i, c_j) \times score_{d_c}(c_i, c_j) \tag{3.5}$$

The final ranked list computed by the rank aggregation approach is defined according to these scores, as $\mathcal{R}_c = \{(c_i, c_j), score_c(c_i, c_j)\}$.

Figure 3.5 summarizes the new six-step approach for selecting classifiers based on rank aggregation.

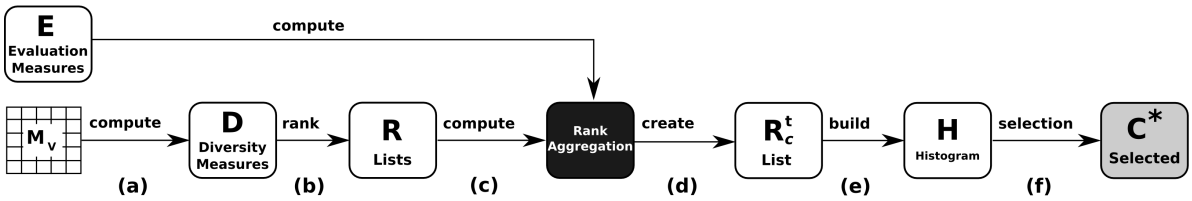


Figure 3.5: The six steps of the new classifier selection are: (a) Compute diversity measures from the validation matrix M_V; (b) Sort $\mathcal{R}$ lists according to scores of diversity measures; (c) Compute rank aggregation using all ranked lists of classifiers ($\mathcal{R}$) and evaluation measures (E); (d) Create a single list $\mathcal{R}_c^t$, which list has the top $t = 100$; (e) Compute a histogram $\mathcal{H}$ that counts the number of occurrences of each classifier; (f) Select the most appropriate classifiers $|\mathcal{C}^*|$ that satisfy a defined threshold $\mathcal{T}$.

Algorithm 3 outlines the proposed steps for selecting classifiers through rank aggregation approaches. Lines 1-10 refer to the use of diversity measures for defining ranked lists containing pairs of classifiers (arrows (a) and (b) of Figure 3.5). Line 11 refers to the use of rank aggregation approaches that combine different diversity and evaluation measures to create a final ranked list (arrow (c)). Next, the top-ranked pairs of classifiers are selected in Line 12 (arrow (d)) and the number of occurrences of each classifier is determined in Lines 13-19 (arrow (e)). Finally, the most suitable classifiers to be used in the fusion step are defined in Line 20 (arrow (f)).

Figure 3.6 illustrates the use of the proposed rank aggregation approach. In this example, we consider four classifiers c_1, c_2, c_3, and c_4 whose agreement is measured by means of three diversity measures (d_1, d_2, and d_3). Each diversity measure defines three

Algorithm 3 *Selection of classifiers based on rank aggregation approaches*

Input: set $\mathcal{C}$ of classifiers, set $\mathcal{D}$ of diversity measures, set E of evaluation measures, and the outcomes of classifiers on validation set V encoded in M_V.

$\mathcal{R} \leftarrow \emptyset$
for each $d_\ell \in \mathcal{D}$ **do**
  $\mathcal{R}_{d_l} \leftarrow \emptyset$
  for each pair $(c_i, c_j) \in \mathcal{C} \times \mathcal{C}$ **do**
    $score_{d_l}(c_i, c_j) \leftarrow d_l(c_i, c_j)$
    $\mathcal{R}_{d_l} \leftarrow \mathcal{R}_{d_l} \cup \{((c_i, c_j), score_{d_l}(c_i, c_j))\}$
  end for
  Sort $\mathcal{R}_{d_l}$ with regard to $score_{d_l}$
  $\mathcal{R} \leftarrow \mathcal{R} \cup \mathcal{R}_{d_l}$
end for
$\mathcal{R}_c \leftarrow$ Create ranked list through rank aggregation approach using $\mathcal{R}$ and E
$\mathcal{R}_c^t \leftarrow$ select the top t ranked pairs of classifiers from ranked list $\mathcal{R}_c$
for each $c_j \in \mathcal{C}$ **do**
  $\mathcal{H}(c_j) \leftarrow 0$
end for
for each $((c_i, c_j), score_{d_l}(c_i, c_j)) \in \mathcal{R}_c^t$ **do**
  $\mathcal{H}(c_i)_{++}$
  $\mathcal{H}(c_j)_{++}$
end for
$\mathcal{C}^* \leftarrow \{c_i \in \mathcal{C}$, such as $|\mathcal{C}^*| = h$ and $\forall c_j \in \mathcal{C} \setminus \mathcal{C}^*$, $\mathcal{H}(c_i) > \mathcal{H}(c_j), Accuracy(c_j) > \mathcal{T}, Accuracy(c_i) > \mathcal{T}\}$

ranked lists ($\mathcal{R}_{d_1}$, $\mathcal{R}_{d_2}$, and $\mathcal{R}_{d_3}$). Furthermore, we consider three evaluation measures (accuracy, kappa, and tau indices) for each classifier whose measures have been computed from validation set (V).

3.3 Meta-learning Approach

Originally proposed by Wolpert [110], meta-learning approach or also called stacked generalization (stacking) is a approach that uses different classifier outcomes (e.g., labels and scores) as input to other learning method (meta learner). Generally, stacking approaches have a more representation of solutions space than no single learning method might learn effectively. Since there is no good simple learning method which achieves good results for any task (No free lunch theorem).

In stacking systems, there is a premisse that instances are considered similar if they are correctly/incorrectly classified to the same class by the same set of classifiers [54].

Figure 3.7 shows an example of matrix M_V, with three different instances I_1, I_2, and

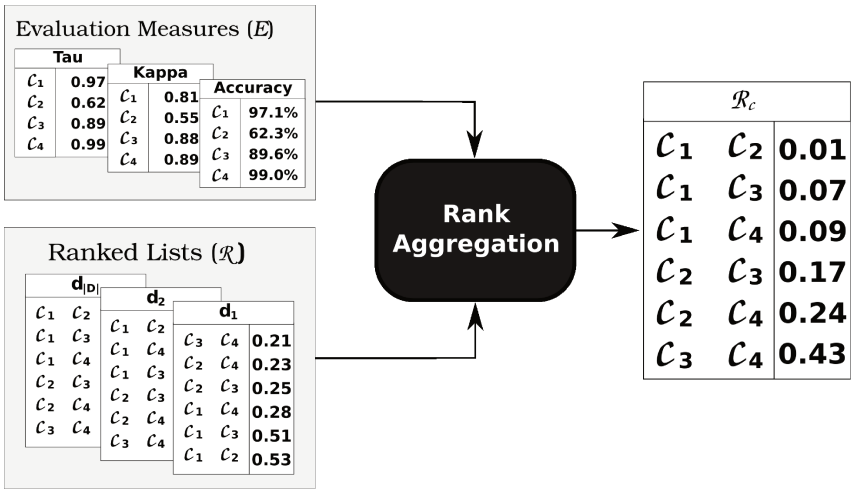


Figure 3.6: Rank aggregation approach for combining ranked lists of classifiers defined by both diversity and evaluation measures.

I_3. The first two belong to the same class (square) and the last instance to the circle class. Also, there are different classifier outcomes ($\mathcal{C}_1, ...\mathcal{C}_{|\mathcal{C}|}$) and ground truth (GT) label associated with each instance. Notice that instances from the same class have similar classifier outcomes. On other hand, instances of different class have different classifier outcomes.

An easier way to see this premisse is when a distance (e.g, the Hamming [42], which counts 0 for equal attributes and 1 for different) has been computed between instances (I_1, I_2, and I_3). We might compute the Hamming Distance using the labed data and show that $I_1 \times I_2$, $I_1 \times I_3$, and $I_2 \times I_3$ have Hamming scores equal to 1, 3, and 4, respectively.

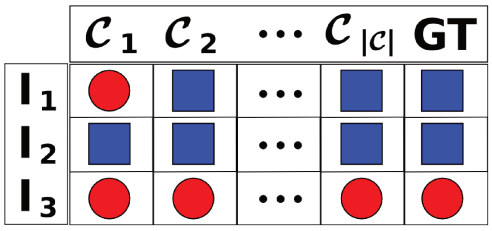


Figure 3.7: Example of matrix M_V.

In the proposed framework (Figure 3.1), the meta learner (SVM) combines $\mathcal{C}^* \subset \mathcal{C}$ classifiers, where $\mathcal{C}^*$ is set of classifiers selected through any selection approach previously described in this chapter.

3.4 Cost-effectiveness Analysis

Our framework may be considered as a highly-parallelizable approach, because there are several steps that could be optimized and which are not addressed in this thesis.

We use a set of independent classifiers, which can be used in different processes/threads in a computer cluster environment or GPU to later form the M_V matrix. The same approach can be used when testing a new image (I). The selection process itself has low cost to be calculated, once diversity measures use only the M_V matrix previously computed for this purpose (Figure 3.2). Once this process has been paralleled, to find an optimal set of classifiers ($\mathcal{C}^*$) for a target problem involves only a few set of operations involving lists as described in the previous section. Finally, the classifier fusion process is dependent on the learning method used, but there are some learning methods in the literature that also use parallelism, as LibSVM library [14].

Our framework can be considered a flexible approach given that it can use different descriptors, set of classifiers and even fusion techniques. Also, techniques such as AdaBoost and Bagging may be used as complementary techniques to our framework in the classifier fusion process. That said, we believe this work opens several research possibilities for researchers to explore innovative ways for combining learning methods as well as for exploring more efficient ways of performing such fusion.

Chapter 4

Experimental Methodology

In this section, we present the experimental methodology adopted in each experiment of this thesis.

4.1 Datasets

In this thesis, we perform experiments on five different datasets, which are related to five different real applications: the Caltech101 dataset has been used in object recognition problems (Section 2.6.1); the Fruits dataset has been used in produce recognition tasks (Section 2.6.2); Two Remote Sensing Image (RSI) datasets are related to two different real tasks (Section 2.6.3): the Coffee dataset has been used for coffee crop recognition tasks; the Urban dataset has been used for urban recognition. Finally, the Freefoto dataset has been used for natural scene classification (Section 2.6.4).

Table 4.1 shows the datasets used in each experiment performed in this thesis.

Table 4.1: Five datasets used in this thesis.

Experiment	**Dataset**				
	Caltech	Coffee	Freefoto	Fruits	Urban
Framework for Classifier Fusion (Section 5.1)	X	X	X	X	
Correlation Analysis between Diversity Measures (Section 5.2)	X	X	X	X	X
Classifier Selection Approaches (Section 5.3)	X				X

4.2 Image Descriptors

Table 4.2, 4.3, and 4.4 present the color, texture, and shape descriptors that were used in our experiments. Given the classification problem, the objective is to use the most complementary features as possible and rely on an effective combination technique.

Table 4.2: Four datasets and seven image descriptors used in our experiments.

Framework for Classifier Fusion (Section 5.1)					
Type	**Descriptor**	**Dataset**			
		Caltech	Coffee	Freefoto	Fruits
Color	ACC [50]	X	X	X	X
	BIC [96]	X	X	X	X
	CCV [78]	X	X	X	X
	GCH [101]	X	X	X	X
Texture	LAS [103]	X		X	X
	QCCH [49]	X	X	X	X
	SID [116]		X		
	UNSER [106]		X		
Shape	EOAC [68]	X		X	X

Table 4.3: Five datasets and six image descriptors used in these experiments.

Correlation Analysis between Diversity Measures (Section 5.2)						
Type	**Descriptor**	**Dataset**				
		Caltech	Coffee	Freefoto	Fruits	Urban
Color	BIC [96]	X	X	X	X	X
	CCV [78]	X	X	X	X	X
	GCH [101]	X	X	X	X	X
Texture	LAS [103]	X		X	X	
	QCCH [49]	X	X	X	X	X
	SID [116]		X			X
	UNSER [106]		X			X
Shape	EOAC [68]	X		X	X	

Table 4.4: Two datasets and six image descriptors used in these experiments.

Classifier Selection Approaches (Section 5.3)			
Type	**Descriptor**	**Dataset**	
		Coffee	Urban
Color	BIC [96]	X	X
	CCV [78]	X	X
	GCH [101]	X	X
Texture	QCCH [49]	X	X
	SID [116]	X	X
	UNSER [106]	X	X

In the experiments with remote sensing images (Coffee and Urban), we have used a different set of bands. Agricultural specialists usually perform analysis of agricultural targets by exploiting vegetation indices, such as NDVI [73]. With those indices, it is possible to estimate production and differentiate some objects in the surface. Thus, in this work, the feature extraction algorithms are performed mainly on the bands corresponding to Red (R), Green (G) and Near-Infrared (NI). These bands are the most interesting for agricultural targets since are the basis for the computation of the main vegetation indices.

4.3 Learning Methods and Baselines

Choosing the most appropriate learning method is also a non-trivial problem. Therefore, our approach evaluates possible candidates and selects the most appropriate ones. In this context, we have used seven learning methods in our framework: Decision Tree (DT), Naïve Bayes (NB), Naïve Bayes Tree (NBT), Simple Logistic (SL), k-Nearest Neighbors (kNN), using $k = 1$, $k = 3$, and $k = 5$. Such methods are simple and fast, being suitable to be combined in a real-time recognition system. In this sense, a support vector machine was avoided here due to its known slow training time. Even though SVMs have sub linear time for testing, in a multi-class scenario, it would need several binary SVMs to perform the multi-class classification as reported in [4, 79].

The proposed framework aims at automatically finding suitable combinations of classifiers formed by descriptors and learning methods. We have used the implementation of those learning methods available in the WEKA (version 3-6-2) data mining library. All learning methods were used with default parameters which means we did not optimize them whatsoever.

As baselines in this work, we considered all of the seven learning methods and eight different ensemble approaches: BAGG-DEFAULT, BAGG-36, BAGG-49, BOOST-DEFAULT, BOOST-36, BOOST-49, MV-36, and MV-49. BOOST-DEFAULT, BOOST-36, and BOOST-36 implement one multi-class Adaboost approach, but they are different depending on the number of iterations in their algorithm. BOOST-DEFAULT uses default parameters, while BOOST-36 and BOOST-49 employ 36 and 49 iterations for tunning the learning parameters, respectively. BAGG-36 and BAGG-49 in turn, relies on the Bagging approach with 36 and 49 classifiers, respectively. The number after a baseline name (e.g., 36 as in MV-36) refers to the number of classifiers that are considered.

Table 4.5: Learning methods used in our experiments with their default parameters.

Type	Learning Method	Acronym
Simple	weka.classifiers.trees.J48 -C 0.25 -M 2	DT
	weka.classifiers.lazy.IBk -K 1	kNN1
	weka.classifiers.lazy.IBk -K 3	kNN3
	weka.classifiers.lazy.IBk -K 5	kNN5
	weka.classifiers.bayes.NaiveBayes	NB
	weka.classifiers.trees.NBTree	NBT
	weka.classifiers.functions.SimpleLogistic	SL
Ensemble	weka.classifiers.meta.AdaBoostM1 -I 10	BOOST-DEFAULT
	weka.classifiers.meta.AdaBoostM1 -I 36	BOOST-36
	weka.classifiers.meta.AdaBoostM1 -I 49	BOOST-49
	weka.classifiers.meta.Bagging -I 10	BAGG-DEFAULT
	weka.classifiers.meta.Bagging -I 36	BAGG-36
	weka.classifiers.meta.Bagging -I 49	BAGG-49

Table 4.6: Simple learning methods combined by our meta-learning technique for each experiment.

Experiment	Learning Method						
	DT	kNN1	kNN3	kNN5	NB	NBT	SL
Framework for Classifier Fusion (Section 5.1)	X	X	X	X	X	X	X
Correlation Analysis between Diversity Measures (Section 5.2)	X	X	X	X	X	X	
Classifier Selection Approaches (Section 5.3)	X	X	X	X	X	X	

4.4 Evaluation Measures

To report the effectiveness of each method, in all of our experiments, we have used evaluation measures based on the confusion matrix: Accuracy, Kappa [10], and Tau [67] indices. Given a confusion matrix as Table 4.7 shows, the measures can be calculated according to Equations 4.1–4.6.

Table 4.7: Confusion Matrix. TP, TN, FP, and FN stand for true positive, true negative, false positive, and false negative, respectively.

		Predicted	
		Class A	Class B
Real	Class A	TP	FP
	Class B	FN	TN

$$Total = TP + FP + FN + TN. \tag{4.1}$$

$$pe = \frac{((TP + FP) \times (TP + FN)) + ((FN + TN) \times (FP + TN))}{Total^2} \tag{4.2}$$

$$pr = \frac{((TP + FP) \times TP) + ((FN + TN) \times TN)}{Total^2} \tag{4.3}$$

$$\textbf{Accuracy} = \frac{TP + TN}{Total} \tag{4.4}$$

$$\textbf{Kappa} = \frac{accuracy - pe}{1 - pe} \tag{4.5}$$

$$\textbf{Tau} = \frac{accuracy - pr}{1 - pr} \tag{4.6}$$

4.5 Validation Protocol

We use the k-fold cross-validation protocol. According to this protocol, given a dataset comprising several examples, we randomly separate it into k subsets, with no repetition. A subset is chosen as testing set, and the remaining $k - 1$ subsets are used for training a learning method. The cross-validation process is repeated k times (called rounds) and each subset is used only once as test set. The final result (the accuracy in classification tasks) from this process is the arithmetic mean of all rounds. In our experiments, we considered a 5-fold cross-validation protocol in which, in each round, we use three folds

for training the classifiers, one for validating the classifiers and for calculating diversity measures, training convergence, etc, and one as the actual testing set. At each round, we switch the training, validation, and test sets.

Chapter 5

Experimental Results

This chapter presents and discusses experimental results. Section 5.1 presents the experiments that we performed to evaluate the robustness of our fusion framework with no selection process. Section 5.2 discusses a correlation analysis between five diversity measures for different datasets. Section 5.3 presents a comparative study between the three classifier selection strategies proposed in this thesis for two different remote sensing image dataset.

5.1 Framework for Classifier Fusion

This section discusses the results regarding the effectiveness and efficiency of the proposed framework using four different datasets (Caltech, Coffee, Freefoto, and Fruits). In Section 5.1.1, we present an effectiveness comparison between our proposed framework and different baselines of the literature. In Section 5.1.2, we also show tests of statistical significance and confidence interval for all involved techniques. Finally, in Section 5.1.3, we show the behavioral analysis of our proposed framework and other techniques in a small training set scenario.

5.1.1 Effectiveness Analysis

In these experiments, five fusion techniques were compared: our approach using SVM (FSVM-KERNEL-49) considering $|C| = 49$, two AdaBoost approaches (BOOST-DEFAULT and BOOST-49), Bagging (BAGG-49), and Majority Voting (MV-49).

Recall that using $|C| = 49$ means that all available classifiers (7 learning methods $\times$ 7 image descriptors) are employed in the fusion process. KERNEL can be two different SVM kernels: PK stands for the polynomial kernel while NORM stands for the normalized

polynomial kernel. Furthermore, we have included the best single classifier (no fusion) between all tested learning methods.

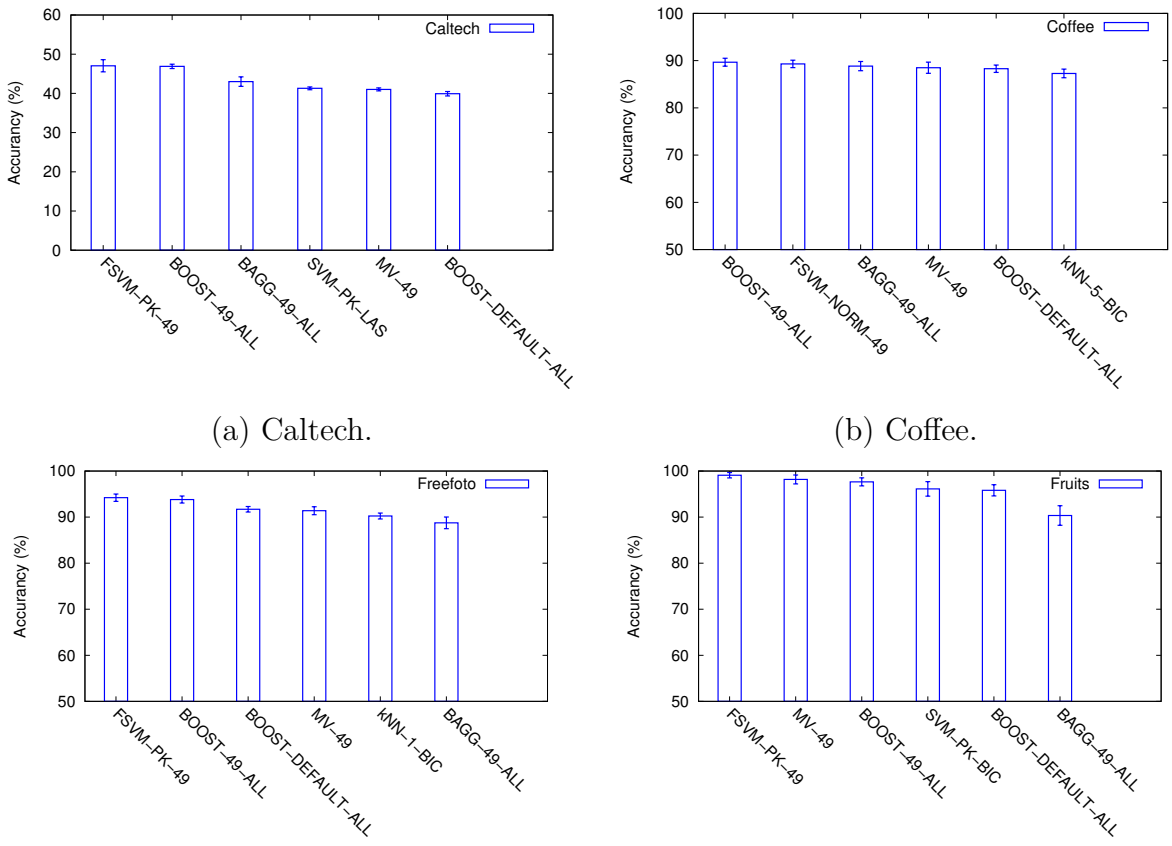
Table 5.1 presents the results obtained for each fusion technique and the best single classifier using four datasets and considering three different evaluation measures (Accuracy, Kappa, and Tau). Notice that BOOST and BAGG techniques show up with the suffix ALL, which means the concatenation of the feature vectors produced by the seven different image descriptors considered. Thus BAGG-49-ALL and BOOST-49-ALL techniques refer to the use of 49 iterations and seven image descriptors.

Table 5.1: Classification effectiveness of the proposed framework and baselines, with their respective standard deviations.

Datasets	**Techniques**	**Measures**		
		Accuracy	**Kappa**	**TAU**
Caltech	**FSVM-PK-49**	**47.05%±1.77**	**0.45±0.02**	**0.46±0.02**
	BOOST-49-ALL	46.90%±0.63	0.45±0.01	0.46±0.01
	BAGG-49-ALL	43.01%±1.38	0.41±0.01	0.42±0.01
	SVM-PK-LAS	41.30%±0.41	0.39±0.00	0.40±0.00
	MV-49	41.02%±0.46	0.38±0.00	0.40±0.00
	BOOST-DEFAULT-ALL	39.92%±0.57	0.38±0.01	0.39±0.01
Coffee	BOOST-49-ALL	89.66%±0.84	0.65±0.02	0.72±0.01
	FSVM-NORM-49	**89.31%±0.91**	**0.63±0.01**	**0.71±0.01**
	BAGG-49-ALL	88.84%±1.10	0.61±0.04	0.69±0.02
	MV-49	88.50%±1.34	0.59±0.04	0.68±0.03
	BOOST-DEFAULT-ALL	88.29%±0.78	0.62±0.02	0.69±0.01
	kNN-5-BIC	87.29%±1.03	0.59±0.03	0.67±0.02
Freefoto	**FSVM-PK-49**	**94.22%±0.92**	**0.93±0.01**	**0.93±0.01**
	BOOST-49-ALL	93.82%±0.87	0.93±0.01	0.93±0.01
	BOOST-DEFAULT-ALL	91.71%±0.60	0.90±0.01	0.90±0.01
	MV-49	91.39%±0.99	0.90±0.01	0.90±0.01
	kNN-1-BIC	90.24%±0.73	0.88±0.01	0.89±0.01
	BAGG-49-ALL	88.76%±1.45	0.87±0.02	0.87±0.02
Fruits	**FSVM-PK-49**	**99.09%±0.66**	**0.99±0.01**	**0.99±0.01**
	MV-49	98.18%±1.11	0.98±0.01	0.98±0.01
	BOOST-49-ALL	97.65%±1.01	0.97±0.01	0.97±0.01
	SVM-PK-BIC	96.13%±1.79	0.96±0.02	0.96±0.02
	BOOST-DEFAULT-ALL	95.82%±1.22	0.95±0.01	0.96±0.01
	BAGG-49-ALL	90.35%±2.42	0.90±0.03	0.90±0.03

In these experiments, our fusion approach (FSVM-KERNEL-49), which uses meta-learning on the outputs of all available classifiers yielded a slightly better classification result considering the three evaluation measures, when compared to other techniques in any tested datasets.

Important to note that BOOST and BAGG techniques use a fusion hybrid (feature and decision level fusion) to achieve similar results to our framework that uses only decision level fusion. For a better visualization, Figure 5.1 depicts all results sorted by classification accuracy. Next section, additional experiments show that FSVM-KERNEL-49 has no statistical difference between the best baseline in each dataset used (see Table 5.2 in Section 5.1.2).



(a) Caltech.

(b) Coffee.

(c) Freefoto.

(d) Fruits.

Figure 5.1: The best results using different learning methods in the Caltech, Coffee, Freefoto, and Fruits datasets.

5.1.2 Tests of Statistical Significance

Paired t-test has been performed to verify the statistical significance of the results. We calculated the confidence intervals for the differences among paired means of each class from the datasets, then we have compared our approach against the best baselines for each dataset. In these tests, if the p-value is less than 0.05 (confidence of 95%) there is a

significant difference between a pair of classifiers.

Table 5.2 shows a statistical comparison of our approach against each one of the best baselines, both using 49 classifiers. This comparison shows that there is significant difference among our approach and other baselines only for the Fruits dataset (FSVM-PK-49 × MV-49). In this test, our approach is statistically better than MV-49 using Fruits dataset.

Table 5.2: Significance tests for FSVM-KERNEL-$|C^*|$, where $|C^*|$ is the number of classifiers used by SVM in each test.

Datasets	**Pair of Classifiers**	**t-test**	**Significant**
		p-value	**Difference**
Caltech	FSVM-PK-49 × BOOST-49-ALL	0.0606	-
Coffee	FSVM-NORM-49 × BAGG-49-ALL	0.3330	-
Freefoto	FSVM-PK-49 × BOOST-49-ALL	0.3798	-
Fruits	FSVM-PK-49 × MV-49	0.0164	Yes

In other three datasets (Caltech, Coffee, and Freefoto), the p-value scores have been greater than 0.05 and nothing might be said. Therefore, we also computed the confidence intervals related to the results of all techniques. Figure 5.2 shows a comparison between involved techniques.

Notice that in the experiments with Caltech and Freefoto datasets, our framework (late fusion) achives similar results to BOOST-49-ALL (hybrid fusion). Importantly to say that hybrid fusion techniques might suffer from two major problems of machine learning area which our approach does not apply such as “curse of dimensionality” and normalization feature.

5.1.3 Training Set Size Impact

This section shows a behavioral study among the classifiers compared in Table 5.1 using reduced training sets. In our experiments, we conducted a study considering five different sizes for the training set (T): $8\%, 16\%, 33\%, 67\%, 100\%$, which represents 5%, 10%, 20%, 40% and 60% of the entire datasets, respectively. These subsets have been selected from original training set. We use again the 5-fold cross-validation protocol previously adopted in our experiments.

Figure 5.3 shows the results for all four datasets (Caltech, Coffee, Freefoto, and Fruits) used in our work. The x-axis represents the number of images in the training set and the y-axis represents the average accuracy in the testing set.

Figure 5.3(a) shows experiments using the Caltech dataset. The FSVM-PK-49 approach using a subset of 8% of training set achieves 39.52% of accuracy. In the same training set, BOOST-49-ALL yields 32.33%, which means that our approaches have a

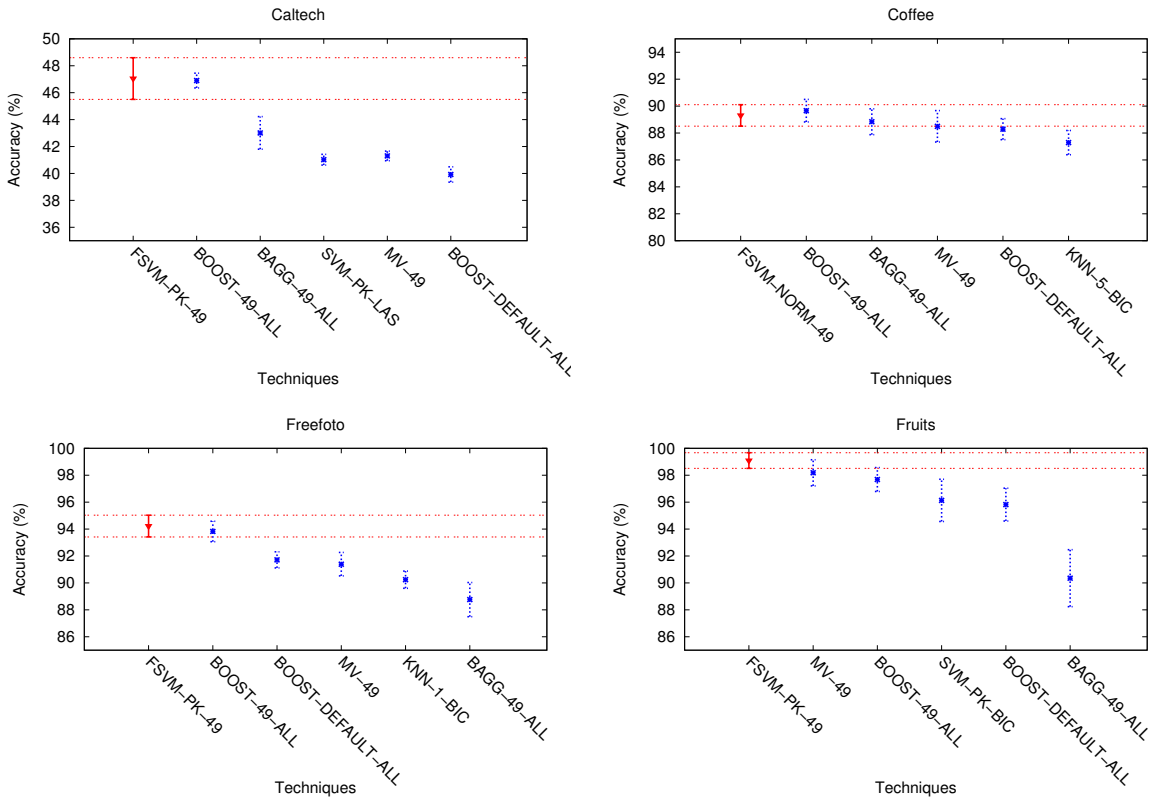
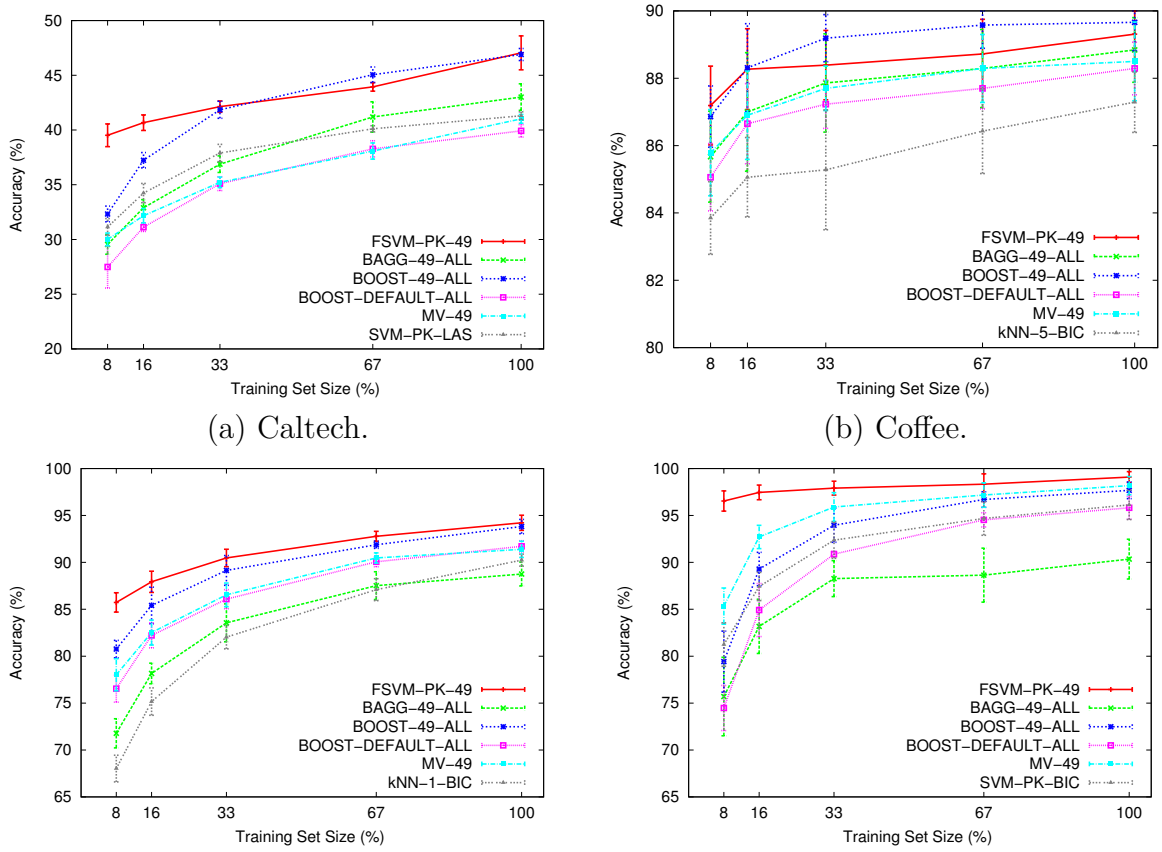


Figure 5.2: Confidence intervals for all techniques compared in Table 5.1.

gain of more than 19% compared to the best baseline. In the subset 16%, our approaches are still better and achieve accuracy results of 40.67% (FSVM-PK-49) against 37.24% of the BOOST-49-ALL. That represents a gain of more than 7% in classification accuracy. From the subset 33% to 100%, the best baseline yields similar performance to our approach. In summary, we can see that the proposed approach are able to learn from small training sets.

Figure 5.3(b) shows experiments using the Coffee dataset. In these experiments, we could note that there is no difference among our approach and the baselines. This phenomenon is observed only on this dataset that is a binary dataset (two classes). We hypothesize that characteristics of each dataset (e.g., number of image and/or classes) might be relevant factors for the baselines, including simple classifiers.

Figure 5.3(c) shows experiments using the Freefoto dataset. In these experiments, the results of the FSVM-PK-49 approach results are similar to those observed in the



(a) Caltech. (b) Coffee.

(c) Freefoto. (d) Fruits.

Figure 5.3: Accuracy scores of all classifiers using training sets with different sizes.

experiments using the Caltech dataset. The FSVM-PK-49 approach using a subset of 8% of training set achieves 85.73% of accuracy. In the same training set, BOOST-49-ALL achieves 80.76%, which means that the FSVM-PK-49 approach produces gains of 6.15 percentage, when compared to the best baseline (BOOST-49-ALL).

Figure 5.3(d) shows experiments using the Fruits dataset. We can notice the same behavior observed for the Caltech dataset, in which the our approach yields better results than the best baseline (MV-49) using reduced training set sizes. The FSVM-PK-49 approach using 8% of the training set achieves 96.54% of accuracy. Using the same training set, MV-49 achieves a testing result of 79.42% classification accuracy. Therefore, FSVM-PK-49 is 13.12 percentage points more accurate than MV-49. Similar behavior can be observed for the subset 16%. In this case, the proposed methods achieve accuracy results of 97.46% (FSVM-PK-49) against 92.71% of the MV-49.

5.2 Correlation Analysis between Diversity Measures

This section shows that the use of different diversity measures may potentially improve the quality of selected classifiers. We exemplify this fact by performing a correlation analysis among the ranked lists defined by diversity measures. We use the well-known Kendall tau [56] measure in this analysis.

Table 5.3 shows the Kendall tau correlation scores for all of diversity measures described in Section 2.4 for five different datasets (Caltech, Coffee, Freefoto, Fruits, and Urban).

As we can observe, the measures COR, DM, and QSTAT have high correlation coefficients between them for all five datasets used. The same behaviour does not show up with the lowest correlation coefficient. In experiments with the Caltech dataset, the measures DFM and IA have the lowest correlation coefficient, 0.03, which is a high value if comparable to the lowest correlation coefficients in other three datasets. This means that all measures have many similar opinions between them.

Performed experiments with Coffee and Urban datasets show that the measure IA has the lowest correlation coefficients for all other measures and datasets analyzed which means it is a very good candidate to be considered when selecting diversity measures. In experiments with Freefoto and Fruits datasets, the measure DFM has the lowest correlation coefficients. Notice that none of the used measures is highly non-correlated with each other. This means that, although they are different diversity measures, all of them have an agreement degree about which classifiers should be combined.

5.3 Classifier Selection Approaches

In this section, we aim at comparing the performance of the three selection approaches on the framework of selection and fusion with respect to the best baselines of the literature. This performance evaluation considers both effectiveness and efficiency aspects. Effectiveness analysis is based on accuracy, kappa, and tau results in the Coffee and Urban datasets. Efficiency analysis, in turn, is based on the number of classifiers on the framework to achieve the same effectiveness results than the best baselines. In Section 5.3.1, we present effectiveness comparison between three classifier selection strategies. Once found the best selection strategy, in Section 5.3.2, we present more details of the selection process. In Section 5.3.3, we maintained the same selection strategy to compare different kinds of fusion technique (early, late, and hybrid). Finally, in Section 5.3.4, we discuss tests of statistical significance for all late fusion techniques.

Table 5.3: Kendall tau between five diversity measures using four different datasets. In blue, highest correlation coefficients and in red, lowest correlation coefficients.

Caltech					
Diversity Measures	**COR**	**DFM**	**DM**	**IA**	**QSTAT**
COR	1.00	0.11	0.83	0.26	**0.98**
DFM	-	1.00	0.08	**0.03**	0.11
DM	-	-	1.00	0.26	0.85
IA	-	-	-	1.00	0.27
QSTAT	-	-	-	-	1.00
Coffee					
Diversity Measures	**COR**	**DFM**	**DM**	**IA**	**QSTAT**
COR	1.00	0.05	0.87	-0.17	**0.95**
DFM	-	1.00	0.05	-0.16	0.06
DM	-	-	1.00	-0.17	0.88
IA	-	-	-	1.00	**-0.18**
QSTAT	-	-	-	-	1.00
Freefoto					
Diversity Measures	**COR**	**DFM**	**DM**	**IA**	**QSTAT**
COR	1.00	-0.09	0.66	0.18	**0.89**
DFM	-	1.00	-0.06	-0.03	**-0.10**
DM	-	-	1.00	0.14	0.68
IA	-	-	-	1.00	0.18
QSTAT	-	-	-	-	1.00
Fruits					
Diversity Measures	**COR**	**DFM**	**DM**	**IA**	**QSTAT**
COR	1.00	0.00	0.42	0.15	**0.78**
DFM	-	1.00	0.01	**-0.02**	-0.01
DM	-	-	1.00	0.10	0.46
IA	-	-	-	1.00	0.17
QSTAT	-	-	-	-	1.00
Urban					
Diversity Measures	**COR**	**DFM**	**DM**	**IA**	**QSTAT**
COR	1.00	-0.04	0.88	-0.07	**0.96**
DFM	-	1.00	-0.05	0.03	-0.04
DM	-	-	1.00	-0.07	0.89
IA	-	-	-	1.00	**-0.08**
QSTAT	-	-	-	-	1.00

5.3.1 Behavioral Analysis

This section shows four different analysis on the behavior of diversity measures in the classifier selection process using the framework proposed in [33], introduced in Section 3.1. This framework is denoted as FSVM-NORM-$|\mathcal{C}^*|$, where NORM denotes the normalized polynomial SVM kernel used in our experiments and $|\mathcal{C}^*|$ is number of simple classifiers that will be combined by the SVM-based meta-learning technique.

Tables 5.4 and 5.5 show the average kappa indices for all performed experiments with the Coffee and Urban datasets. The columns refer to the number of classifiers $|\mathcal{C}^*|$, which have values range from 5 to 36, where 5 is the lowest number of classifiers selected and 36 is the total amount of possible classifiers that can be selected (six image descriptors and six learning methods result in 36 different simple classifiers). The rows denote the classification effectiveness measured in terms of the kappa index.

In these experiments, we compare four selection strategies: SINGLE, '*ALL*', '*Kendall*', and Rank Aggregation. SINGLE refers to the selection process use only one diversity measure. '*ALL*' refers to the Consensus strategy described in Section 3.2.1, which uses all the five diversity measures in the selection process. '*Kendall*', in turn, refers to the Kendall strategy described in Section 3.2.2, which uses the two less correlated diversity measures (in the case, IA and $QSTAT$) in the selection process. These diversity measures were defined according to an *a priori* correlation analysis (see Table 5.3). Finally, Rank Aggregation refers to the use of the rank aggregation strategy described in Section 3.2.3.

In these experiments, we consider different configurations of the rank aggregation approaches:

1. *Eff-Acc+Kappa+Tau_div-ALL* approach: This configuration uses three different evaluation measures (accuracy, kappa, and tau) and five diversity measures (ALL).

2. *Eff-Acc+Kappa_div-IA+QSTAT* approach: This configuration uses two different evaluation measures (accuracy and kappa) and two diversity measures (IA and $QSTAT$).

3. *Eff-Acc+Kappa+Tau_div-DFM+IA+QSTAT* approach: This configuration uses three evaluation measures (accuracy, kappa, and tau) and three diversity measures (DFM, IA, and $QSTAT$).

4. *Eff-Kappa_div-ALL* approach: This configuration uses a single evaluation measure (kappa) and five diversity measures (ALL).

5. *Eff-Kappa_div-DFM+IA+QSTAT* approach: This configuration uses one evaluation measure (kappa) and three diversity measures (DFM, IA, and $QSTAT$). DFM, IA, and $QSTAST$ were chosen due to their good results in previous experiments reported in Table 5.3.

In the Table 5.4, we can observe effectiveness results for each kind of selection approach (SINGLE, *ALL*, *Kendall*, and Rank Aggregation). The best Kappa indices for each number of classifiers ($|\mathcal{C}^*|$) defined in the selection process are presented in bold. Moreover, we highlight in blue the number minimum of classifier that each approach needs to achieve similar result than the FSVM-NORM-$|\mathcal{C}^*|$ using all classifiers ($|\mathcal{C}^*| = 36$). SINGLE and

ALL approaches need to use $|\mathcal{C}^*| = 15$ classifiers. *Kendall* approach achieves similar result using $|\mathcal{C}^*| = 10$ classifiers. Finally, rank aggregation approach with configuration *Eff-Kappa_div-ALL* is able to yield very effective results with only $|\mathcal{C}^*| = 5$ classifiers.

Similar effectiveness performance were observed for the methods in the Urban dataset (Table 5.5). In this dataset, *ALL* and *Kendall* yield different results from those observed for the Coffee dataset. *ALL* starts with better results than DFM for $|\mathcal{C}^*| \in \{5, \ldots, 20\}$ and has the same good results observed for the COR, $QSTAT$, and DM-based methods. *Kendall* has not obtained the best results for almost all $|\mathcal{C}^*|$ values, except for $|\mathcal{C}^*| = \{15\}$.

In the experiments with the Urban dataset, we also observe that there is no large difference among the results of all SINGLE approaches. Perhaps this high correlation might be the cause of the poor performance of the *Kendall* approaches. Furthermore, we can observe the good results of IA and $QSTAT$ which need $|\mathcal{C}^*| = 10$ classifiers to achieve similar results than FSVM-NORM-36. *ALL* approach needs to use $|\mathcal{C}^*| = 15$ classifiers, *Kendall* approach needs $|\mathcal{C}^*| = 10$ classifiers, and rank aggregation approach with configuration *Eff-Kappa_div-*$DFM+IA+QSTAT$ needs only $|\mathcal{C}^*| = 5$ classifiers.

Notice that there is no selection approach that achieves the best results for any number of classifiers. However, the majority of good results (in bold, Tables 5.4, 5.5) have been achieved for different configurations of our approach based on rank aggregation. In summary, the investigation of optimal combinations of diversity measures has showed to be a promising research venue and rank aggregation approaches showed to be a good solution to address this problem.

5.3.2 Fine-grained Analysis of the Selection Process

In this section, a more detailed analysis of the best classifier selection approach is performed in the previous Section 5.3.1. Rank aggregation approaches using the late fusion described in Section 3.1 achieved better results with $|\mathcal{C}^*| = 5$ classifiers. Eff-Kappa_div-*ALL* and Eff-Kappa_div-$DFM+IA+QSTAT$, which are associated with the results highlighted in blue in Tables 5.4 and 5.5, have been considered for the Coffee and Urban datasets, respectively.

Figures 5.4 and 5.5 show the histograms $\mathcal{H}$ created in the selection process, while Figures 5.6 and 5.7 show the accuracy performances of all simple/non-complex classifiers using the validation set V. We highlight in green bars the 5 classifiers that have been selected by our rank aggregation approaches.

Notice in Figures 5.4 and 5.5 that, although NB classifier has achieved the highest frequency, our selection approach does not choose any NB classifier as candidate for fusion (green bars). This is due to our policy of also considering the individual accuracy performance of classifiers in the selection process. As the accuracy performance of NB

Table 5.4: Kappa indices computed for all diversity measures using the 5-fold cross-validation protocol for different number of classifiers ($|\mathcal{C}^*|$) in the Coffee dataset. Similar effectiveness performances were observed for other evaluation measures (accuracy and Tau index).

Coffee								
Approaches	**Diversity Measures**	**Number of Classifiers $\|\mathcal{C}^*\|$**						
		5	**10**	**15**	**20**	**25**	**30**	**36**
SINGLE	**COR**	0.515	0.562	0.597	0.610	0.618	0.624	0.628
	DFM	0.490	0.540	**0.610**	0.620	0.620	**0.630**	
	DM	0.507	0.549	0.577	0.611	0.622	0.622	
	IA	0.557	0.579	0.601	0.606	0.607	0.613	
	QSTAT	0.515	0.562	0.597	0.610	0.618	0.624	
Consensus	**ALL** [33]	0.472	0.553	**0.592**	0.610	0.628	0.623	0.628
Kendall	**IA+QSTAT (*lowest*)** [34]	0.560	**0.590**	0.594	0.615	0.618	0.617	0.628
	COR+QSTAT (*highest*) [34]	0.497	0.529	0.551	0.610	0.609	0.609	
Rank Aggregation [35]	**Eff-Acc+Kappa+Tau_div-ALL**	0.553	**0.614**	0.616	**0.624**	0.628	0.622	0.628
	Eff-Acc+Kappa+Tau_div-DFM+IA+QSTAT	0.580	0.582	**0.619**	0.615	0.626	0.620	
	Eff-Kappa_div-ALL	**0.586**	0.589	0.597	0.604	**0.634**	0.617	
	Eff-Kappa_div-DFM+IA+QSTAT	0.551	0.584	0.596	0.620	0.629	0.625	

Table 5.5: Kappa indices computed for all diversity measures using the 5-fold cross-validation protocol for different number of classifiers ($|\mathcal{C}^*|$) in the Urban dataset. Similar effectiveness performances were observed for other evaluation measures (accuracy and Tau index).

Urban								
Approaches	**Diversity Measures**	**Number of Classifiers $\|\mathcal{C}^*\|$**						
		5	**10**	**15**	**20**	**25**	**30**	**36**
SINGLE	**COR**	0.575	0.595	0.595	0.597	0.607	0.606	0.612
	DFM	0.527	0.550	0.572	0.604	0.604	0.607	
	DM	0.558	0.581	0.593	0.603	0.608	0.608	
	IA	0.588	**0.596**	0.601	0.604	0.603	0.605	
	QSTAT	0.573	**0.596**	0.595	0.597	0.607	0.606	
Consensus	**ALL** [33]	0.564	0.570	**0.594**	0.604	0.609	0.606	0.612
Kendall	**IA+QSTAT (*lowest*)** [34]	0.566	**0.592**	**0.604**	0.600	0.605	0.604	0.612
	COR+QSTAT (*highest*) [34]	0.575	0.595	0.595	0.597	0.607	0.606	
Rank Aggregation [35]	**Eff-Acc+Kappa+Tau_Div-DFM+IA+QSTAT**	0.591	**0.600**	0.597	0.600	**0.612**	**0.614**	0.612
	Eff-Acc+Kappa_div-IA+QSTAT	0.579	**0.600**	0.593	0.600	0.611	**0.614**	
	Eff-Kappa_div-ALL	0.581	0.588	0.600	**0.607**	**0.612**	0.610	
	Eff-Kappa_div-DFM+IA+QSTAT	**0.593**	0.592	0.602	0.605	0.610	0.609	

is below than the employed threshold values (blue line, 79.88% in the Coffee dataset and 70.97% in the Urban dataset), this classifier is not selected.

Selected classifiers for the Coffee dataset are kNN1-BIC, kNN3-BIC, kNN5-BIC, kNN1-QCCH, and kNN1-UNSER. For the Urban dataset, our approach selects NBT-BIC, DT-CCV, NBT-CCV, kNN3-CCV, and kNN5-CCV. In experiment with the Coffee dataset, three selected classifiers use the BIC descriptor, while in the Urban dataset, four classifiers use CCV descriptor. We can observe, therefore, a huge impact of using BIC and CCV image descriptors for the Coffee and Urban dataset, respectively.

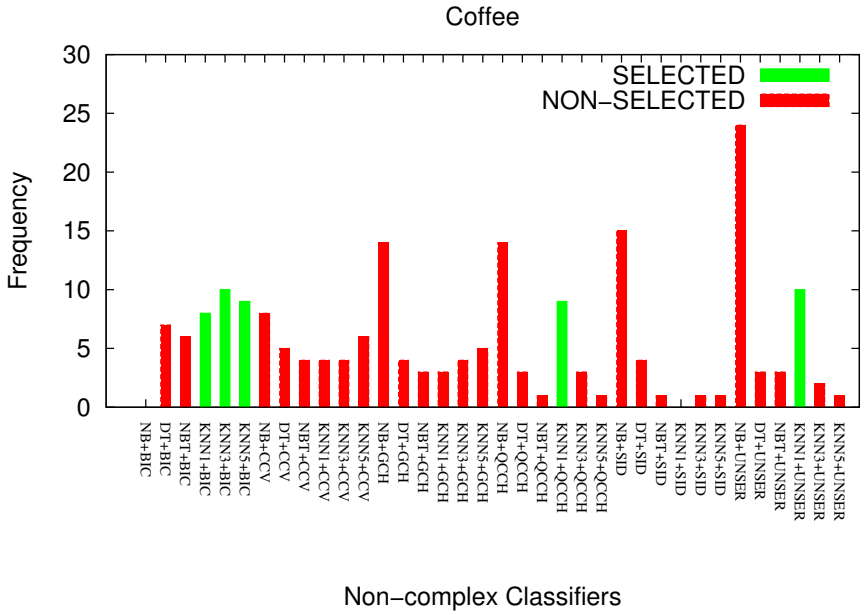


Figure 5.4: Histograms related to the occurrence of classifiers in the selection process for the Coffee dataset.

5.3.3 Effectiveness Analysis considering Early, Late, and Hybrid Fusion

In this section, we evaluate three fusion schemes: Early, Late, and Hybrid.

For the early fusion techniques, we consider the concatenation of the feature vectors produced by the six different image descriptors adopted in this work. These classifiers show up with the suffix *ALL* on the front (DT-*ALL*, NB-*ALL*, NBT-*ALL*, kNN1-*ALL*, kNN3-*ALL*, and kNN5-*ALL*).

We consider seven Late fusion techniques: the meta-learning approach that uses SVM and all 36 simple available classifiers (FSVM-NORM-36), the same meta-learning ap-

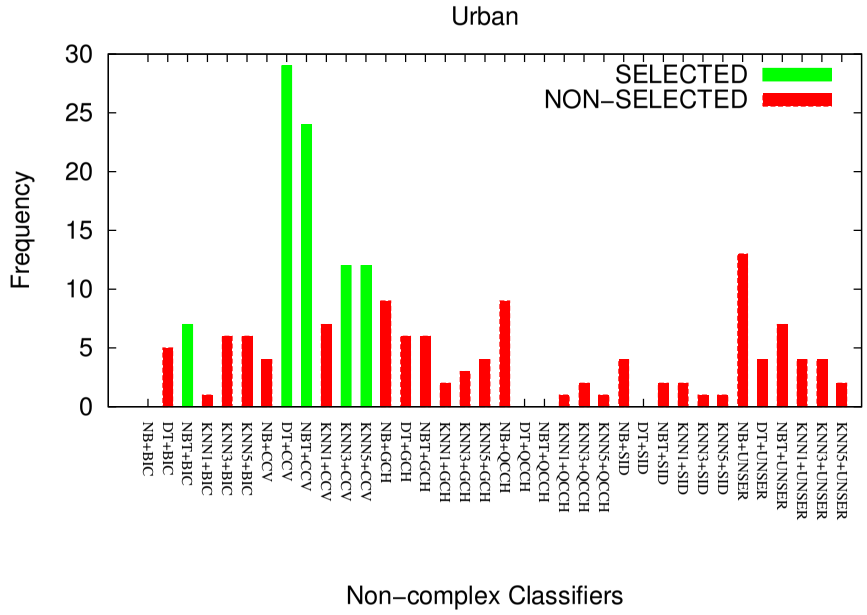


Figure 5.5: Histograms related to the occurrence of classifiers in the selection process for the Urban dataset.

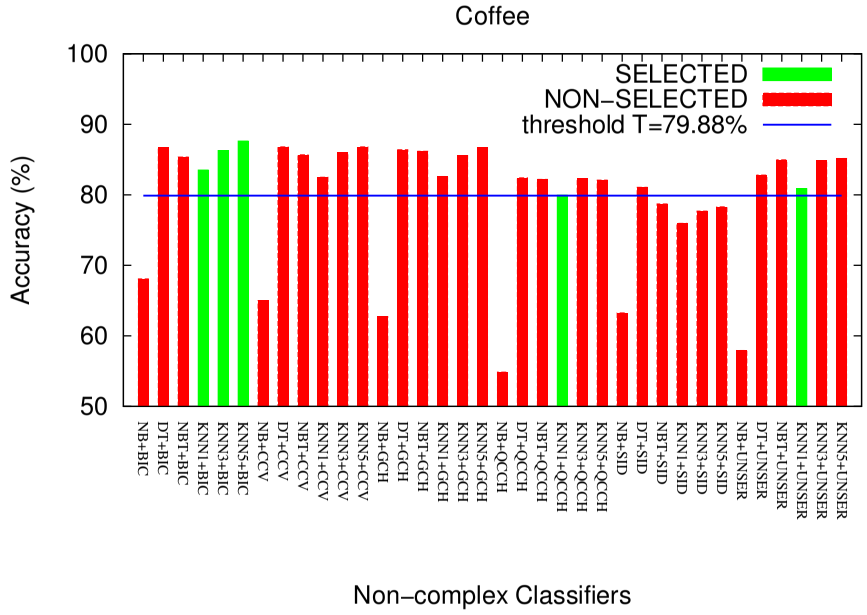


Figure 5.6: Average accuracy performances of all non-complex classifiers used in our experiments for the Coffee dataset. The blue line defines the employed threshold ($\mathcal{T}$) values, as described in Section 3.2.

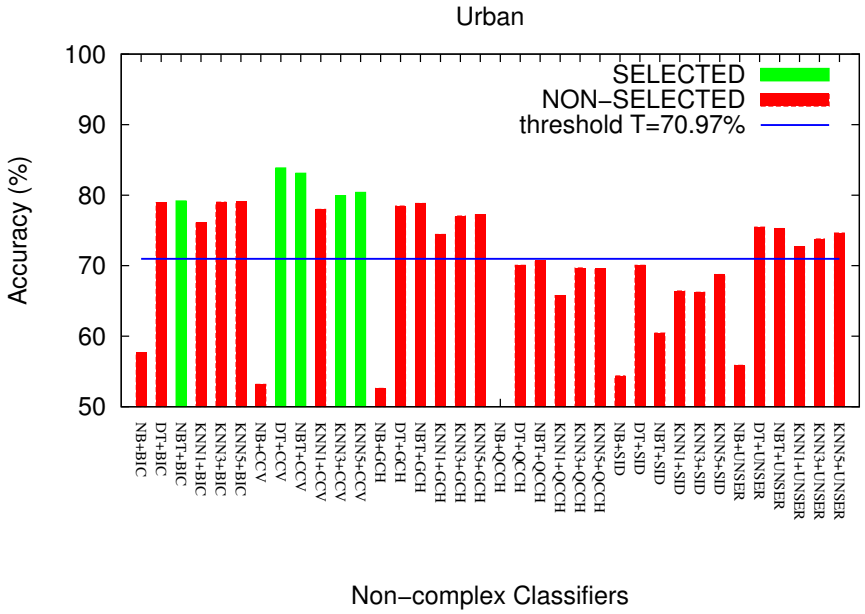


Figure 5.7: Average accuracy performances of all non-complex classifiers used in our experiments for the Urban dataset. The blue line defines the employed threshold ($\mathcal{T}$) values, as described in Section 3.2.

proach considering fewer classifiers (FSVM-NORM-5) which effectively selects the most promising simple classifiers using the new rank aggregation approach (methods with results in red in Tables 5.4 and 5.5), AdaBoost (BOOST-*SIMPLE*-5 and BOOST-*SIMPLE*-36), Bagging (BAGG-*SIMPLE*-36 and BAGG-*SIMPLE*-5), and Majority Voting (MV-36). BOOST and BAGG are multi-class AdaBoost that bootstrap aggregating techniques from the WEKA library (our implementation considers the use of 5 and 36 classifiers). MV-36 is a majority voting technique that uses the same simple/non-complex classifiers used by FSVM-NORM-36, but with a different late fusion technique. FSVM-NORM uses SVM as a meta-learning approach for discovering patterns between simple/non-complex classifier outcomes, while majority voting (MV) finds only a consensus between those simple classifiers.

As hybrid fusion, we mixed early and late fusion and then created seven other new techniques. In our meta-learning approach, we added the six early fusion techniques to the 36 existing simple classifiers. In total now, we have 42 classifiers to select and combine (FSVM-NORM-42 and FSVM-NORM-*ALL*-5). Notice that, in this case, FSVM-NORM-*ALL*-5 selects 5 classifiers out of the 42 available classifiers. Other evaluated hybrid fusion techniques include the combination with AdaBoost- and Bagging-based early fusion strategies: BOOST-*ALL*-5, BOOST-*ALL*-36, BAGG-*ALL*-36, and BAGG-

ALL-5. Finally, we consider a majority voting technique that uses 42 classifiers in the combination process (MV-42).

Table 5.6 presents achieved results for each fusion techniques, considering three different evaluation measures (Accuracy, Kappa, and Tau) in the two used datasets (Coffee and Urban). Recall that early and hybrid fusion techniques named with the suffix '*ALL*' refer to the concatenation of the feature vectors produced by the six different image descriptors considered. NORM denotes the normalized polynomial SVM kernel used in our experiments and *SIMPLE* means that the late fusion technique uses a single image descriptor. Two different image descriptors, BIC and CCV, have been chosen for *SIMPLE* in the Coffee and Urban datasets, respectively.

In general, the early fusion techniques (e.g., NB-*ALL* and kNN1-*ALL*) yield good results. These methods rely on the concatenation of different features vectors, and therefore their use with features of different domains may lead to two big challenges: curse of dimensionality and normalization problems.

Regarding the late fusion approaches (e.g., BOOST-*SIMPLE*-5 and BAGG-*SIMPLE*-36), the best results of the baselines were defined through an exhaustive and computationally expensive search involving all simple image descriptors. Our FSVM-NORM-5 approach, in turn, achieves similar results to all baselines selecting with fewer simple classifiers. Notice that our approach does not test all possible combinations of classifiers.

Similar conclusions can be raised for the results related to hybrid fusion techniques. Note for example, that the proposed fusion scheme is among the best methods. This method, however, does not present the drawbacks observed for the other fusion approaches (e.g., curse of dimensionality and expensive search of appropriate descriptors).

5.3.4 Statistical Test of Significance (t-test)

In this section, we perform statistical tests to verify if the results obtained by the proposed fusion approach differ from those observed for the baselines. T-tests have been performed to verify the statistical significance of the results, which are presented in Table 5.6. In these tests, if the $p - value$ is less than 0.05 (confidence of 95%) there is a significant difference between a pair of classifiers.

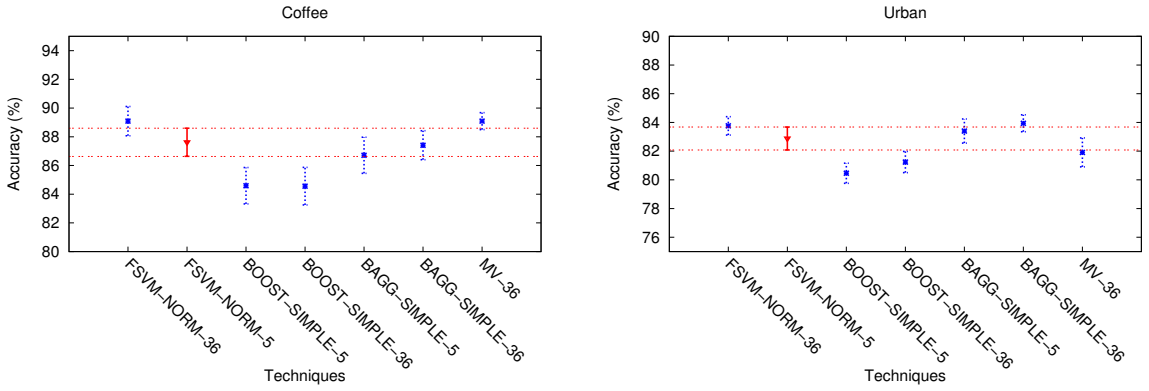
Table 5.7 shows a statistical comparison of our new rank aggregation approach (FSVM-NORM-5) with baselines. Our approach uses fewer classifiers that are selected using the methodology described in Section 3.2.3. The results considered in the tests refer to the best selection process using a late fusion technique: *Eff-Kappa_div-ALL* for the Coffee dataset (see results in red in Table 5.4) and *Eff-Kappa_div-*$DFM+IA+QSTAT$ for the Urban dataset (see Table 5.5). As it can be observed, the proposed method is better than three baselines in the Coffee dataset and better than two baselines in the Urban dataset.

Table 5.6: Classification effectiveness of different classifier fusion techniques, with their respective standard deviations. In red are rank aggregation approach, which achieves better results.

Fusion	**Techniques**	**Coffee**			**Urban**		
		Accuracy	**Kappa**	**Tau**	**Accuracy**	**Kappa**	**Tau**
Early	NB-*ALL*	69.85%±1.63	0.36±0.03	0.45±0.03	57.91%±2.97	0.27±0.04	0.41±0.03
	DT-*ALL*	85.06%±1.30	0.53±0.02	0.62±0.01	81.37%±0.49	0.57±0.02	0.64±0.01
	NBT-*ALL*	85.16%±0.55	0.53±0.03	0.63±0.02	79.45%±0.75	0.53±0.02	0.62±0.01
	kNN1-*ALL*	84.50%±1.29	0.53±0.03	0.62±0.02	79.11%±1.30	0.52±0.03	0.61±0.02
	kNN3-*ALL*	86.88%±0.74	0.59±0.02	0.66±0.01	80.75%±0.89	0.55±0.01	0.64±0.01
	kNN5-*ALL*	87.80%±0.87	0.61±0.03	0.68±0.02	81.91%±0.61	0.58±0.02	0.65±0.01
Late	FSVM-NORM-36	89.09%±1.16	0.63±0.02	0.70±0.02	83.76%±0.73	0.61±0.01	0.68±0.01
	FSVM-NORM-5	87.62%±1.12	0.59±0.04	0.67±0.02	82.88%±0.91	0.59±0.02	0.67±0.01
	BOOST-*SIMPLE*-5	84.59%±1.44	0.42±0.05	0.59±0.02	80.46%±0.80	0.48±0.04	0.61±0.02
	BOOST-*SIMPLE*-36	84.56%±1.50	0.42±0.05	0.59±0.02	81.24%±0.85	0.53±0.04	0.63±0.02
	BAGG-*SIMPLE*-5	86.71%±1.44	0.54±0.06	0.65±0.03	83.40%±0.96	0.60±0.02	0.67±0.02
	BAGG-*SIMPLE*-36	87.41%±1.15	0.56±0.04	0.66±0.02	83.94%±0.68	0.61±0.01	0.68± 0.01
	MV-36	89.09%±0.67	0.63±0.04	0.70±0.02	81.91%±1.15	0.58±0.03	0.65±0.02
Hybrid	FSVM-NORM-42	88.90%±1.15	0.63±0.03	0.70±0.02	84.71%±0.64	0.64±0.02	0.70±0.01
	FSVM-NORM-*ALL*-5	88.13%±1.29	0.60±0.04	0.68±0.03	83.14%±0.96	0.59±0.02	0.67±0.01
	BOOST-*ALL*-5	84.67%±1.42	0.42±0.04	0.59±0.02	80.83%±0.69	0.48±0.02	0.61±0.01
	BOOST-*ALL*-36	85.85%±1.76	0.50±0.06	0.62±0.03	82.15%±0.66	0.56±0.02	0.65±0.01
	BAGG-*ALL*-5	87.27%±0.89	0.57±0.03	0.66±0.02	84.32%±0.72	0.62±0.02	0.69±0.01
	BAGG-*ALL*-36	88.33%±0.87	0.60±0.04	0.68±0.02	85.56%±0.80	0.65±0.03	0.71±0.02
	MV-42	89.40%±0.50	0.64±0.02	0.71±0.01	83.07%±1.06	0.61±0.03	0.67±0.02

Table 5.7: Significance tests comparing our approach with all baselines.

Pair of Techniques	**Coffee**		**Urban**	
	p-value	**Significant**	**p-value**	**Significant**
FSVM-NORM-5 × FSVM-NORM-36	0.0190	Yes	0.1422	-
FSVM-NORM-5 × BOOST-*SIMPLE*-5	0.0005	Yes	0.0015	Yes
FSVM-NORM-5 × BOOST-*SIMPLE*-36	0.0005	Yes	0.0560	-
FSVM-NORM-5 × BAGG-*SIMPLE*-5	0.2296	-	0.3446	-
FSVM-NORM-5 × BAGG-*SIMPLE*-36	0.6870	-	0.0192	Yes
FSVM-NORM-5 × MV-36	0.0934	-	0.1020	-



(a) (b)

Figure 5.8: Confidence intervals for all late fusion techniques compared in Table 5.6.

We also computed the confidence intervals related to the results of all late fusion techniques. Figure 5.8 shows a comparison between all techniques. Notice that our approach achieves similar results to those observed for almost all baselines compared, but with fewer classifiers. The same behavior can be observed in the experiments involving our hybrid fusion technique (FSVM-NORM-*ALL*-5) and corresponding baselines.

Effectiveness results between FSVM-NORM-ALL-5 and FSVM-NORM-5 show that FSVM-NORM-ALL-5 achieves better results than FSVM-NORM-5. This fact indicates the earlier techniques were selected among selected classifiers.

Chapter 6

Conclusions

6.1 Contributions

The increase in visual data, due to the large number of monitoring cameras and popularization of mobile devices have contributed to the so-called "big-data revolution." This revolution sparked classification challenges in many areas of knowledge which have been widely applied learning techniques to solve these challenges. As there is no single extraction and learning technique that achieves good results for any application domain, data fusion approaches have been adopted.

This thesis presented a framework for selection and fusion of simple classifiers using diversity measures and meta-learning on top of classifier outcomes. Particularly in this work, we have used the support vector machine (SVM) and majority voting (MV) techniques but other learning methods could be used as well as Optimum-Path Forest (OPF), Bagging (BAGG), and AdaBoost (BOOST). Moreover, we compared several different learning methods and image descriptors in four different classification problems (scene/object classification and crop/produce recognition) showing that the proposed method is general enough to be used in a diverse range of problems [33]. Another novelty of this thesis relies on the use of diversity measures to determine which learning and image descriptor methods are more suitable to be combined in a given classification problem. Thus, three different strategies for classifier selection have been proposed.

The first selection strategy which uses a consensus approach of five different diversity measures *Double-Fault Measure* (DFM), *Q-Statistic* ($QSTAT$), *Interrater Agreement k* (IA), *Correlation Coefficient* ρ (COR), and *Disagreement Measure* (DM) to select the most complementary classifiers to be combined by a SVM technique through a meta-learning approach. Furthermore, many experiments and four datasets in different applications have been performed [33].

The second novel selection strategy is based on the *Kendall* correlation among different

diversity measures. These diversity measures were used to rank pairs of simples classifiers and the agreement of ranked lists was employed to guide the classifier selection process. In addition, we performed three different analysis with diversity measures in the classifier selection process of the classifier selection and fusion framework. First, a correlation analysis using *Kendall* score has showed to be possible that different diversity measures have different opinions. In this experiment with remote sensing datasets, we have showed that $COR \times QSTAT$ achieved the highest correlation coefficients, while $IA \times QSTAT$, the lowest. High correlation coefficients mean that both diversity measures have similar opinions about which classifiers might be selected. Low correlation coefficients in turn, mean that both diversity measure have a certain degree of divergence about which classifiers to select. We also performed a behavioral analysis, based on which we showed two forms for selecting classifiers: (1) Single, which used only one diversity measure in the classifier selection process; (2) *Kendall*, which used two measures combined through *Kendall* correlation coefficients. Finally, a comparison using the classifier accuracy has been performed using the best classifier selection approach that we could find using Tables 5.4 and 5.5 by means of the use of diversity measures and the proposed methodology based on *Kendall*. The $IA + QSTAT$ approach has achieved the same results than all baselines using fewer classifiers than the original approach (ALL). Statistical tests have been performed to corroborate the claims [34].

The third novel strategy for selecting classifiers is based on a rank aggregation approach. This approach combines different diversity and evaluation measures to create a final and unified list to guide the classifier selection process. In addition, we performed two different analysis with diversity measures in the classifier selection process of a fusion framework [33]. Obtained results demonstrate that the proposed fusion approach yield comparable results to those observed for baselines. In our solution, however, fewer classifiers are usually selected. Performed statistical tests confirm our claims [35].

The main home message of this work is that for solving complex problems such as the ones involving different applications, we need to take advantage of different and complementary information regarding description and learning methods. However, choosing the most appropriate methods for combining is not a trivial task. Previous work in the literature has proposed interesting forms for selecting and combining complementary methods such as the ones based on diversity measures [33].

In this thesis, we go beyond previous efforts for combining classifiers and explore the power of three different selection strategies (consensus, *Kendall*, and ranked list aggregation) and diversity measures. The proposed approaches allows for a much more efficient combination of classifiers. For instance, with only five classification methods, the proposed approach has the same statistical classification results of the works proposed in [33, 34] when using 10 and 15 classifiers, respectively. Besides being more efficient while select-

ing promising complementary classifiers, the proposed approach is not custom-tailored for different applications, more specifically for remote sensing imagery. The approach is general enough to be used within any classification problem dealing with complementary features and classifiers. Thus, the hypothesis of this thesis has been confirmed.

The proposed framework overcomes the best used classifier and the well-known majority voting, Bagging, and AdaBoost fusion approaches. In fact, our framework is able to combine classifiers more effectively than the baselines. Different from other approaches, our method is able to select classifiers and also learn, indirectly, which descriptors (and therefore visual properties) are more appropriate for the target application. To keep a high recognition rate with the minimum computational effort, our classifier selection strategy explores the use of diversity measures, which allow the combination of less correlated and highly-effective classifiers. Furthermore, we also performed statistical tests aiming at finding the lowest number of classifiers for a given problem which achieve the same effectiveness while more efficiently than the best baselines for each dataset tested.

One might argue that using different classifiers and descriptors during training might increase the training time substantially. Although this is partly true, we note that most of fusion methods in the literature sacrifice part of the training while aiming at finding more discriminative learning methods to be used during deployment (operation). With respect to this, it is clear that our method brings an interesting property to the table that is different from some methods in the literature in the sense it is highly flexible and parallelizable. It is flexible in the sense it can use any set of descriptors or classifiers for fusion. In addition, we can use any learning method in the late fusion after selecting the best learners through diversity measure analysis. It is highly parallelizable since each combination of descriptor and learning method can be used in a different thread or even processor and even this task can also be divided by means of modern classifiers.

Another important advantage of the proposed method is that it considers each descriptor and learning method independently, therefore, it does not incur in normalization problems that result of direct combination of features (e.g., by means of concatenating feature vectors). The concatenation also brings problems regarding dimensionality, which is also not a problem for the proposed method. Moreover, as we showed in Section 5.1.3, in most cases, our approach achieves statistically better results than all baselines using reduced training sets. This means that the proposed methods can compensate the additional operations with diversity measures by requiring less training examples.

6.2 Future Work

This thesis has created opportunity for further investigations in relation to all the research challenges presented. Figure 6.1 shows, in red, steps of the proposed framework that could

be improved/extended in future work.

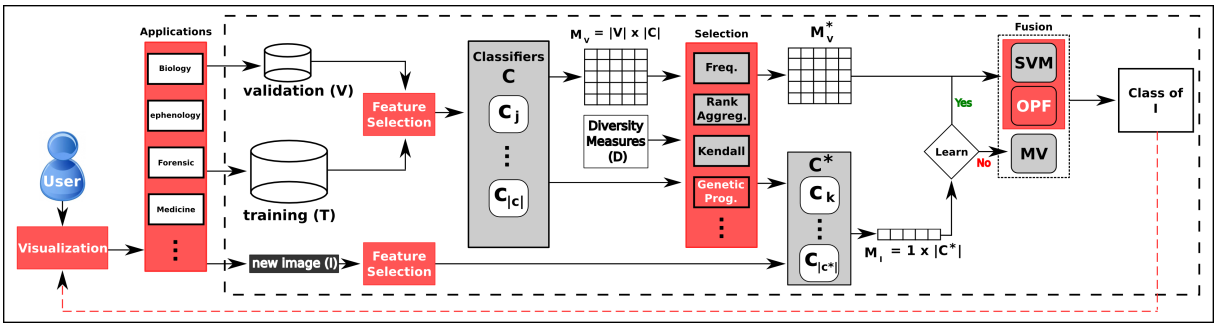


Figure 6.1: Steps/modules of the proposed framework that could be extended/improved in future work.

Visualization Techniques

- Adding different visualization techniques, in the proposed framework, offering user a better understanding of the achieved results, then guiding the for decision-making process in real-world scenarios [52, 55];

- Investigating visualization techniques, in the proposed framework, to guide machine learning technique developers in the classifier selection process through user interaction. Users might to interact with the framework in the classifier learning step, thus improving the classifier selection [76].

- Exploring visualization techniques to help identify novel classifier selection strategies.

Feature Selection

- Exploring feature selection techniques to reduce descriptor dimension size through filter and wrapper techniques. One possible strategy would be to use Principal Components Analysis (PCA) [53], Independent Components Analysis (ICA) [51], and Genetic Programming (GP) [60] approaches in this step.

Classifier Selection

- Implementing novel classifier selection strategies based on evolutionary algorithm approaches (e.g., genetic programming [60] and genetic algorithm [71]).

- Exploring re-ranking approaches to improve the classifier selection process [80–82].

Classifier Fusion

- Evaluating the proposed framework with classifier Optimum-Path Forest (OPF) [77]. OPF is a classification method based on graph, in which instances are represented as nodes of this graph and the edges are distance values calculated between two instances.

- Exploring an approach based on association rules, Lazy Associative Classification (LAC) [107], to combine simple classifiers.

Diversity Measures

- Investigating non-pairwise diversity measures (e.g., entropy and Kohavi-Wolpert variance), which were not used in this thesis. Maybe they might to describe better the real diversity inside the set of classifiers [61].

Applications

- Validating the proposed framework in other applications (e.g., phenology, spoofing detection, biometric recognition, and metadata deduplication).

6.3 List of Publications

This thesis has generated publications directly and indirectly related to its content.

List of journal papers:

1. *A Framework for Selection and Fusion of Pattern Classifiers in Multimedia Recognition* [33]. **Faria, F. A.** ; Santos, J. A. ; Rocha, A. ; Torres, R. da S. in the Pattern Recognition Letters, 39:52-64, 2014. http://dx.doi.org/10.1016/j.patrec.2013.07.014

2. *Rank Aggregation for Pattern Classifier Selection in Remote Sensing Images* [35]. **Faria, F. A.** ; Pedronette, D. C. G. ; Santos, J. A. ; Rocha, A. ; Torres, R. da S. In: IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 7(4):1103-1115, 2014. http://dx.doi.org/10.1109/jstars.2014.2303813

List of conference papers:

1. *Evaluation of Time Series Distance Functions in the Task of Detecting Remote Phenology Patterns* [15]. Conti, J ; **Faria, F. A.** ; Almeida, J. ; Camolesi, L. ; Alberton, B. Morellato, L. ; Torres, R. da S. In: International Conference of Pattern Recognition (ICPR), 2014. *To appear.*

2. *Classifier Selection based on the Correlation of Diversity Measures: When Fewer is More* [34]. **Faria, F. A.** ; Santos, J. A. ; Sarkar, S. ; Rocha, A. ; Torres, R. da S. In: Conference on Graphics, Patterns and Images (SIBGRAPI), pages 16-23, 2013. http://dx.doi.org/10.1109/SIBGRAPI.2013.12

3. *Automatic fusion of region-based classifiers for coffee crop recognition* [31]. **Faria, F. A.** ; Santos, J. A. ; Torres, R. da S. ; Rocha, A. ; Falcão, A. X. . In: IEEE International Geoscience and Remote Sensing Symposium (IGARSS), pages 2221-2224, 2012. http://dx.doi.org/10.1109/IGARSS.2012.6351058

4. *Automatic Classifier Fusion for Produce Recognition* [32]. **Faria, F. A.** ; Santos, J. A. ; Rocha, A. ; Torres, R. da S. In: Conference on Graphics, Patterns and Images (SIBGRAPI), pages 252-259, 2012. http://dx.doi.org/10.1109/SIBGRAPI.2012.42

5. *Descriptor Correlation Analysis for Remote Sensing Image Multi-Scale Classification* [21], Santos, J.A. ; **Faria, F. A.** ; Torres, R .da S. ; Rocha, A. ; Gosselin, P-H. ; Philipp-Foliguet, S. ; Falcão, A. X. In: International Conference on Pattern Recognition (ICPR), pages 3078-3081, 2012.

6. *Improving Image Classification through Descriptor Combination* [69]. Mansano, A. ; Matsuoka, J. A. ; Afonso, L. C. S. ; Papa, J. P. ; **Faria, F. A.** ; Torres, R. da S. In: Conference on Graphics, Patterns and Images (SIBGRAPI), pages 324-329, 2012. http://dx.doi.org/10.1109/SIBGRAPI.2012.52

7. *RECOD at ImageCLEF 2011: Medical Modality Classification using Genetic Programming* [30]. **Faria, F. A.** ; CALUMBY, R. T. ; Torres, R. da S. In: Conference on Multilingual and Multimodal Information Access Evaluation (CLEF), pages 1-8, 2011.

Bibliography

[1] M. Antonie, O. R. Zaïane, and A. Coman. Application of data mining techniques for medical image classification. In *Workshop on Multimedia Data Mining at ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pages 94–101, 2001.

[2] J. A. Benediktsson, J. Chanussot, and W. M. Moon. Very high-resolution remote sensing: Challenges and opportunities [point of view]. *Proceedings of the IEEE*, 100(6):1907–1910, 2012.

[3] J. A. Benediktsson, J. R. Sveinsson, O. K. Ersoy, and P. H. Swain. Parallel consensual neural networks. *IEEE Transactions on Neural Networks*, 8(1):54–64, January 1997.

[4] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 1 edition, 2006.

[5] R. M. Bolle, J. H. Connell, N. Haas, R. Mohan, and G. Taubin. Veggievision: A produce recognition system. In *IEEE Workshop on Applications of Computer Vision*, pages 1–8, Sarasota, USA, 1996.

[6] B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In *Workshop on Computational Learning Theory*, COLT '92, pages 144–152, 1992.

[7] M. R. Boutell, J. Luo, X. Shen, and C. M. Brown. Learning multi-label scene classification, 2004.

[8] L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

[9] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, October 2001.

[10] R. L. Brennan and D. J. Prediger. Coefficient kappa: Some uses, misuses, and alternatives. *Educational and psychological measurement*, 1981.

[11] D. Brooks. The philosophy of data. In *The New York Times*, page A23, 2013. February, 5th.

[12] Gavin Brown, Jeremy Wyatt, Rachel Harris, and Xin Yao. Diversity creation methods: A survey and categorisation. *Journal of Information Fusion*, 6:5–20, 2005.

[13] A. Çarkacıoğlu and F. T. Yarman-Vural. SASI: A Generic Texture Descriptor for Image Retrieval. *Pattern Recognition (PR)*, 36(11):2615–2633, 2003.

[14] C. Chang and C. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):27:1–27:27, May 2011.

[15] J Conti, F. A. Faria, J. Almeida, L. Camolesi, B. Alberton, L. Morellato, and R. da S. Torres. Evaluation of time series distance functions in the task of detecting remote phenology patterns. In *International Conference of Pattern Recognition (ICPR)*, 2014.

[16] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, 2000.

[17] W. Bruce Croft. Combining approaches to information retrieval. In W. B. Croft, editor, *Advances in Information Retrieval*, volume 7 of *The Information Retrieval*, pages 1–36. Springer US, 2002.

[18] T. J. de Carvalho, C. Riess, E. Angelopoulou, H. Pedrini, and A. Rocha. Exposing digital image forgeries by illumination color classification. *IEEE Transactions on Information Forensics and Security (TIFS)*, 8(7):1182–1194, 2013.

[19] J. Deng, A. C. Berg, K. Li, and L. Fei-Fei. What does classifying more than 10,000 image categories tell us? In *European Conference on Computer Vision (ECCV)*, pages 71–84, Berlin, Heidelberg, 2010. Springer-Verlag.

[20] L. Didaci, G. Fumera, and F. Roli. Diversity in classifier ensembles: Fertile concept or dead end? In *International Conference on Multiple Classifier Systems (MCS)*, pages 37–48, 2013.

[21] J. A. dos Santos, F. A. Faria, R. da S. Torres, A. Rocha, P.-H. Gosselin, S. Philipp-Foliguet, and A. X. Falcão. Descriptor correlation analysis for remote sensing image multi-scale classification. In *International Conference of Pattern Recognition (ICPR)*, pages 3078–3081, 2012.

[22] J. A. dos Santos, P.-H. Gosselin, S. Philipp-Foliguet, R. da S. Torres, and A. X. Falcão. Interactive multiscale classification of high-resolution remote sensing images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing (JSTARS)*, PP(99):1–15, 2013.

[23] J. A. dos Santos, P.H. Gosselin, S. Philipp-Foliguet, R. da S. Torres, and A. X. Falcão. Multiscale classification of remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing (TGARS)*, 50(10):3764–3775, 2012.

[24] J. A. dos Santos, O. A. B. Penatti, and R. da S. Torres. Evaluating the potential of texture and color descriptors for remote sensing image retrieval and classification. In *International Conference on Computer Vision Theory and Applications*, pages 203–208, Angers, France, May 2010.

[25] J.A. Dos Santos, P.-H. Gosselin, S. Philipp-Foliguet, R. Da S Torres, and A. Xavier Falcao. Interactive multiscale classification of high-resolution remote sensing images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing (JSTARS)*, 6(4):2020–2034, Aug 2013.

[26] P. Du, J. Xia, W. Zhang, K. Tan, Y. Liu, and S. Liu. Multiple classifier system for remote sensing image classification: a review. *Sensors (Basel, Switzerland)*, 12(4):4764, 2012.

[27] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification (2Nd Edition)*. Wiley-Interscience, 2000.

[28] R. P. W. Duin and E. Pekalska. Open issues in pattern recognition. In *International Conference on Computer Recognition Systems (CORE)*, volume 30, pages 27–42, 2005.

[29] S. Džeroski and B. Ženko. Is combining classifiers with stacking better than selecting the best one? *Machine Learning*, 54(3):255–273, March 2004.

[30] F. A. Faria, R. T. Calumby, and R. da S. Torres. Recod at imageclef 2011: Medical modality classification using genetic programming. In *CLEF (Notebook Papers/Labs/Workshop)*, pages 1–8, 2011.

[31] F. A. Faria, J. A. dos Santos, R. da S. Torres, A. Rocha, and A. X. Falcão. Automatic fusion of region-based classifiers for coffee crop recognition. In *IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, pages 2221–2224, 2012.

[32] F. A. Faria, J. A. dos Santos, A. Rocha, and R. da S. Torres. Automatic classifier fusion for produce recognition. In *XXIV Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 252–259, 2012.

[33] F. A. Faria, J. A. dos Santos, A. Rocha, and R. da S. Torres. A framework for selection and fusion of pattern classifiers in multimedia recognition. *Pattern Recognition Letters (PRL)*, 39:52–64, april 2013.

[34] F. A. Faria, J. A. dos Santos, S. Sarkar, A. Rocha, and R. da S. Torres. Classifier selection based on the correlation of diversity measures: When fewer is more. In *XXV Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 16–23, Arequipa, Peru, august 2013.

[35] F. A. Faria, D. C. G. Pedronette, J. A. dos Santos, A. Rocha, and R. da S. Torres. Rank aggregation for pattern classifier selection in remote sensing images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing (JSTARS)*, 7(4):1103–1115, April 2014.

[36] L. Fei-fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *IEEE International Conference on Computer Vision and Pattern Recognition Workshop.*, page 178, 2004.

[37] B. Fernando, E. Fromont, D. Muselet, and M. Sebban. Discriminative feature fusion for image classification. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3434–3441, june 2012.

[38] Y. Freund and R. E. Schapire. Experiments with a new boosting algorithm. *International Conference on Machine Learning (ICML)*, pages 148–156, 1996.

[39] J. Friedman, T. Hastie, and R. Tibshirani. *The Elements of Statistical Learning.* Springer, 1 edition, 2001.

[40] P. Gehler and S. Nowozin. On feature combination for multiclass object classification. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 221–228, 29 2009-oct. 2 2009.

[41] L. Guigues, J. Cocquerez, and H. Le Men. Scale-sets image analysis. *International Journal of Computer Vision (IJCV)*, 68(3):289–317, 2006.

[42] R. W. Hamming. Error detecting and error correcting codes. *Bell System Technical Journal*, 29(2):147–160, 1950.

[43] II Hampshire, J. B. and A. Waibel. The meta-pi network: building distributed knowledge representations for robust multisource pattern recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 14(7):751–769, jul 1992.

[44] Z. Harchaoui, M. Douze, M. Paulin, M. Dudik, and J. Malick. Large-scale image classification with trace-norm regularization. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3386–3393, june 2012.

[45] Jonathon Hare, Jamie Davies, Sina Samangooei, and Paul Lewis. Placing photos with a multimodal probability density function. In *Proceedings of International Conference on Multimedia Retrieval (ICMR)*, pages 329:329–329:336, New York, NY, USA, 2014. ACM.

[46] Tin Kam Ho, J. J. Hull, and S.N. Srihari. Decision combination in multiple classifier systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 16(1):66–75, jan 1994.

[47] J. Hong, J. Min, U. Cho, and S. Cho. Fingerprint classification using one-vs-all support vector machines dynamically ordered with naive bayes classifiers. *Pattern Recognition (PR)*, 41(2):662–671, February 2008.

[48] J. Hou, B. Zhang, N. Qi, and Y. Yang. Evaluating feature combination in object classification. In *International Conference on Advances in Visual Computing–Volume Part II*, pages 597–606, 2011.

[49] C. Huang and Q. Liu. An orientation independent texture descriptor for image retrieval. In *International Conference on Communications, Circuits and Systems*, pages 772–776, 2007.

[50] J. Huang, R. Kumar, M. Mitra, W. Zhu, and R. Zabih. Image indexing using color correlograms. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 762–768, 1997.

[51] A. Hyvärinen and E. Oja. Independent component analysis: algorithms and applications. *Neural Networks*, 13(4-5):411–430, 2000.

[52] Alfred Inselberg. *Parallel Coordinates: Visual Multidimensional Geometry and Its Applications*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2009.

[53] I. T. Jolliffe. *Principal Component Analysis*. Springer, second edition, October 2002.

[54] A. Jurek, Y. Bi, S. Wu, and C. Nugent. Classification by cluster analysis: a new meta-learning based approach. In *International Conference on Multiple Classifier Systems (MCS)*, pages 259–268, 2011.

[55] Daniel A. Keim. Designing pixel-oriented visualization techniques: Theory and applications. *IEEE Transactions on Visualization and Computer Graphics*, 6(1):59–78, January 2000.

[56] M. G. Kendall. A New Measure of Rank Correlation. *Biometrika*, 30(1/2):81–93, June 1938.

[57] D. Kim, Y. Baek, and W. Kim. Reducing overfitting of adaboost by clustering-based pruning of hard examples. In *Proceedings of the 7th International Conference on Ubiquitous Information Management and Communication*, page 90, 2013.

[58] B. C. Ko, J. W. Gim, and J. Y. Nam. Cell image classification based on ensemble features and random forest. *Electronics Letters*, 47(11):638–639, 26 2011.

[59] R. Kohavi. Scaling up the accuracy of naive-bayes classifiers: a decision-tree hybrid. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 202–207. AAAI Press, 1996.

[60] J. Koza. *Genetic Programming: On the programming of computers by natural selection*. MIT Press, 1992.

[61] L. I. Kuncheva. *Combining Pattern Classifiers: Methods and Algorithms*. Wiley-Interscience, 2004.

[62] L. I. Kuncheva and C. J. Whitaker. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine Learning*, 51, 2003.

[63] D. Lee and H. Kim. A fast content-based indexing and retrieval technique by the shape information in large image database. *Journal of Systems and Software*, 56(2):165–182, March 2001.

[64] J. Li, J. M. Bioucas-Dias, and A. Plaza. Spectral-spatial classification of hyperspectral data using loopy belief propagation and active learning. *IEEE Transactions on Geoscience and Remote Sensing (TGARS)*, 51(2):844–856, 2013.

[65] Yuanqing Lin, Fengjun Lv, Shenghuo Zhu, Ming Yang, T. Cour, Kai Yu, Liangliang Cao, and T. Huang. Large-scale image classification: Fast feature extraction and svm training. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1689–1696, june 2011.

[66] S. Lomax and S. Vadera. A survey of cost-sensitive decision tree induction algorithms. *ACM Computing Surveys (CSUR)*, 45(2):16:1–16:35, March 2013.

[67] Z. Ma and R. L. Redmond. Tau coefficients for accuracy assessment of classification of remote sensing data. *Photogrametric Engineering and Remote Sensing*, 1995.

[68] F. Mahmoudi, J. Shanbehzadeh, A. Eftekhari-Moghadam, and H. Soltanian-Zadeh. Image retrieval based on shape similarity by edge orientation autocorrelogram. *Pattern Recognition (PR)*, 36(8):1725–1736, 2003.

[69] A. Mansano, J. A. Matsuoka, L. C. S. Afonso, J. P. Papa, F. A. Faria, and R. da S Torres. Improving image classification through descriptor combination. In *XXIV Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 324–329, aug. 2012.

[70] R. M. Martins, D. B. Coimbra, R. Minghim, and A. C. Telea. Visual analysis of dimensionality reduction quality for parameterized projections. *Computers & Graphics*, 41(0):26 – 42, 2014.

[71] Zbigniew Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs (3rd Ed.)*. Springer-Verlag, London, UK, UK, 1996.

[72] E. Morais, S. Goldenstein, A. Ferreira, and A. Rocha. Automatic tracking of indoor soccer players using videos from multiple cameras. In *Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 174–181, aug. 2012.

[73] R. B Myneni, F. G. Hall, P. J. Sellers, and A. L. Marshak. The interpretation of spectral vegetation indexes. *IEEE Transactions on Geoscience and Remote Sensing (TGARS)*, 33(2):481–486, 1995.

[74] E. F. Nakamura, A. A. F. Loureiro, and A. C. Frery. Information fusion for wireless sensor networks: Methods, models, and classifications. *ACM Computing Surveys (CSUR)*, 39(3), September 2007.

[75] N. C. Oza and K. Tumer. Classifier ensembles: Select real-world applications. *Information Fusion*, 9(1):4–20, January 2008.

[76] J. G. S. Paiva, W. R. Schwartz, H. Pedrini, and R. Minghim. Semi-supervised dimensionality reduction based on partial least squares for visual analysis of high dimensional data. *Journal Computer Graphics Forum*, 31(3pt4):1345–1354, June 2012.

[77] J. P. Papa, A. X. Falcao, and C. T. N. Suzuki. Supervised pattern classification based on optimum-path forest. *International Journal of Imaging Systems and Technology*, 19(2):120–131, 2009.

[78] G. Pass, R. Zabih, and J. Miller. Comparing images using color coherence vectors. In *ACM International Conference on Multimedia (MM)*, pages 65–73, 1996.

[79] A. Passerini, M. Pontil, and P. Frasconi. New results on error correcting output codes of kernel machines. *IEEE Transactions on Neural Networks*, 15(1):45–54, January 2004.

[80] D. C. G. Pedronette and R. da S. Torres. Exploiting pairwise recommendation and clustering strategies for image re-ranking. *Information Sciences*, 207:19–34, 2012.

[81] D. C. G. Pedronette and R. da S. Torres. Image re-ranking and rank aggregation based on similarity of ranked lists. *Pattern Recognition (PR)*, 46(8):2350–2360, 2013.

[82] D. C. G. Pedronette and R. da S. Torres. Exploiting contextual spaces for image re-ranking and rank aggregation. In *ACM International Conference on Multimedia Retrieval (ICMR)*, pages 13:1–13:8, 2011.

[83] O. A. B. Penatti, E. Valle, and R. da S. Torres. Comparative study of global color and texture descriptors for web image retrieval. *Journal of Visual Communication and Image Representation (JVCI)*, 23(2):359 – 380, 2012.

[84] M. P. Perrone and L. N. Cooper. *When Networks Disagree: Ensemble Methods for Hybrid Neural Networks*. Chapman and Hall, 1993.

[85] F. Petitjean, C. Kurtz, N. Passat, and P. Gançarski. Spatio-temporal reasoning for the classification of satellite image time series. *Pattern Recognition Letters (PRL)*, 33(13):1805– 1815, 2012.

[86] J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1(1):81–106, March 1986.

[87] J. R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993.

[88] C. C. O. Ramos, A. N. Souza, G. Chiachia, A. X. Falcão, and J. P. Papa. A novel algorithm for feature selection using harmony search and its application for non-technical losses detection. *Computers and Electrical Engineering*, 37(6):886–894, November 2011.

[89] A. Rocha, D. C. Hauagge, J. Wainer, and S. Goldenstein. Automatic fruit and vegetable classification from images. *Elsevier Computers and Electronics in Agriculture (COMPAG)*, 70(1):96–104, 2010.

[90] A. Rocha, J. P. Papa, and L. A. A. Meira. How far do we get using machine learning black-boxes? *International Journal of Pattern Recognition and Artificial Intelligence*, 26(2):1261001–1–1261001–23, 2012.

[91] L. Rokach. Ensemble-based classifiers. *Journal Artificial Intelligence Review*, 33(1-2):1–39, February 2010.

[92] Arun A. Ross, Karthik Nandakumar, and Anil K. Jain. *Handbook of Multibiometrics (International Series on Biometrics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[93] R. E. Schapire. A brief introduction to boosting. In *International Joint Conference on Artificial Intelligence*, pages 1401–1406, 1999.

[94] N. Segata and E. Blanzieri. Fast and scalable local kernel machines. *Journal Machine Learning Research*, 99:1883–1926, August 2010.

[95] C. A. Shipp and L. I. Kuncheva. An investigation into how adaboost affects classifier diversity. *IPMU*, pages 203–208, 2009.

[96] R. Stehling, M. Nascimento, and A. Falcao. A compact and efficient image retrieval approach based on border/interior pixel classification. In *ACM Conference on Information and Knowledge Management (CIKM)*, pages 102–109, 2002.

[97] B. Sun, J. Luo, S. Shu, and N. Yu. Introduce randomness into adaboost for robust performance on noisy data. In *Fuzzy Systems and Knowledge Discovery (FSKD), 2010 Seventh International Conference on*, volume 4, pages 1858–1861, 2010.

[98] Y. Sun, S. Todorovic, and J. Li. Reducing the overfitting of adaboost by controlling its data distribution skewness. *International Journal of Pattern Recognition and Artificial Intelligence*, 20(07):1093–1116, 2006.

[99] C. T. N. Suzuki, J. F. Gomes, A. X. Falcao, J. P. Papa, and S. Hoshino-Shimizu. Automatic segmentation and classification of human intestinal parasites from microscopy images. *IEEE Transactions on Biomedical Engineering*, 60(3):803–812, March 2013.

[100] C. T. N. Suzuki, J. F. Gomes, A. X. Falcao, J. P. Papa, and S. H. Shimizu. Automatic segmentation and classification of human intestinal parasites from microscopy images. *IEEE Transactions on Biomedical Engineering*, 60(3):803–12, 2012.

[101] M. Swain and D. Ballard. Color indexing. *International Journal of Computer Vision (IJCV)*, 7(1):11–32, 1991.

[102] P. Tan, M. Steinbach, and V. Kumar. *Introduction to Data Mining, (First Edition)*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2005.

[103] B. Tao and B. Dickinson. Texture recognition and image retrieval using gradient indexing. *Journal of Visual Communication and Image Representation (JVCI)*, 11(3):327–342, 2000.

[104] R. da S. Torres, A. X. Falcão, M. A. Gonçalves, J. P. Papa, B. Zhang, W. Fan, and E. A. Fox. A genetic programming framework for content-based image retrieval. *Pattern Recognition (PR)*, 42(2):283–292, February 2009.

[105] D. Tuia, M. D. Mura, M. Volpi, R. Flamary, and A. Rakotomamonjy. Discovering relevant spatial filterbanks for vhr image classification. In *International Conference on Pattern Recognition (ICPR)*, pages 3212–3215, 2012.

[106] M. Unser. Sum and difference histograms for texture classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 8(1):118–125, 1986.

[107] A. Veloso, W. Meira, M. Gonçalves, H. M. Almeida, and M. Zaki. Calibrated lazy associative classification. *Information Sciences*, In Press, Corrected Proof, 2010.

[108] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages I–511–I–518, 2001.

[109] R. Weber, H. Schek, and S. Blott. A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces. In *Proceedings of the International Conference on Very Large Data Bases (VLDB)*, pages 194–205, 1998.

[110] David H. Wolpert. Stacked generalization. *Neural Networks*, 5:241–259, 1992.

[111] David H. Wolpert. The lack of a priori distinctions between learning algorithms, 1996.

[112] Michał Woźniak, Manuel Graña, and Emilio Corchado. A survey of multiple classifier systems as hybrid systems. *Information Fusion*, 16(0):3– 17, 2014. Special Issue on Information Fusion in Hybrid Intelligent Fusion Systems.

[113] X. Xia and L. O. Gorman. Innovations in fingerprint capture devices. *Pattern Recognition (PR)*, 36(2):361–369, 2003.

[114] Jianxiong Xiao, J. Hays, K.A. Ehinger, A. Oliva, and A. Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3485–3492, june 2010.

[115] C. Yang, J.H. Everitt, Q. Du, B. Luo, and J. Chanussot. Using high-resolution airborne and satellite imagery to assess crop growth and yield variability for precision agriculture. *Proceedings of the IEEE*, 101(3):582–592, 2013.

[116] J. Zegarra, N. Leite, and R. Torres. Wavelet-based feature extraction for fingerprint image retrieval. *Journal of Computational and Applied Mathematics*, 227(2):294–307, 2008.

[117] W. Zhou, G. Huang, A. Troy, and M. L. Cadenasso. Object-based land cover classification of shaded areas in high spatial resolution imagery of urban areas: A comparison study. *Remote Sensing of Environment*, 113:1769–1777, 2009.