INTEGRATIVE SPARSE MODELING AND CLASSIFICATION OF BIOMEDICAL IMAGING PATTERNS

by

KENI ZHENG

A DISSERTATION

Submitted in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in the Applied Mathematics and
Mathematical Physics Graduate Program
of Delaware State University

DOVER, DELAWARE
August 2018

This dissertation is approved by the following members of the Final Oral Review Committee:

Dr. Sokratis Makrogiannis, Committee Chairperson, Department of Mathematical Sciences, Delaware State University
Dr. Jinjie Liu, Committee Member, Department of Mathematical Sciences, Delaware State University
Dr. Xiquan Shi, Committee Member, Department of Mathematical Sciences, Delaware State University
Dr. Thomas Planchon, Committee Member, Department of Physics and Engineering, Delaware State University
Dr. Predrag Bakic, External Committee Member, Department of Radiology, University of Pennsylvania

COPYRIGHT

DEDICATION

To my loving parents
and family.

ACKNOWLEDGEMENTS

Firstly, I would like to express my sincere gratitude to my adviser Prof. Dr. Sokratis Makrogiannis for the continuous support of my Ph.D study and related research, as well as for his patience, motivation, and immense knowledge. His insights and understanding of the field constantly played a significant role in the increase of my own knowledge of the field and greatly deepened my understanding of the work I have conducted. It is also because of him that I was able to grow as an individual from knowing very little about this field, into the person I am now completing research projects and writing this dissertation. Throughout the process of my research, from the topic selection to the specific experiment, and from the analysis of the results to the writing of the article and the revision of the paper, each step of the way carries with it the hard work and wisdom I have received from my adviser. His rigorous scientific research has helped and guided me to be able to form a correct and sound research ideology. His direction is clear, and he contributes his time, ideas, and suggestions to share his knowledge with me, so that I can continue to learn and progress. I would also like to share my thanks for the funding that I have been providing for the past few years, so that I could successfully complete my Ph.D. studies. And I would like to thank the provided facilities and MIVIC Lab Office, which allowed us to have the opportunity to discuss and communicate with each other.

I would also like to extend my sincere gratitude to the committee members: Dr. Xiquan Shi, Dr. Jinjie Liu, Dr. Bakic Predrag and Dr. Thomas A. Planchon. Thanks to all the members of the committee for taking the time to read and make suggestions and amendments to my dissertation. Thanks to Dr. Shi for helping me since my graduate studies and for promoting the accurate establishment of my knowledge in the study of mathematics as well as my in-depth understanding of the mathematical field. I have benefited enormously from several of Dr. Shi's mathematics courses in the establishment of my mathematical abstract thinking skills and the consolidation of my foundation of mathematical knowledge. I also thankful for Dr. Liu, who always patiently explained and answered any questions that I encountered in my studies. Dr. Liu's courses both strengthened my grasp of the research field and contributed greatly to my knowledge reserve. Thanks also to Dr. Bakic for providing me with numerous suggestions for strengthening and promoting my scientific research, as well as sharing ideas on the direction I should take. And I would like to thank Dr. Planchon, who through his insight in the field of physics also supplied me with a plethora of advice in my field of study.

I am grateful to the Department of Mathematics Science and the Applied Mathematics and Mathematical Physics program for cultivating me. They have provided continued support of my tuition from my graduate studies all the way to the end of my doctoral studies. Before I received funding support from my adviser, the Department of Mathematics Science provided me with many work opportunities that would allow me to complete my studies. I would also like to thank all of the faculty and staff members at the Department of Mathematics Science. Each and every one of them made me feel the collective warmth of the Department of Mathematics Science. Every greeting they shared with me made me happy to study here and share interactions with everyone. Just like a family, we cared for and greeted each other.

My thanks also goes out to OSCAR for the various facilities and research seminars they provided. The safe and healthy environment here allows us to feel at ease and enjoy our research. Many thanks as well to all of the faculty and staff here, who created for us this perfect research environment, and made sure that all the affairs here were carried out in an orderly manner.

Last but certainly not least, I would like to thank my parents, family and friends. My parents will always love me unconditionally, accept me, and support me in all of my endeavors. No matter what I encounter, they are always there to help me so that I am not afraid, and give me the courage to face whatever crosses my path. Their education and enlightenment also made me a better researcher. I never felt alone, in fact I felt as if I was in the company of all my loved ones because of my familys extended care and support even though we may be far apart. Thanks also to all my friends for always quietly standing by my side. Some of them gave me positive advice and encouragement at each turning point I faced in my life. As for my other friends, even though we did not always have time to talk often, anytime I needed them, they would immediately appear at my side. It is these people that made me feel that not everything was as difficult as I anticipated. Without them, it would have been much harder for me to complete my Ph.D. degree, and for that, my thanks to them is something my words cannot begin to express.

Integrative Sparse Modeling and Classification of Biomedical Imaging Patterns

Keni Zheng

Faculty Advisor: Dr. Sokratis Makrogiannis

ABSTRACT

The analysis and characterization of imaging patterns is a significant research area with several applications to biomedicine (computer-aided diagnosis), remote sensing (urban planning, environmental monitoring), homeland security (face recognition, object recognition, biometrics) social networking, and numerous other domains.

In this dissertation we study and develop mathematical methods and algorithms for disease diagnosis and tissue characterization. The central hypothesis is that we can predict the occurrence of diseases with certain level of confidence using supervised learning techniques that we apply to medical imaging datasets that include healthy and diseased subjects that can be used for training.

In the first stage of this work we propose to diagnose diseased patterns using texture characteristics that are derived from medical imaging modalities. The texture feature set consists of fractal dimension, local binary patterns, discrete wavelet frames, Gabor filters, discrete Fourier and Cosine Transforms, statistical co-occurrence indices, edge histogram, and Laws energy maps. Next, we implemented feature selection using correlation-based techniques to reduce the feature dimensionality. In the learning stage we employed bagging methods using fast decision tree learners, Random Forests, Bayes network, or naïve Bayes techniques. These techniques are also used for comparisons at the later stages of this work.

Next, we develop methods for calculation of sparse representations to classify imaging patterns and we explore the advantages of this technique over traditional texture-based classification. We also introduce integrative sparse classifier systems that utilize structural block decomposition to address difficulties caused by high dimensionality. We propose likelihood functions for classification and decision tuning strategies. These likelihood scores may also be used to determine a type of confidence interval for prediction.

The two application domains are osteoporosis diagnosis in radiographs of the calcaneus bone, and breast lesion characterization in mammograms. Both of these applications are very significant for improving public health. Osteoporosis results in deterioration of bone

quality and affects the quality of life of aging populations. Timely diagnosis of osteoporosis can effectively predict fracture risk and prevent the disease. Furthermore, breast cancer is one of the leading causes of death among women. Early detection and characterization of breast lesions is important for increasing the life expectancy and quality of health of women.

We performed bone osteoporosis classification experiments on the TCB challenge dataset and breast lesion characterization experiments Mammographic Image Analysis Society data set. In TCB there are 87 healthy and 87 osteoporotic subjects in the calcaneus trabecular bone. MIAS includes benign and malignant breast cancer lesions. In both of these two data sets, the scans of healthy and diseased subjects show little or no visual differences, and their density histograms have significant overlap.

In the experiments, our method of block-based sparse representation produced the best classification accuracy on these two datasets. We compared the conventional sparse representations classification (SRC) and texture-based methods with our method in a leave-one-out (LOO) cross-validation (CV) framework. The top texture-based classification performances are 67.8% ACC (classification accuracy) and 70.9% AUC (Area Under the Receiver Operating Curve) for bone characterization, and 63.4% ACC and 62.1% AUC for breast lesion characterization. The top performance of our integrative sparse model method by using a decision threshold equal to zero is 100% ACC and AUC for bone characterization by block-based maximum a posteriori sparsity-based (BBMAP-S) decision function, as well as 100% for bone characterization by block-based log likelihood sparsity-based (BBLL-S) decision function, 98.6% ACC and 97.8% AUC for breast lesion characterization by BBMAP-S decision function, and 100% ACC and 100% AUC for breast lesion characterization by BBLL-S decision function for breast lesion characterization.

We also used 10-fold and 30-fold cross-validation to evaluate the classification performances of our classification methods. The top rate of accuracy produced by the texture-based method is 66.7% and corresponding AUC is 67.5% for bone characterization using 10-fold cross-validation. Our method using integrative sparse models has obtained the highest ACC for 30-fold cross-validation is 69.33% and 70.2% with BBMAP-S decision function. It also achieved 70.7% ACC and 74.4% AUC with BBLL-S decision function for bone characterization. In 10-fold cross-validation experiments for bone characterization, BBLL-S produced 60.6% ACC and 62.5% AUC. In the breast lesion characterization application, the best performance over all the ROI sizes is 71.2% ACC and 69.8% AUC using texture-based methods and the conventional SRC method reached 55.0% ACC and 51.8% AUC using 10-fold cross-validation. For our system, the top performance is 86.7% ACC and 88.2% AUC for 30-fold experiments and 68.9% ACC and 73.7% AUC for 10-fold experiments.

Our results show that ensemble sparse representations of imaging patterns provide very good separation between groups of healthy and diseased subjects in two challenging diagnostic applications.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES AND ILLUSTRATIONS

LIST OF ABBREVIATIONS

ACC	Classification accuracy
AUC	Area under the ROC curve
BBLL	Block-based log likelihood decision function
BBLL-R	Block-based log likelihood approximation residual-based decision function
BBLL-S	Block-based log likelihood approximation sparsity-based decision function
BBMAP	Block-based maximum a posteriori decision function
BF	Best first search
BMD	Bone mineral density
BN	Bayesian network
BoK	Bag of keypoints
BP	Basis pursuit
CAD	Computer-aided diagnosis/detection
CC	Cranial caudal
CDF	Cumulative distribution function
CFS	Correlation-based feature selection
CL	Classifier
CT	Computed tomography
CV	Cross-validation
DCT	Discrete cosine transform
DDSM	Digital database for screening mammography
DFT	Discrete Fourier transform

DICOM	Digital imaging and communications in medicine
DXA	Dual energy X-ray absorptiometry
EH	Edge histogram
FD	Fractal dimension
FFDM	Full-field digital mammography
FSM	Feature selection method
GA	Genetic algorithm-based search
GLCM	Gray-level co-occurrence matrix
IG	Information gain
K-SVD	K - singular-value decomposition
LBP	Local binary patterns
LDA	Linear discriminant analysis
LLS	Likelihood score
LMS	Least mean square
LOO	Leave-one-out
LP	Linear Programming
LPP	Local preserving projection
MAP	Maximum a posteriori estimation
MIAS	Mammographic image analysis society digital mammographic database
MLE	Maximum likelihood estimation
MLO	Mediolateral-oblique
MLP	Multilayer perceptron
MP	Matching pursuit
MRI	Magnetic resonance imaging
NB	Naïve Bayes

NPE	Neighborhood preserving embedding
OMP	Orthogonal matching pursuit
PCA	Principal component analysis
PDS	Probability decision scores
QCQP	Quadratically constrained quadratic program
QP	Quadratic program
RF	Random forests
ROC	Receiver operating characteristic/curve
ROI	Region of interest
SCI	Sparsity concentration index
SOCP	Second order cone programming
SPP	Sparsity preserving projections
SQP	Sequential quadratic programming
SRC	Sparse representation classification
SVM	Support vector machine
TCB	Texture characterization of bone radiograph images
TIFF	Tagged image file format
TNR	True negative rate
TPR	True positive rate

Chapter I: INTRODUCTION

In this chapter we first introduce the fields of machine learning and pattern recognition including the main approaches, systems, stages and related fields. We also describe the goals, motivation, significance and contributions of this work. Finally, we provide an outline of the dissertation.

1.1 Machine Learning and Pattern Recognition

1.1.1 Machine Learning

Machine learning is an artificial intelligence (AI) technique that applies statistical learning methods to learn and identify objects from their measurements. One of the main goals of machine learning is to explore and develop methods for learning and creating models and rules that predict the state of new samples at sufficient accuracy levels using given input samples of known states.

Machine learning algorithms may train a model given the input data and use statistical techniques to yield a prediction score in a fixed range of numerical, categorical or other types of values. In other words, machine learning can create a model in order to automatically determine the state of test data.

One example is the decision tree learner, whose nodes process one variable at a time. One decision rule can be learned by a branch of the decision tree. To improve the accuracy of learning, more branches can be built for the decision tree corresponding to different types of input data. After the decision tree is created, new data samples can be given as input to this model for prediction.

Machine learning has developed into a multidisciplinary field in the past 30 years or so, involving concepts and techniques from probability theory, statistics, approximation theory, convex analysis, and computational complexity theory.

It has been widely used in medical diagnostics, computer vision, data mining and biometrics. One of the applications of machine learning is text classification [90], in which pool-based active learning with support vector machines (SVMs) has been performed. Diagnostic and prognostic prediction of neuroimaging measurements in psychiatry frequently employs machine learning classification techniques [65]. In addition, high diagnostic accuracy has been achieved in Alzheimer's disease by machine learning [21].

1.1.2 Pattern Recognition

Pattern recognition uses mathematical methods and algorithms to analyze patterns and to classify the patterns or related information. Patterns can be objects or signals which we aim to recognize, such as face, voices, fingerprints, diseases. We may consider machine learning and pattern recognition as two different facets of the same subject. Machine learning terminology is mostly used in computer science, while pattern recognition in engineering disciplines [8].

The more relevant the patterns that we select, the better decisions we can make [43]. Pattern recognition includes a training or learning stage, in which the model is created by learning from the input patterns. The training stage may be challenging in terms of representing the input patterns and also time consuming. Training is important since it affects the performance of the system. The training stage includes the pre-processing, feature selection and feature extraction stages as well.

Pattern recognition is widely applied to many cutting edge research areas. For example, face recognition is a widely studied topic. Face analysis requires the extraction of efficient descriptors. In [4] the authors introduced local binary pattern texture features extracted from local facial regions as local descriptors. A local descriptor has the advantage of robustness with respect to illumination and facial expression changes. Medical image analysis is another popular topic, where pattern recognition technique plays a very important role [59].

1.1.3 Classification and Regression

Machine learning and pattern recognition can be categorized by the desired output of a machine-learned system. One of these categories is classification, where the input data is split into two or more subsets of data. Then the learner creates a model by using one of few subsets from these subsets of data, and testing is applied to the unseen subset of data. Techniques of classification include Naive Bayes, entropy and support vector machines.

Another purpose is regression, where the output of regression problem consists of one or more continuous variables. Extreme learning machine can be applied for regression [42].

Classification and regression both are supervised problems. The error of classification and regression can be decomposed into a bias term and a variance term [43].

1.2 Machine Learning Paradigms

Supervised Learning

Supervised learning utilizes the input data and its labels –which is the desired output– to create a model and/or learn a decision function that is then applied to unlabeled data. In [47], the procedure of applying supervised machine learning has been described. First, the dataset is pre-processed because the collected data are not all informative and relevant, some are not available for induction, and may have been corrupted by noise. Some methods have been cited in [47] to deal with missing data, noise, and unavailable data for learning. Then feature selection helps to reduce the data dimensionality. The learning algorithm is the main step for the model creation. This is based on the problem domain to choose the algorithm, such as decision trees, Naive Bayes, Bayesian networks, SVM and so on. In the training stage learning algorithm is applied on the collected data, so the parameters of the learning algorithm can be tuned and cross-validation can be applied. The test data should not overlap with training data, but they are expected to have similar properties.

Unsupervised Learning

In unsupervised learning the class labels are not available, so the system does not know whether the classification results are correct or not. It analyzes the input data and finds the potential rules for classification by minimizing an objective function. A typical example of unsupervised machine learning is clustering. Clustering seeks similar characteristic features to group data samples that have no class labels. Therefore, a clustering algorithm usually only needs to know how to calculate the similarity. Clustering may be a component of other techniques such as artificial neural networks [1].

Semi-Supervised Learning

Semi-supervised learning utilizes a training set that includes the input data and some labels of the input data, therefore some of the output may be missing. It is considered a combination of supervised and unsupervised learning as there are labeled and unlabeled data samples. The data labeling procedures may be difficult and time consuming, therefore semi-supervised learning requires less human effort than supervised learning and may still yield high accuracy rates [105]. Semi-supervised learning has been applied to image processing, bioinformatics, and information retrieval [17]. In [17] the authors propose to apply unsupervised learning on all data first and then apply supervised learning to the labeled data only.

Reinforcement Learning

Reinforcement learning utilizes an incentives-or-punishments system and learns under stimulation from the system, resulting in habitual behavior that can maximize benefits. It is mostly used in operations research, cybernetics, etc. The difference with supervised machine learning is reinforcement learning does not require the correct input/output pairs.

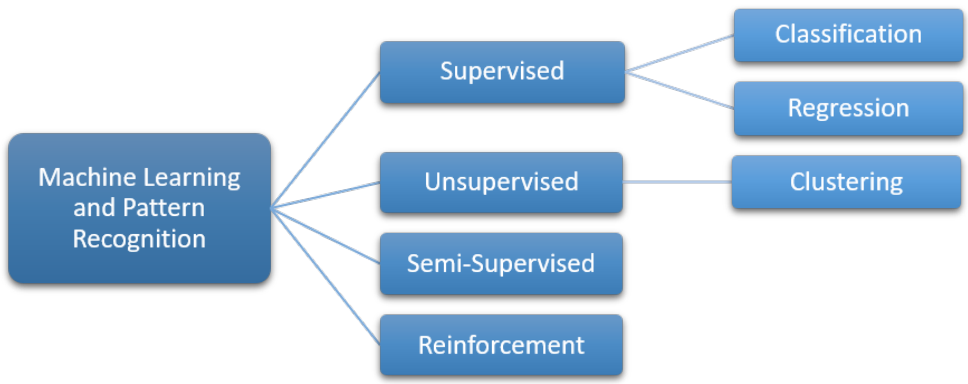

Figure 1.1: Machine learning and pattern recognition learning strategies.

1.3 Pattern Recognition Categories

Statistical

The statistical pattern recognition has been designed for many recognition systems. A pattern is represented by a set of d features that form a $d-$dimensional feature vector [43]. These methods use statistical approaches, such as estimation of probability distributions of patterns in each class, to determine the decision functions and the decision boundaries between classes.

The decision boundaries can also be determined by discriminant analysis methods. A discriminant function can be a linear, quadratic or other type of function. Based on the patterns, we can assume the type of discriminant function and find the best decision boundaries based on the classification of training patterns [43].

These systems also include a training and a classification stage. The training stage implements pre-processing, texture computation, feature selection/extraction and model learning. In the classification stage, test patterns are classified by the trained classifiers.

Syntactic

Syntactic pattern recognition uses the structure of patterns and focuses on the interrelationships between the primitives. The primitive is the simplest and elementary pattern such that more complex patterns are presented by these primitives. If we know the concept of the formal grammar, then we can design a syntax classifier based on the formal grammar.

1.4 Related Fields

1.4.1 Probability Theory

In pattern recognition, probability theory provides the foundation for building learning models, and expressing and analyzing uncertainty in knowledge [8]. We next introduce some probability theory principles, which we will be used in our methods.

Probability Distributions

A probability distribution models the probability of occurrence of values of one or more random variables. Among the different types of probability distributions, the Normal or Gaussian distribution is one of the most common types with wide applicability to pattern recognition and machine learning and is given by

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right] \tag{1.1}$$

where μ is the mean and σ is standard deviation. The μ and σ values can be used to quantify the data separation and analyze data properties.

The cumulative distribution function (CDF) is the probability that a real valued random variable X takes on a value not greater than x,

$$F_X(x) = P(X \leq x), \qquad P(a < X \leq b) = F_X(b) - F_X(a). \tag{1.2}$$

Multivariate Normal Density

The multivariate normal density has been investigated for a while, mainly because of its analytical tractability [26]. Given a continuous valued feature vector $\mathbf{x}$ for a given class

ω_i, we define general multivariate normal density as,

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{d/2}|\mathbf{\Sigma}|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^t\mathbf{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right] \tag{1.3}$$

where d is the dimensionality, $\mathbf{x}$ is a column vector $\in \mathbb{R}^d$, $\boldsymbol{\mu} \in \mathbb{R}^d$ is the mean vector , $\mathbf{\Sigma} \in \mathbb{R}^{d\times d}$ is covariance matrix.

It is frequently convenient to transform the multivariate normal distribution to a spherical one, that is having a covariance matrix proportional to the identity matrix $\mathbf{I}$. This is known as Whitening transformation and is given by [26],

$$\mathbf{A} = \mathbf{\Phi}\mathbf{\Lambda}^{-1/2} \tag{1.4}$$

where $\mathbf{\Phi}$ is a matrix that columns are orthonormal eigenvectors of $\mathbf{\Sigma}$, $\mathbf{\Lambda}$ is the diagonal matrix of the corresponding eigenvalues.

Density Estimation

Parametric Techniques - Maximum Likelihood Estimation

Maximum likelihood estimation (MLE) is method for estimating the parameters of a probabilistic model under the given observations. It is one of the methods that does not use the prior distributions for estimating. The observations and the probabilistic model define the properties of the parameters.

The maximum likelihood estimation can be defined as,

$$\widehat{\theta} = \{\arg\max_{\theta\in\Theta}\mathcal{L}(\theta; x)\} \tag{1.5}$$

where $\mathcal{L}(\theta; x)$ is the likelihood function, θ is a set of parameters for a specific distribution model $\{f(\cdot;\theta)|\theta \in \Theta\}$.

The natural logarithm of likelihood function is log-likelihood function. It is more convenient to use, since log-likelihood function is a strictly increasing function and can be used in maximum likelihood estimation and decision functions as well.

Nonparametric Techniques - Kernel Density Estimation

In contrast to parametric density estimation, nonparametric techniques can be used with any distributions and without knowing the underlying densities.

Kernel density estimation utilizes Parzen windows and is widely used in signal processing, statistics, and econometrics fields as discussed in [70, 79].

In the fundamental approach we first consider a region $\mathcal{R}_n$ that is a hypercube with dimension d and the length of this hypercube edge is denoted as h_n. The volume of this hypercube is $V_n = h_n^d$. We define another function k_n that returns the number of samples falling in this hypercube [26]

$$k_n = \sum_{i=2}^{n} \phi\left(\frac{\mathbf{x} - \mathbf{x}_i}{h_n}\right) \tag{1.6}$$

where $\phi(\mathbf{u})$ may be defined as a unit hypercube with origin at its center. $\phi(\mathbf{u})$ is 1 when $|u_j| \leq 1/2$, $j = 1, ..., d$, otherwise is 0. Hence, the estimated function is

$$p_n(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} \delta_n(\mathbf{x} - \mathbf{x}_i) \tag{1.7}$$

where $\delta_n(\mathbf{x}) = 1/V\phi(x/h_n)$. The value of h_n affects the amplitude and the width of δ_n. Small values of h_n generate smoothly varying estimates of $p_n(\mathbf{x})$. Another common estimator for $\phi(\mathbf{u})$ is the Gaussian kernel, or radial basis function.

1.4.2 Decision Theory

Probability theory provides the foundation for representing uncertainty in pattern recognition. Decision theory helps us determine the state of an unlabeled sample usually via probability theory tools.

Bayesian Theory

The Bayes' theorem plays an important role in pattern recognition and machine learning [8]. It provides the relationship between conditional probabilities of random variables and the distribution of marginal probability. The Bayesian formula may be used for introducing new evidence to modify the existing decision function.

$$P(\omega_j|x) = \frac{p(x|\omega_j)P(\omega_j)}{p(x)} \tag{1.8}$$

In the Bayes theorem, we normalize the product of the prior probability and the likelihood function, to yield the posterior probability. Prior probability $P(\omega_j)$ is the probability available before we know the state of object ω_j, the posterior probability is the probability that we know the state of ω_j after x has been measured. The likelihood function $p(x|\omega_j)$, expresses the likelihood of occurrence with different ω_j. The integral of likelihood of ω_j may not equal to one.

Discriminant Functions

Discriminant functions are functions that are designed for classifying patterns. The discriminant functions do not have to be unique, and we can multiply by same positive constant or shift them by same constant [26]. The discriminant function learns a function that maps into x and directly to the decision function [8].

The linear discriminant function is a linear combination of the components of $\mathbf{x}$ can be formulated as [26]

$$g(\mathbf{x}) = \mathbf{w}^t\mathbf{x} + w_0 \tag{1.9}$$

$$= r \parallel \mathbf{w} \parallel \tag{1.10}$$

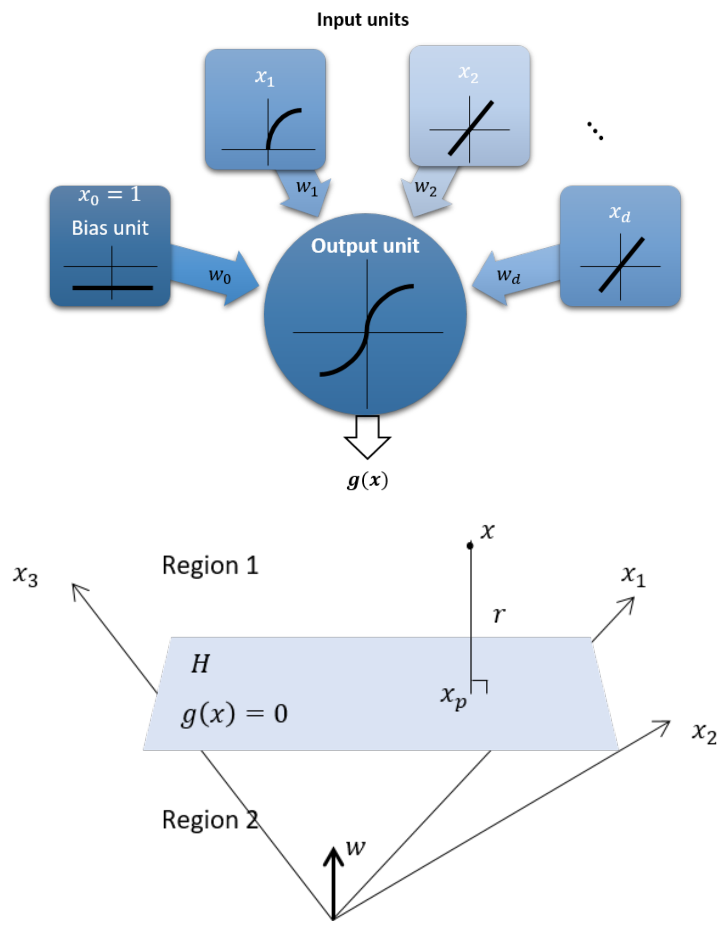

Figure 1.2: Linear discriminant function outline (left) and decision surface (right)

where $\mathbf{w}$ is a weight vector, w_0 is the bias or threshold weight, and r is the desired algebraic distance. We can express $\mathbf{x}$ using $\mathbf{x}_p$ that is the normal projection of $\mathbf{x}$ onto the hyperplane H which divides the feature space into different regions by

$$\mathbf{x} = \mathbf{x}_p + r \frac{\mathbf{w}}{\parallel \mathbf{w} \parallel} \tag{1.11}$$

Each component of $\mathbf{x}$ is an input and by obtaining the corresponding weight $\mathbf{w}$ and bias w_0, we calculate the output by the inner product as shown in Fig 1.2.

On the other hand, nonlinear discriminant functions can be step discriminant functions, quadratic, or of another type. The discriminant function of multivariate normal of (1.3) is

$$g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i)^t \boldsymbol{\Sigma}_i^{-1}(\mathbf{x} - \boldsymbol{\mu}_i) - \frac{d}{2}\ln 2\pi - \frac{1}{2}\ln|\boldsymbol{\Sigma_i}| + \ln P(\omega_i) \tag{1.12}$$

1.5 Main Stages of Machine Learning and Pattern Recognition

To implement a recognition system, we use the classifiers to perform classification on the data that need to be identified. Next, we describe the four main stages in a pattern recognition system: pre-processing, feature computation, feature extraction, and classification decision function.

Pre-processing

Pre-processing is an important step for pattern recognition as it can help to obtain the useful information from the input data. Pre-processing includes removing the noise from imaging data, filtering the irrelevant and redundant information, normalization, undersampling, and finding the region of interest (ROI).

Feature Calculation

This stage includes feature calculation and analysis. Different characteristics of the data can be represented based on the feature type. In image-based systems the computed features may be related to texture, appearance and shape. Texture features include fractal dimension, wavelet texture descriptors, Law's texture energy masks, discrete Fourier and cosine transforms and several others.

Feature Extraction

The data we use for pattern recognition usually lie in a high denominational feature space. In order to effectively implement classification, it is helpful to select the more relevant features for classification. Feature extraction and selection can reduce the complexity of data

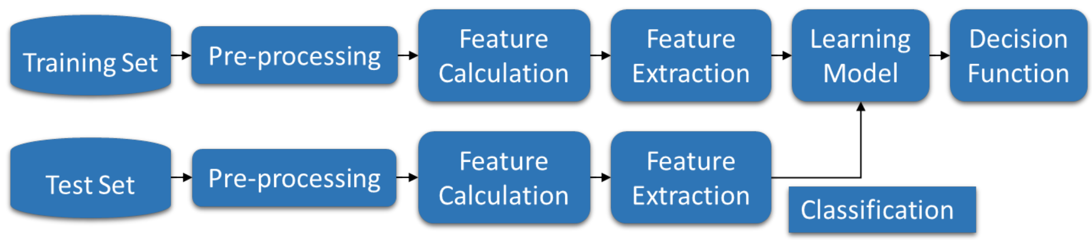

Figure 1.3: The pattern recognition process.

and save computational time.

Learning Decision Function/Model

The classification decision function can be learned by statistical, numerical or other methods to classify the objects into a category in the feature space. Certain rules of decision have been created based on the training set, then minimize the false rate of classification according to the decision function.

1.6 Classification Performance

1.6.1 Bayes Error Rate and Dimensionality

The minimum classification error obtained by Bayes decision classifiers is the Bayes Error Rate. This rate can be estimated by analytical or numerical methods and determines the separation capability of a classification system. The Bayes optimal decision boundary minimizes the probability of classification error. The two types of error are false positive and false negative. The Bayes decision boundary and the corresponding error rate are determined by the point of equal posterior probabilities for all classes.

For the two class multivariate normal case and equal prior probabilities the Bayes error rate is [26]

$$P(e) = \frac{1}{\sqrt{2\pi}} \int_{r/2}^{\infty} e^{-u^2/2} du \tag{1.13}$$

where $r^2 = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^t \boldsymbol{\Sigma}_i^{-1} (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$. Then $P(e) \to 0$ as $r \to \infty$.

If $\mathbf{\Sigma} = diag(\sigma_1^2, ..., \sigma_d^2)$, then we have

$$r^2 = \sum_{i=1}^{d} \left(\frac{\mu_{i1} - \mu_{i2}}{\sigma_i} \right)^2. \tag{1.14}$$

The Eq. (1.14) shows that each additional feature will increase r^2 and decrease $P(e)$. More useful features have the property of big differences between the class-conditional means and small standard deviations within each class.

On the other hand, the increased dimensionality of features may cause difficulties in classification. One consideration is the curse of dimensionality that sets requirements for increased number of samples for each additional feature in order to estimate the likelihood and discriminant functions. Given a fixed number of training samples, increased dimensionality implies sparser points. Overfitting is a problem that becomes more substantial in sparse feature spaces.

An overfitting classifier uses a complex model to explain the property of the data points, although the true underlying decision surface may be approximated by lower order. This may happen when the training set's size is small in comparison with the feature dimensionality and because the measured data points may contain errors and random noise. Overfitting may result in decreased predictive capability in the testing stage. For example, the model can have high order terms whereas the problem can be learned by a linear function model as in Fig. 1.4.

Another important problem is that the computational complexity of the classifier increases with the number of dimensions. We use O and Θ for computational complexity. For example, in Eq. (1.12), assuming $n > d$, for each individual components, we have

$$g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x} - \underbrace{\widehat{\boldsymbol{\mu}}_i}_{O(dn)})^t \underbrace{\widehat{\mathbf{\Sigma}}_i^{-1}}_{O(nd^3)} (\mathbf{x} - \boldsymbol{\mu}_i) - \underbrace{\frac{d}{2} \ln 2\pi}_{O(1)} - \underbrace{\frac{1}{2} \ln |\widehat{\mathbf{\Sigma}_i}|}_{O(d^3)} + \underbrace{\ln P(\omega_i)}_{O(n)} \tag{1.15}$$

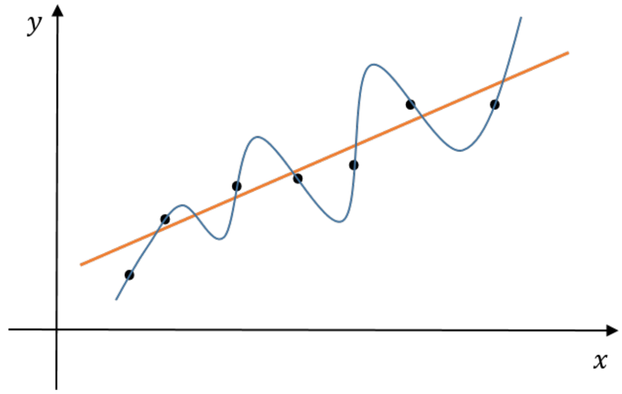

Figure 1.4: Blue line is overfitting, orange line is decision function line.

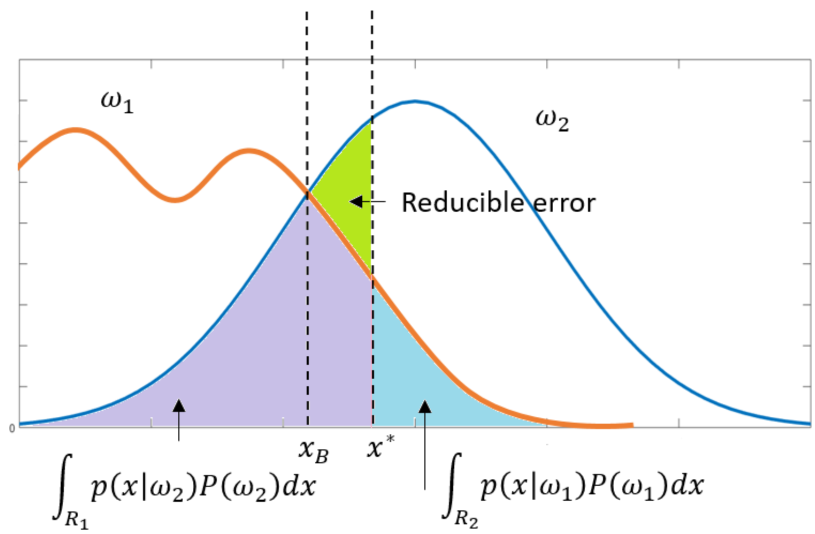

Figure 1.5: Selection of optimal Bayes criterion based on error minimization.

Then the computational complexity for the Bayes classifier is $O(cd^2n)$. Given the computational complexity of each equation, we can estimate the complexity of the problem and find the most efficient way to solve the problem.

1.6.2 Bias and Variance

There are two main approaches for measuring the matches between the classification problem and the learning algorithm; bias and variance. The lower the bias or the variance, the better match the learning algorithm produces. Also, bias and variance are related [23].

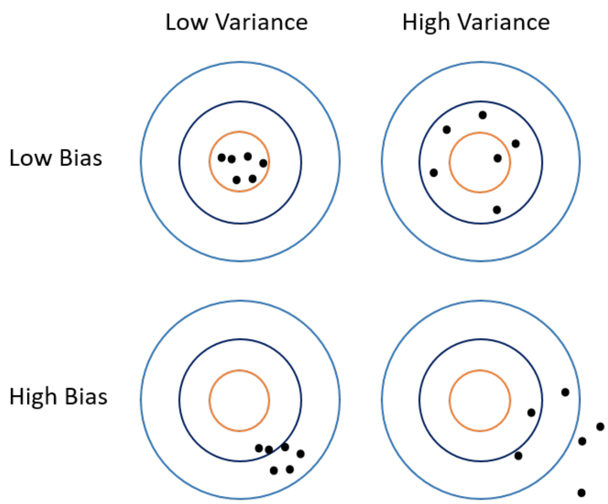

Figure 1.6: Example of effect of bias and variance on classification/regression output.

Bias was discussed in [61] as a criterion for choosing one generalization over another. Bias expresses the deviation of all measurements from the true value, includes the inaccuracy of measuring instruments, not enough features, the design of experiments, etc. In the biomedical research related fields, bias quantifies the systematic errors caused by design, implementation, data processing and analysis, and the interpretation and inference of results. Bias can not be avoided, but more features and multiple sampling can help to reduce the bias.

Variance describes the fluctuation of classification performance with respect to variation in the training data. Therefore, the greater the variance is, the greater the fluctuation of the error is; the smaller of the variance, the smaller of the fluctuation of the error. Such as in Fig. 1.6. In general a model with many parameters may achieve low bias but high variance. Conversely, a model with few parameters may not fit the data very accurately but the model will not change very much for diffferent training datasets.

The mean-square error is the average over all training sets D with a fixed size n [26],

$$\varepsilon_D[(g(\mathbf{x};D)-F(\mathbf{x}))^2] = \underbrace{(\varepsilon_D[g(\mathbf{x};D)-F(\mathbf{x})])^2}_{\text{bias}^2} + \underbrace{\varepsilon_D[(g(\mathbf{x};D)-\varepsilon_D[g(\mathbf{x};D)])^2]}_{\text{variance}} \tag{1.16}$$

where $F(\mathbf{x})$ is the true function, and $g(\mathbf{x};D)$ is the estimated regression function. The prior information may help to achieve low bias and variance for regression.

The classification error rate is given by [26]

$$\Pr[g(\mathbf{x};D) \neq y_B] = \Phi\left[\mathrm{Sgn}[F(\mathbf{x})-1/2]\frac{\varepsilon_D[g(\mathbf{x};D)]-1/2}{\sqrt{\mathrm{Var}[g(\mathbf{x};D)]}}\right] \tag{1.17}$$

$$= \Phi\left[\underbrace{\mathrm{Sgn}[F(\mathbf{x})-1/2][\varepsilon_D[g(\mathbf{x};D)]-1/2]}_{\text{boundary bias}}\underbrace{\mathrm{Var}[g(\mathbf{x};D)]^{-1/2}}_{\text{variance}}\right] \tag{1.18}$$

where

$$\Phi[t] = \frac{1}{\sqrt{2\pi}}\int_t^\infty e^{-1/2u^2}du = \frac{1}{2}[1-\mathrm{erf}(t/\sqrt{2})] \tag{1.19}$$

$$\Pr[g(\mathbf{x};D)=y] = \Pr[y_B(\mathbf{x}))=y] = \min\left[F(\mathbf{x}), 1-F(\mathbf{x})\right] \tag{1.20}$$

From Eq. (1.18), the variance is affected by the sign of the boundary bias, so the low variance is important for classification accuracy rate, but bias is not need be. Some types of bias can be reduced by low variance, and this can significantly reduce the effects of biases associated with simple estimators such as Nave Bayes [33].

More parameters in g can decrease classification bias and increased n may decrease the variance decrease. In many cases, bias and variance grow in the opposite way, low bias with high variance and high bias with low variance. To reach low generalization error, achieving low variance is more effective than achieving low bias. The prior information of $F(\mathbf{x})$ and large n are effective for low bias and variance, since the larger n we have, the more new parameters in model g can be added.

1.6.3 Classification Performance Measures

We can measure the classification performance by calculating the true positive rate (TPR), true negative rate (TNR), classification accuracy (ACC), and area under the ROC curve (AUC).

Receiver Operating Characteristic (ROC) Curve

The ROC curve is a graph of the true positive rate (TPR) versus the false positive rate (FPR). We can utilize the receiver operating characteristic (ROC) curve to test the classifier's performance because when we make decisions, the ROC curve is not affected by the cost and benefit of the model, and gives objective and neutral performance results. When there are many trials, the probabilities can be determined, and the false and hit rates as well. We use these hit and false rates to plot a 2-D graph that enables us to select the best detection model and set the optimal threshold in the same model. Different decision thresholds will produce points on the curve, because the hit and false rates will change as the threshold changes.

1.6.4 Cross Validation

Cross validation is a practical method for statistically separating a data sample into smaller subsets. Some of the subsets are called training sets, and the remaining subsets are called test sets. The training set is used for analysis and finding the model parameters, while the test set is used for confirmation and verification of this analysis. The cross validation method for the training set aims to reduce problems such as over-fitting.

A generalization of cross validation is k-fold cross validation, $k = 1, 2, ...$ We divide data set into k subsets, where one of these subsets is test set and the remaining $k-1$ subsets are training sets. The cross-validation is repeated k times, then each subset is verified once.

Estimates can be obtained by the average of these k times, or other combinations methods. The advantage of this method is that it can randomly generate subsets of samples and can be repeated for training and testing. 10-fold cross-validations and leave one out cross validation ($k = 1$) are mostly used.

1.7 Computer-Aided Diagnosis and Tissue Characterization

Pattern recognition supports the development of computer-aided detection/diagnosis (CAD) systems. A CAD system can localize, delineate and label structures like tumors or lesions. The research fields of computer-aided tissue characterization, diagnosis, and prognosis have gained significant interest in the past few decades [7, 85]. These techniques combine concepts from image analysis, pattern recognition and machine learning to separate diseased from healthy subjects. Applications span a wide range of clinical areas and diseases such as detection of microcalcifications in mammography screening systems [66, 50], early diagnosis of Alzheimer [76, 55], cancer [29, 81], soft and hard tissue characterization for age-related diseases [49, 27, 68], and cardiovascular diseases. This popularity is mainly attributed to the potential for timely characterization of tissues that may reduce the mortality rate from diseases. Frequently, these automated diagnostic systems extract information from medical imaging modalities such as MRI and CT scans to produce a binary decision or a likelihood score that characterizes the state of a lesion as healthy or diseased. Sometimes multiclass classification may be needed to characterize different lesion types as in cancer applications. An interesting recent research topic in the CAD field studies the application to multiple databases of a CAD system that has been trained on a single database. For example, the goal would be to train a breast lesion CAD system on a single mammography database and use it for diagnosis of breast lesions on other mammography databases.

1.7.1 Breast Cancer

Breast cancer is one of the leading causes of death among women [29]. The cells in the breast start to grow out of control. Breast cancer can be diagnosed on an x-ray, or a lump which can be felt. If the tumor is malignant, the cancer cells may spread into surrounding tissues, blood, or the lymph system. The breast cancer can start anywhere in the breast, but mostly starts from the ducts which carry milk to the nipple (ductal cancers). The breast cancer does not always cause a lump that an expert can feel, therefore many of the breast cancers are found on screening mammograms. If the breast cancer can be diagnosed early, when it is small and has not spread, it can be treated successfully. Mammograms can help to find breast cancer at an early stage. Because of its significance, the research area of CAD systems for breast cancer is very popular [40, 66, 67, 93, 58, 63, 48, 73, 64].

There are some widely used mammographic databases, for example mammographic image analysis society digital mammogram database (MIAS) [88], digital database for screening mammography (DDSM), Trueta, and BancoWeb [63]. The MIAS database consists of 322 digitized MLO images with 68 benign, 51 malign lesions and 203 normal images. DDSM contains 10,480 LJPEG images, benign and malignant lesions and normal images from two views (CC and MLO) of each breast. BancoWeb is a new database that was made public in 2010. It contains 320 cases and 1473 TIFF images in CC and MLO views. BancoWeb includes more information about patients and annotations than other older databases.

1.7.2 Osteoporosis

Osteoporosis is a skeletal disorder characterized by decreased bone strength that may lead to susceptibility of fracture [7]. Osteoporosis has been operatively defined on the basis of bone mineral density (BMD) [96], in [95, 46] has described the criteria of osteoporosis, the T-score less than 2.5 standard deviations, that is a BMD lies on 2.5 or lower than the average of young healthy women. The most used technique to measure BMD is dual energy X-ray absorptionmetry (DXA), and the development of pharmaceutical interventions in osteoporosis is based on T-score for BMD [28, 32, 38]. There are many researchers working on diagonosis of osteoporosis, such as [91, 69].

One of the osteoporosis datasets is provided by the TCB challenge. Many published works in the literature have proposed analysis and diagnosis of osteoporosis. The authors in [84] applied histogram, gray-level co-occurrence matrix (GLCM) and principal component analysis (PCA) analysis to compute and extract texture chracteristics and used support vector machines (SVM) as classifier. In [69], the anisotropic discrete dual-tree wavelet transform was proposed for texture computation and SVM for classification.

1.8 Thesis Outline

1.8.1 Topic and Goals

In this dissertation we study and develop mathematical methods and algorithms for computer aided-diagnosis. The two application domains are osteoporosis diagnosis in radiographs of the calcaneus bone, and breast lesion characterization in mammograms.

These two applications are both related to the quality of human's life and the risk of death, so early detection and characterization is very important for preventing deaths. While automated diagnosis in both applications is very challenging since scans of healthy and diseased subjects show little or no visual differences, and their density histograms have significant overlap.

We have proposed a system that is using pattern classification and machine learning for CAD (computer-aided detection/diagnosis) systems. We will explore the use of sparse modeling and classification for classifying diseased from healthy subjects. Then we will propose ensemble sparse techniques to find more accurate solutions than individual classification techniques. We will also develop and test other classification techniques based on texture features, or patch-based techniques such as the Bag of Keypoints.

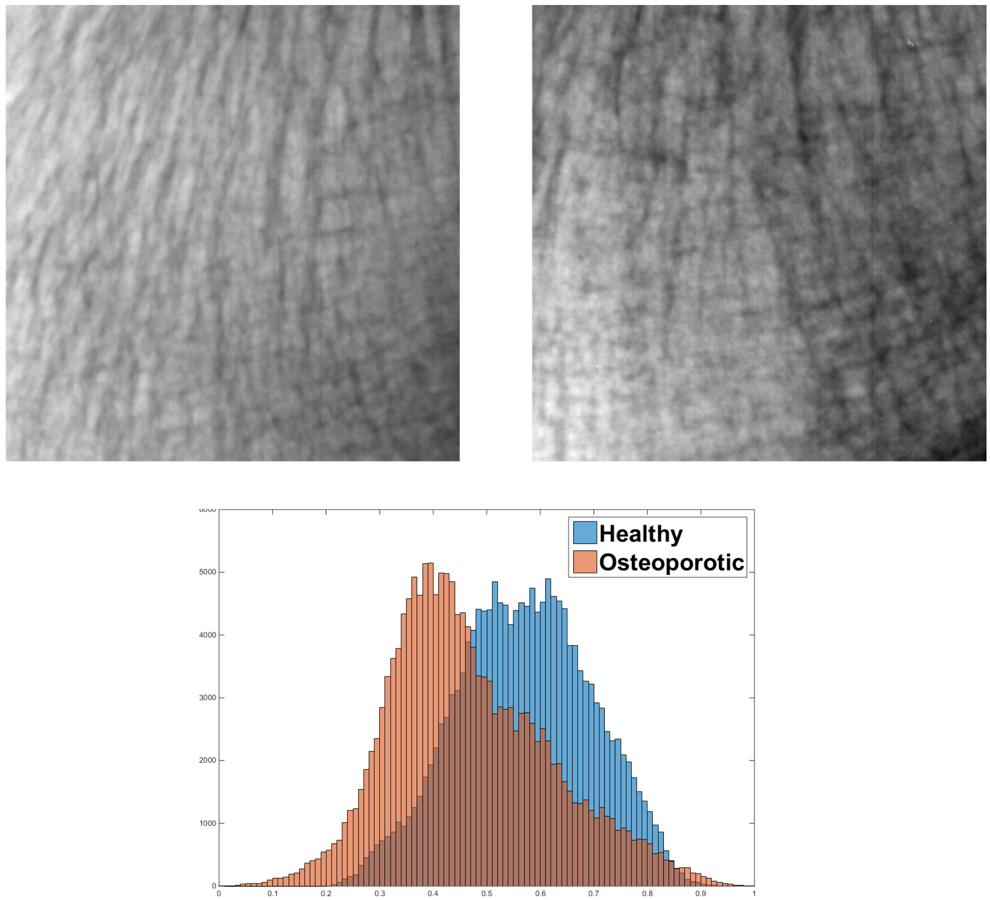

Figure 1.7: Example of textures of a control subject (top left) and a subject with osteoporosis (top right). The histograms of these two scans overlap significantly therefore rendering the diagnosis a challenging task (bottom).

1.8.2 Background and Motivation

The osteoporosis and breast lesion applications are very significant for improving public health. There are more than 3 millions of people diagnosed with osteoporosis in the U.S. per year. The risk is increasing with age, especially the people who are over 40+. Osteoporosis results in deterioration of bone quality and affects the quality of life of aging populations. Timely diagnosis of osteoporosis can effectively predict fracture risk and prevent the disease.

Furthermore, breast cancer is one of the leading causes for women. More than 200,000 new population per year in the U.S. have the disease. It also has higher risk with increasing age. The treatment depends on the stage of the cancer, surgery may needed if it diagnosis is late. So early detection and characterization of breast lesions are important for increasing the life expectancy and quality of health of women.

Another reason for the popularity of this research topic is also the significant overlap between the histograms of disease and healthy imaging patterns. Classification of these two datasets is a hard problem, therefore a potential solution of this problem will have a significant impact on related classification and recognition applications.

1.8.3 Dissertation Structure

In Chapter 2, we describe benchmark classification systems that we developed in the early stages of this work. They include a texture based classification system and the Bag of Keypoints method that utilizes patches. Then we introduce sparse classification methods and related mathematical programming problems and solvers in Chapter 3. The integrative sparse representation classification based system that we proposed is presented in Chapter 4, as well as the decision functions which are related to our proposed classification system. In Chapters 5 and 6 we discuss the experiments and results obtained by our system for osteoporosis diagnosis and breast lesion characterization and we compare these results with other methods described in Chapters 2 and 3. In Chapter 7 we summarize the methods that we developed and the main findings of this work.

1.8.4 Points of Contribution

In the first part, we explore texture based characteristics for separating diseased form healthy subjects. In the feature computation stage we have studied and implemented fractal dimension, local binary patterns, discrete wavelet frames, Gabor filters, discrete Fourier and Cosine Transforms, statistical co-occurrence indices, edge histogram, and Laws energy maps. We select the more relevant features from the features what we obtained. The feature extraction can help reduce the dimensionality and the computational time. The classification techniques includes Random Forests, Bayes network, or naïve Bayes techniques and Bagging.

The main contribution of this work is related to the development and evaluation of sparse representation based methods for classification. We show that sparse representation and classification may be more advantageous than the texture based technique for specific problems. Then we propose a block-based sparse representation method that uses a spatial block decomposition methodology for training an ensemble of classifiers to address irregularities of the approximation problem. Based on the sparse representation method, we divide the image into blocks, and develop three decision functions: maximum a posterior decision function, log likelihood score-based decision function and log sparsity decision function. Also, we propose methods for setting thresholds for decision functions using minimum Bayes error criteria. We compare the conventional sparse representation classification and texture-based methods with the block based sparse representation technique. The significant improvement of the classification for bone characterization and breast lesion characterization will be discussed in detail.

Chapter II: NON-SPARSE CLASSIFICATION TECHNIQUES: TEXTURE-BASED AND PATCH-BASED

Here we introduce our texture feature method for computer-aided diagnosis of diseased and healthy subjects. Our premise is that the deterioration of disease can be captured by textural features. We first computed texture features based on wavelet decomposition, discrete Fourier and Cosine transforms, fractal dimension, statistical co-occurrence indices, and structural texture descriptors. We employed feature selection techniques that consider the individual feature predictive ability and inter-feature redundancy to find the most discriminant feature set. In the classification stage we employed Naïve Bayes, Multilayer Perceptron, Bayes Network, Random Forests and Bagging models for diagnosis.

2.1 Introduction to Texture-based Classification

Texture is an image property that can be used for segmenting and classifying images into different objects. We can define a texture as a structure consisting of a group of related elements [86]. The pixels in this group are called texture primitives or texture elements, also called texels sometimes.

Texture analysis techniques are mainly applied to texture recognition and texture based shape analysis [86]. Generally, people consider texture as fine when the texture element is small and there are large differences between element, and coarse when the element is large and only few element in the image, grained and smooth, etc. For scientific applications of texture, we use more precise feature such as tone and structure [37]. Tone is more about pixel intensity and structure is about the spatial relationship between texture elements.

Statistical and syntactic approaches are employed for texture description. Statistical approaches compute the properties of texture. Syntactic approaches are good for the elements that have been labeled, then elements can be described by their properties.

There are many methods for texture extraction, such as wavelet analysis, Gabor filters and discrete cosine transform(DCT). The distributional based multivariate methods will be introducted, WW-test [34] and Kantorovich Wasserstein distance [35]. In [74], the authors present a patch based method and applied the multivariate WW-test and KWass techniques, also compared with wavelet, DCT and Gabor methods. The multivariate WW-test is based on the WW-test, WW-test can applied to hypothesis $\mathbf{H}_0$ to test is there are any two multi-dimensional point samples from same multivariate distribution [74] by method MST-graph [100]. The multivariate WW-test is defined [74] as,

$$W = \frac{R - E[R]}{\sqrt{Var[R|C]}} \tag{2.1}$$

where R is test statistic obtained of disjoint subtrees, and $E[R]$ and $Var[R|C]$ is given in [34]. Kantorovich-Wasserstein distance (KWass) is the distance between two stochastic distributions [74],

$$d_w(\mu, v) = \inf_j \{\mathbf{E}[d(X,Y)] : L(X) = \mu, L(Y) = v\} \tag{2.2}$$

where X and Y are discrete distributions, the infimum is taken of all the joint distributions with marginals μ, v [35].

2.2 Calculation of Texture Features

In this stage we compute texture descriptors that can be used to form morphometric signatures for separation between groups of healthy and disease subjects. This is usually performed in a high-dimensional feature space to reduce the Bayes error rate as explained in Chapter 1. We describe our feature set next.

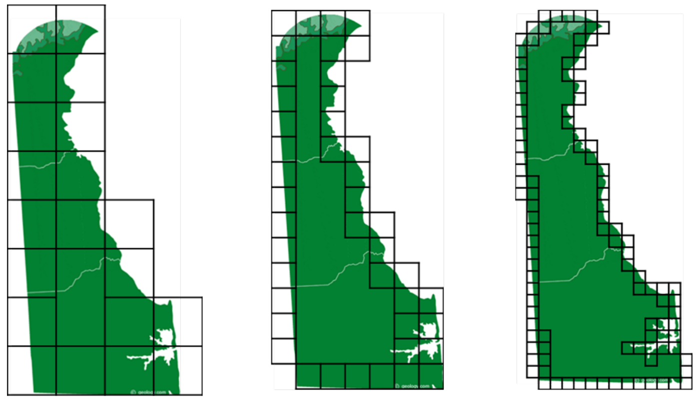

Figure 2.1: Box counting to compute the fractal dimension of Delaware state boundary.

2.2.1 Fractal Dimension

We computed Fractal Dimension attributes that have shown promise in texture classification applications. A fractal is defined as a mathematical set whose Hausdorff dimension exceeds the fractal's topological dimension [72]. It has been shown that fractal dimension correlates well with a function's roughness. Therefore, we used fractal dimension to measure the roughness and granularity of the image intensity function. The topological dimension of this function is equal to 3, consisting of 2 spatial dimensions plus the intensity.

We utilized the method of box counting to compute the fractal dimension explained as follows. Assuming a fractal structure with dimension D, we let $N(\epsilon)$ be the number of non-empty boxes of size ϵ required to cover the fractal support. Using the relation $N(\epsilon) \simeq \epsilon^{(-D)}$, we can numerically estimate D from

$$D = \lim_{\epsilon \to 0} \frac{\log N(\epsilon)}{-\log \epsilon} \tag{2.3}$$

by least squares fitting.

For the case of grayscale images or continuous functions, we generated 8 binary sets using multiple Otsu thresholding, then computed the fractal dimension, area, and mean intensity for each point set as in [19].

2.2.2 Wavelet Texture Descriptors

A multi-scale texture descriptor is usually very useful for classification. Gabor and wavelet transforms are both multi-scale spatial-spatial frequency filtering techniques. The discrete wavelet transform is frequently applied using tree or pyramid hierarchies for texture representation. Multi-band analysis offers advantages over the traditional discrete Fourier transform, but wavelet transform does not produce as exact a result as the Fourier transform.

Discrete Wavelet Frames

Discrete wavelet frames employ a filter bank for multi-scale decomposition. The Haar wavelet with a low-pass filter

$$H(z) = (1 + z)/2 \tag{2.4}$$

and a corresponding high-pass filter

$$G(z) = (z - 1)/2 \tag{2.5}$$

is frequently used because of its efficiency and computational simplicity.

The largest filter kernels will have size 2^{maxlevel}, where the *maxlevel* is the number of multiresolution levels. At each level, we filter the image by using the filter combinations:

$$H_x H_y,\ H_x G_y,\ G_x H_y,\ G_x G_y, \tag{2.6}$$

where H_x is the low-pass filter along the x direction, and G_y is the high-pass filter along the y direction.

To produce the wavelet frame representation we compute the discrete wavelet transform for all possible signal shifts at multiple scales. The filters are used to decompose the image in subbands. We compute the orthogonal projections and residuals for a full discrete wavelet expansion. We then compute energy, variance, entropy, contrast, skewness, and kurtosis signatures to form the texture descriptor. These characteristics are calculated as follows.

Contrast

Contrast measures the intensity contrast between a pixel and its neighbor of an image, in the range $[0 \quad (size(\text{Method},1)-1)^2]$, the formula is presented as,

$$\sum_{i,j} |i-j|^2 p(i,j) \tag{2.7}$$

Energy

Returns the sum of squared elements in $[0 \quad 1]$,

$$\sum_{i,j} p(i,j)^2 \tag{2.8}$$

Skewness

Skewness is a measure of the lack of symmetry. For a random variable x, the skewness is the third standardized moment γ_1 [11],

$$\gamma_1 = \mathrm{E}\left[\left(\frac{X-\mu}{\sigma}\right)^3\right] = \frac{\mu_3}{\sigma_3} = \frac{\mathrm{E}[(X-\mu)^3]}{(E[(X-\mu)^2])^{3/2}} = \frac{\kappa_3}{\kappa_2^{3/2}} \tag{2.9}$$

where μ is mean, σ is standard deviation, μ_3 is central moment, E is expectation operator and κ_i is the ith cumulants.

Kurtosis

Kurtosis measures the heavy-tailed or light-tailed of data in a normal distribution. If kurtosis is high, then it has heavy-tail.

$$\mathrm{Kurt}[X] = E\left[\left(\frac{X-\mu}{\sigma}\right)^4\right] = \frac{mu_4}{\sigma^4} = \frac{\mathrm{E}[(X-\mu)^4]}{(\mathrm{E}[(X-\mu)^2])^2} \tag{2.10}$$

Entropy

Entropy presents the state of a system, such as the disorder and randomness of the system. The changes of entropy of this system is determined by the initial states and final states of the entropy. The wavelet entropy is defined in [9],

$$S(p) = -\sum_{j<0} p_j \dot{\ln} p_j \tag{2.11}$$

Wavelet Gabor Filter Bank

The Gabor filter is a linear filter that can extract relevant characteristics for multiple frequencies and orientations (Fig. 2.2), similarly to the human visual system.

Gabor functions form a complete but non-orthogonal basis. In the spatial domain, a 2D Gabor filter is a Gaussian kernel function modulated by a sinusoidal plane wave. Gabor filters are often used for texture identification, and good results have been achieved. The filter has a real and an imaginary component representing orthogonal directions,

$$\text{Complex:} \qquad g(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \exp\left(i(2\pi\frac{x'}{\lambda} + \psi)\right) \tag{2.12}$$

$$\text{Real:} \qquad g(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \cos\left(2\pi\frac{x'}{\lambda} + \psi\right) \tag{2.13}$$

$$\text{Imaginary:} \qquad g(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \sin\left(2\pi\frac{x'}{\lambda} + \psi\right) \tag{2.14}$$

and

$$x' = x\cos\theta + y\sin\theta \tag{2.15}$$

$$y' = -x\sin\theta + y\cos\theta \tag{2.16}$$

where λ is wavelength of the sinusoidal factor, θ is orientation of the normal to the parallel stripes of a Gabor function, ψ is phase offset, σ is standard deviation of the Gaussian envelope and γ is spatial aspect ratio.

The filter dictionary can be produced by dilations and rotations of the mother Gabor wavelet.

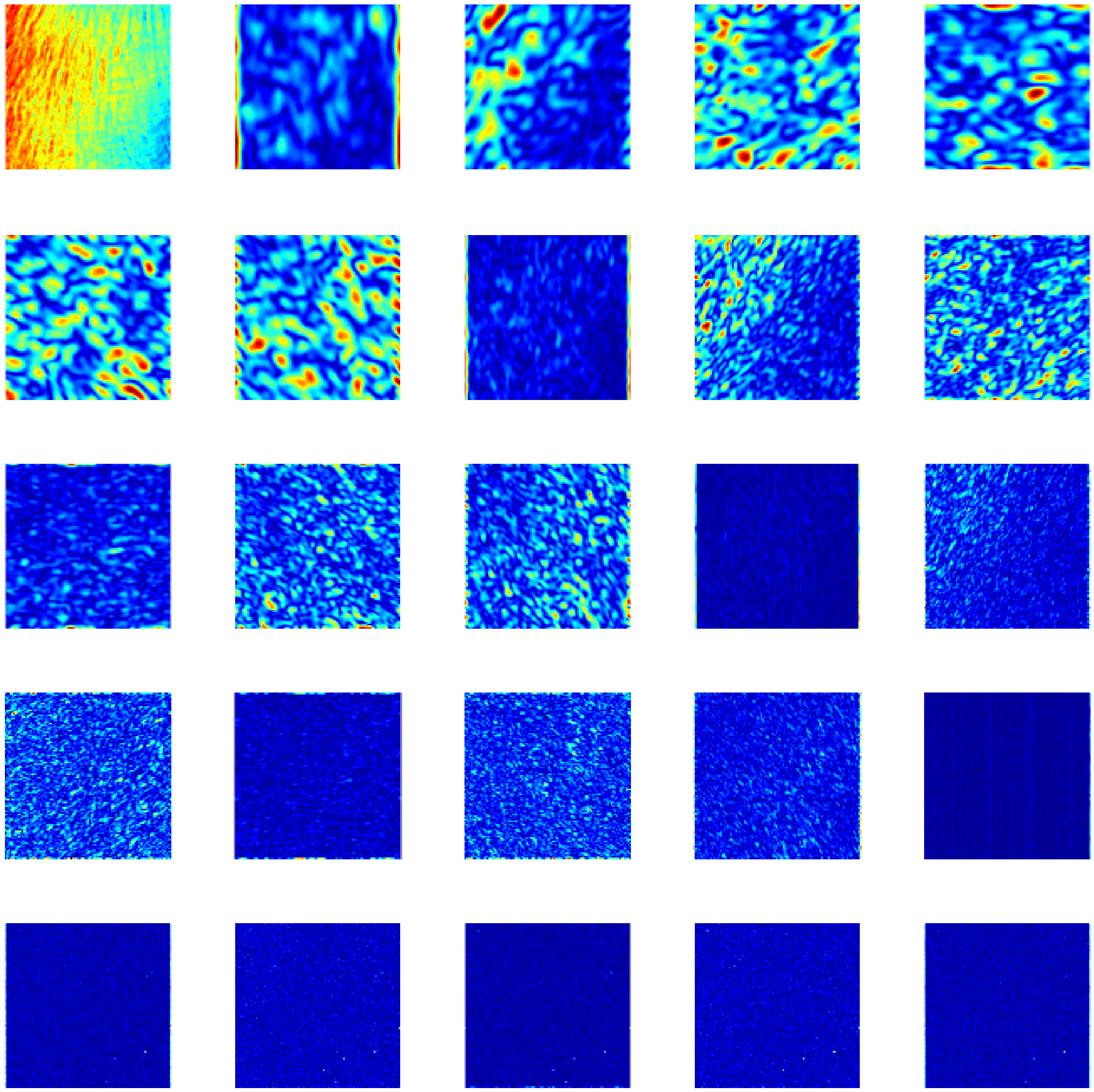

Figure 2.2: The original bone radiograph and the Gabor texture components of a healthy subject using 4 scales and 6 orientations. The 24 components are calculated using the mother wavelet function by using the original image (top left). While these maps pronounce the texture characteristics, visual interpretation is still particularly challenging. Therefore a machine learning technique is needed to distinguish healthy from osteoporotic subjects.

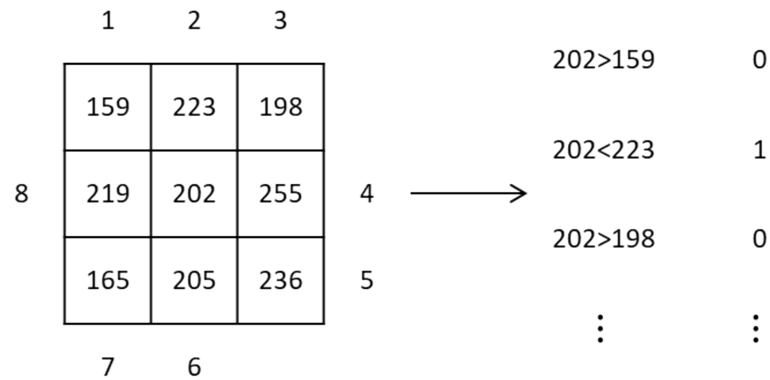

Figure 2.3: Process used to create the LBP

2.2.3 Local Binary Patterns (LBP)

For each pixel pix in the image, we compare the intensity of pix to the intensities of its eight neighbors. If the intensity of pix is greater or equal to its ith (where $i = 1, 2, ..., 8$) neighbor, we set $b_i = 0$, otherwise $b_i = 1$. From these eight neighbors we construct an eight-digit binary number $b_1b_2b_3b_4b_5b_6b_7b_8$. We use the histogram of these numbers as a texture descriptor [83]. Fig.2.3 shows the process of local binary patterns, the LBP = 01011101 = 93.

If a images with size $p \times q$, the LBP matrix is computed as $p - 2 \times q - 2$, and copy the boundary of the LBP matrix add it to the boundary of the LBP matrix, then the LBP matrix will be $p \times q$.

2.2.4 Discrete Fourier and Cosine Transforms

We utilize discrete Fourier transform and the discrete Cosine transform coefficients to capture spectral characteristics of texture. For example, fine texture has greater high frequency components, whereas coarse texture is represented by lower frequencies. The discrete Fourier and Cosine transforms are defined as follows, Discrete Fourier transform (DFT):

$$F(k,l) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m,n) \cdot e^{-j2\pi(\frac{mk}{M} + \frac{nl}{N})}, \tag{2.17}$$

where $k = 0, 1, 2, ..., N-1, l = 0, 1, 2, ..., M-1$.

Discrete Cosine Transform (DCT) uses only cosine basis functions:

$$C(k,l) = \sqrt{\frac{\alpha}{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} c_{mn} \cos \frac{\pi(2m+1)k}{2M} \cos \frac{\pi(2n+1)l}{2N}, \tag{2.18}$$

where $\alpha = 1, \text{if } k = l = 0; \alpha = 4, \text{if } 1 \leq k \leq M-1, 1 \leq l \leq N-1$.

We use the 8×8 coefficients corresponding to lower frequencies for classification.

2.2.5 Law's Texture Energy Masks

The texture energy is computed by a set of 5×5 convolution masks (level, edges, waves, spots, and ripples) to measure the amount of variation within a fixed-size window. We use the average level (intensity) feature to normalize intensity range and then we use the remaining 24 components to form the texture vector, as in Fig. 2.4. Next, we calculate the mean, variance, energy, skewness, kurtosis, and entropy for each component.

Level	$L5 = [\;1 \quad 4 \quad 6 \quad 4 \quad 1]$
Edge	$E5 = [-1 \; -2 \quad 0 \quad 2 \quad 1]$
Spot	$S5 = [-1 \quad 0 \quad 2 \quad 0 \; -1]$
Wave	$W5 = [-1 \quad 2 \quad 0 \; -2 \quad 1]$
Ripple	$R5 = [\;1 \; -4 \quad 6 \; -4 \quad 1]$

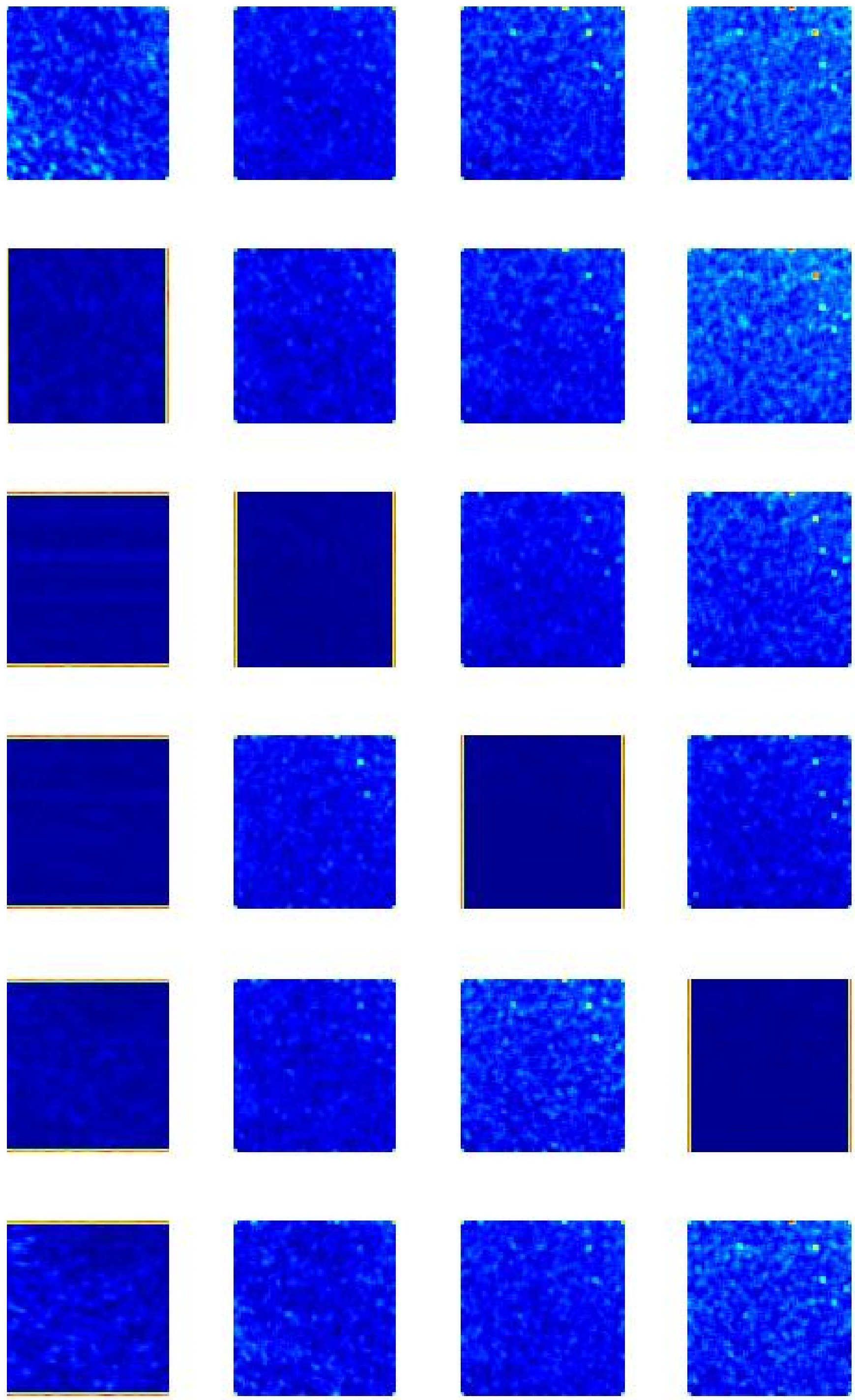

Figure 2.4: Law's Texture Energy Masks of a healthy subject calculated from a bone radiograph.

2.2.6 Edge Histogram

We compute the intensity gradient magnitude $|\nabla f|$ and then calculate its histogram by

$$p_{|\nabla f|}\left(|\nabla f| = r_k\right) = \frac{n_k}{N}, \quad k = 0, ..., L-1 \tag{2.19}$$

$$\nabla f = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \cdots, \frac{\partial f}{\partial x_N}\right)^{\top} \tag{2.20}$$

2.2.7 Gray Level Co-Occurrence Matrix (GLCM)

The GLCM calculates how the frequency of occurence of gray-level pairs (i, j) in horizontal, vertical, or diagonal pixel adjacencies on the image plane, shows in Fig.2.5 left. Horizontal ($0°$), vertical ($90°$), and diagonal ($-45°, -135°$) dimensions of analysis are denoted by P_0, P_{90}, P_{45}, and P_{135}, respectively. After we create the GLCMs, we compute contrast, correlation, energy and homogeneity measures.

The offset of GLCM example and figure is given by following and shows in Fig.2.5 right,

$$\begin{aligned} \mathit{Offset} = & [0 \quad 1; 0 \quad 2; 0 \quad 3; 0 \quad 4; ... \\ & -1 \quad 1; -2 \quad 2; -3 \quad 3; -4 \quad 4; ... \\ & -1 \quad 0; -2 \quad 0; -3 \quad 0; -4 \quad 0; ... \\ & -1 \ -1; -2 \ -2; -3 \ -3; -4 \ -4] \end{aligned}$$

2.3 Feature Selection

Feature selection aims to select relevant and informative features for classification. It is applied to improve classification performance, to reduce computational complexity, and to interpret data.

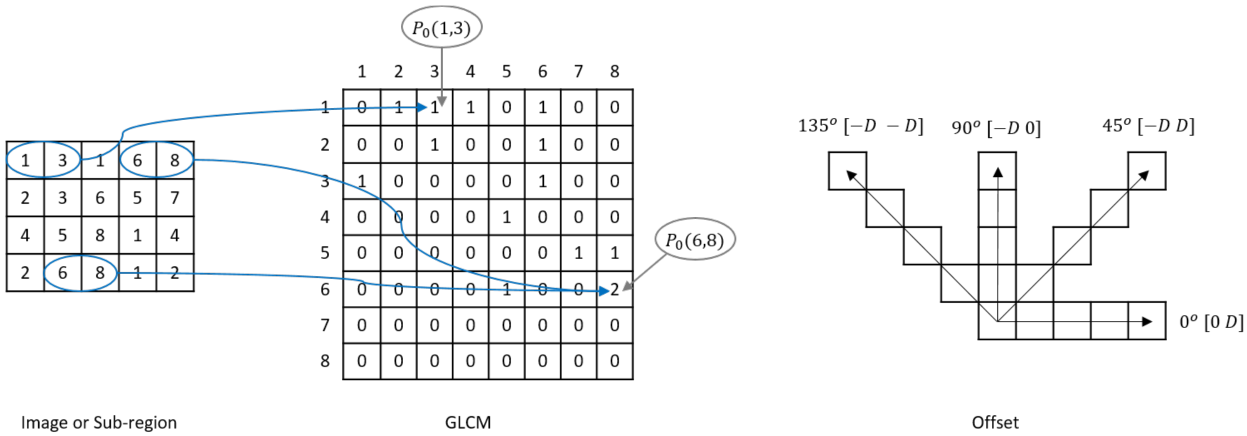

Figure 2.5: Process used to create the GLCM (left) and Offset of GLCM (right)

2.3.1 Correlation-based Feature Selection (CFS)

This method selects features that are highly correlated with the pattern classes, but have low correlation with the remaining features. The subset evaluation function is given by:

$$Merit_S = \frac{\bar{k}_{r_{cf}}}{\sqrt{k + k(k-1)\bar{r_{ff}}}} \tag{2.21}$$

where $Merit_S$ is the merit of the selected feature set S, $\bar{k}_{r_{cf}}$ is the mean correlation between the features and class with $f \in S$, and $\bar{r_{ff}}$ is the mean pairwise feature correlation. The numerator expresses predictive capacity, while the denominator expresses feature redundancy.

Best First Search (BF)

Searches the space of feature subsets by greedy hillclimbing that may include backtracking. Best first may search forward, or backward, or, consider all possible single feature additions and deletions at a given point using a bi-directional strategy.

Genetic Algorithm-based Search (GA)

Genetic search works by having a population of variables representing feature sets and performs the operations of reproduction, cross-over and mutation in each generation to get the offspring that optimizes a feature set-related objective function.

2.3.2 Information Gain (IG)

This function measures the information gain with respect to the class:

$$\text{InfoGain(Class,Attribute)} = H(\text{Class}) - H(\text{Class}|\text{Attribute}) \tag{2.22}$$

where H is the entropy of each class given by $H(\text{Class}) = -p_{\text{Class}} \log p_{\text{Class}}$ We select the attributes by individual ranking evaluation.

2.3.3 Ranker

Using Ranker as a search means that we will rank the features based on the features' individual evaluations. A threshold can be set in Ranker, and features that are smaller than this threshold will be removed from the feature set. Ranker used with attribute evaluators, such as Information Gain (IG), feature selection and entropy, etc.

2.4 Classifiers-Discriminant Functions

2.4.1 Naïve Bayes (NB)

$$p(\omega_j|x) = \frac{p(x|\omega_j)P(\omega_j)}{p(x)}$$

Bayes formula can be expressed informally in English by saying that

$$posterior = \frac{likelihood \times prior}{evidence}$$

This model assumes conditional statistical independence $p(\mathbf{x}|\omega_j) = \prod_{k=1}^{D} p(x_k|\omega_j)$ where $\mathbf{x} = (x_1, x_2, \ldots, x_D)^T$ and D is the dimensionality of the feature space. The posterior probability is based on Bayes' formula. The MAP decision rule is typically used for classification.

Suppose we have two categories ω_1 and ω_2 with discriminant functions $g_1(x), g_2(x)$, where

$$g_i(x) = -\frac{1}{2}(x-\mu_i)^T\Sigma_i^{-1}(x-\mu_i) + \frac{D}{2}\ln 2\pi - \frac{1}{2}\ln|\Sigma_i| + \ln P(\omega_i)$$

Then we can define a single discriminant function by

$$g(x) = g_1(x) - g_2(x)$$

The decision rule is :

$$\begin{cases} \omega_1, & \text{if } g(x) > 0 \\ \omega_2, & \text{if } g(x) < 0 \end{cases}$$

2.4.2 Multilayer Perceptron (MLP)

A multilayer perceptron (MLP) is a feedforward artificial neural network system that maps input patterns onto class labels. An MLP has multiple layers of nodes that are fully connected to the next layer. Each node is a neuron with a nonlinear activation function. MLP utilizes backpropagation for supervised learning [78, 80]. Because MLP has multiple layers of logistic regression models, it can distinguish data that are not linearly separable. In learning by backpropagation -that can be considered as an extension of the LMS algorithm-we adjust the connection weights, according to the amount of error in the output compared to the expected result.

2.4.3 Bayes Network (BN)

A Bayes network, is a probabilistic graphical model that uses a directed acyclic graph to represent a set of random variables and their conditional dependencies. In a Bayesian network the joint probability density function can be written as the product of univariate conditional density functions dependent on their parent variables:

$$p(x) = \Pi_{v \in V} p\left(x_v | x_{pa(v)}\right) \tag{2.23}$$

where $pa(v)$ denotes the parents of v. In the graph, the parents are vertices directly connected to v by a single edge.

2.4.4 Bagging

For a training set S with size k, Bagging generates j training subsets denoted as S_i with size $k' < k$, by sampling from S uniformly and with replacement. We denote the original set as A. In the training stage, we first have $D = \emptyset$ and j is the number of classifiers to train. Then for $p = 1, 2, ..., j$, we take a bootstrap sample S_p from A to train classifier D_p. Then we add the classifier D_p to the current ensemble, $D = D \cup D_p$. We obtain the class label prediction for the input x by majority voting on the individual classifier decisions produced by $D_1, ..., D_j$ [26].

Algorithm 1 Bagging

Input: Training set S of size k
Generate j training subsets S_i ($n' < n$) with replacement.
Training stage
    Initialize Original set as A, $D = \emptyset$.
    Build a classifier D_p, using a bootstrap sample S_p from A as the training set,
    $D = D \cup D_p$, where $p = 1, 2, ..., j$
Classification stage
    Input x
    Perform classification decisions from $D_1, ..., D_m$ on x
Output: Voting decision for x.

2.4.5 Random Forests (RF)

Random forests are an ensemble learning method that constructs multiple decision trees from subsets of the training set and uses random feature selection for node splitting. RF decide the class after applying voting to the predicted classes by the individual trees for classification, or by calculating the mean prediction for regression. Random forests address the overfitting tendency of the decision trees and have shown robustness with respect to noise [62].

2.5 Patch-based Classification

2.5.1 Bag of Keypoints

In our experiments we have also evaluated the classification performance of the Bag of Keypoints (BoK) [20, 102] that is another patch-based technique comparable to our method. Bag of Features methods have been applied to image recognition and classification and have produced very good results. The Bag of Keypoints technique originates from the Bag of Features. These methods apply feature detection, extraction and clustering for finding the most representative features in the training database. In the next step they build a vocabulary that consists of the frequency of occurrence of these features. In the testing stage, features are extracted from the unlabeled image and encoded using the vocabulary that was built during training. Then a learning method is applied to classify the test pattern into one of the classes.

In this work we employed the support vector machine (SVM) classifier for learning a discriminant function from the encoded features and classifying unlabeled samples. In SVM we evaluated the use of linear or radial basis function kernels. We utilized radial basis function kernels for our experiments to address possible non-linearity of the decision boundary. The main parameters that we tuned were the fraction of features to keep for building the vocabulary, the vocabulary size, the penalty coefficient for misclassification of training samples in SVM, and the kernel scale.

Chapter III: SPARSITY-BASED TECHNIQUES

The concept of sparsity has been used in many methods of mathematics, computer science and engineering and plays an important role in machine learning and pattern recognition. In this chapter, we introduce the standard sparse technique, the details of this method, and other related sparse techniques.

3.1 Overview of Sparse Modeling Methods

Sparse techniques are based on a matrix in which the majority or a significant number of its entries are zeros. This means that there is a redundancy in representation and only a fraction of the features may be needed for approximating a pattern. These features are expected to be more relevant to training and classification than the features with zero or very small coefficients. The sparsity property may be used for finding compact representations that simplify the pattern recognition problem.

Sparse techniques have been applied in many fields in the past years. Especially in high dimensional problems, low dimensional structures may be extracted to represent relevant information. For example, in face recognition, only few features are sufficient for representation because the sparse model includes only few nonzero entries [97].

Tissue classification is typically achieved by supervised machine learning approaches. Among numerous techniques that proposed generative or discriminative models, use of kernels, and linear or nonlinear approaches, sparse classification techniques have shown promise and applicability for characterizing visual patterns in region of interest (ROI)-based analyses. Sparse representation techniques have been applied to extensive fields including coding, feature extraction and classification, superresolution [98], and regularization of inverse problems [30]. Exploration of signal's sparsity may provide insight into the important patterns of prototyping of objects category. The sparse representation is more concise for compression and naturally discriminative for classification [97]. Sparse representation techniques calculate a sparse linear combination of atoms for describing a vector sample using an overcomplete dictionary of prototypes. If the representations of these linear combinations are sufficiently sparse, then they can be used for object recognition and classification of imaging patterns.

Sparse representation methods have been applied to a wide range of fields including coding, feature extraction and classification, superresolution, and regularization of inverse problems [97, 104, 103, 75]. In addition, sparse representation may provide insight into significant patterns that form object category prototypes. Sparse representation techniques describe a vector sample by sparse linear combinations of atoms from an overcomplete dictionary of prototypes. If these representations are sparse enough, then the representations reveal characteristic imaging patterns of disease and can be used for object recognition and classification. The authors in [97] proposed the sparse representation classification (SRC) method to recognize 2,414 frontal-face images of 38 individuals of Yale B Database and over 4,000 frontal images for 126 individuals of AR Database, the recognition rates are above 90% for both database. In the cases of recognition under random corruption and under varying level of contiguous occlusion, the recognition rates increased further. A regression and spectral graph analysis based method has been used for sparse representation, and compared with other methods, such as principal component analysis (PCA), SparsePCA, and linear discriminant analysis (LDA) in [14]. The proposed method was evaluated on CMU, PIE, and Yale-B datasets. Other notable applications of sparse coding methods were published in [103, 75] reporting high levels of classification accuracy.

The sparsity preserving projections (SPP) technique was proposed in [75]. It solves a modified sparse representation problem to create a sparse reconstructive weight matrix. Then a low dimensional feature space is calculated as a minimizer of an objective function that includes the weight matrix.The advantage of this method is the invariant to rescaling, rotation and translation of the data. It also produces natural discriminant representations for supervised and unsupervised problems. The weight vector $\mathbf{s}_i = [s_{i1}, ..., s_{i,i-1}, 0, s_{i,i+1}, ..., s_{in}]$ is constructed as follows,

$$\min_{\mathbf{s}_i} \parallel \mathbf{s}_i \parallel_1 \tag{3.1}$$

$$\text{s.t. } \mathbf{x}_i = \mathbf{X}\mathbf{s}_i \tag{3.2}$$

$$1 = \mathbf{1}^T \mathbf{s}_i \tag{3.3}$$

where $\mathbf{x}$ is training sample, $\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_n]$,$\mathbf{1}$ is a vector with all ones. Then the sparse reconstructive weight matrix can be expressed as

$$\mathbf{S} = [\widehat{\mathbf{s}}_1, \widehat{\mathbf{s}}_2, ..., \widehat{\mathbf{s}}_n]^T \tag{3.4}$$

The SPP method was applied to face recognition on the Yale, AR and extended Yale B datasets. It was compared with PCA, local preserving projection (LPP) and neighhborhood preserving embedding (NPE). SPP yielded the highest accuracy for these four data sets among the compared methods [75].

Dictionary learning techniques have also emerged as solutions for sparse representation in the recent years. The utilization of K-SVD, where SVD denotes singular-value decomposition, for dictionary learning has been studied to produce a dictionary aiming for more accurate representation [3]. In [54], the K-SVD technique has been used for color image restoration to handle nonhomogeneous noise and information missing problems. The authors in [101] observed that K-means may yield as good precision rate as K-SVD when we use the same number of atoms. The SRC method with dictionary learning was applied to classification of pulmonary patterns of diffuse lung disease in [104]. 1161 volumes of interest were used for classification and yielded very high accuracy. Additional algorithms such as matching pursuit (MP), orthogonal matching pursuit (OMP), and basis pursuit (BP) have been proposed for codebook design [3].

3.2 Sparse Representation and Classification

Sparse Representation techniques construct a dictionary from labeled training samples to calculate a linear representation of a test sample. This representation can be used to make a decision for the class of the test sample. Assuming that a dataset has k distinct classes, s samples, and for ith class there are s_i samples, so that $s = \sum_i s_i$, we define a dictionary matrix M from the training set as

$$M = [v_{1,1}, v_{1,2}, ..., v_{k,s_k}]. \tag{3.5}$$

where $M \in \mathbb{R}^{l \times s}$, and $v_{i,h}$ is a column vector for the hth sample from ith class. In image classification applications, a $p \times q$ grayscale image forms a vector $v \in \mathbb{R}^l$, $l = p \times q$ using lexicographical ordering.

A new test sample $y \in \mathbb{R}^l$, can be represented by a linear combination of samples $y = \sum_{i=1}^{k} \beta_{i,1} v_{i,1} + \beta_{i,2} v_{i,2} + \cdots + \beta_{i,s_i} v_{i,s_i}$, where $\beta_{i,h} \in \mathbb{R}$ are scalar coefficients. Hence, the test sample y can be rewritten as:

$$y = Mx_0 \in \mathbb{R}^l. \tag{3.6}$$

where x_0 is a sparse solution. If there are sufficient training samples, the components of x_0 are equal to zero except for the components corresponding to the ith class. Then $x_0 = [0, 0, ..., \beta_{i,1}, \beta_{i,2}, ..., \beta_{i,s_i}, 0, 0, ..., 0]^T \in \mathbb{R}^s$.

In [25], it was proved that whenever $y = Mx$ for some x, if there are less than $l/2$ nonzero entries in x, x is the unique sparse solution: $\widehat{x}_0 = x$. Finding an accurate sparse representation of an underdetermined system of linear equations is an NP-hard problem [22, 6], therefore only approximate solutions can be found. The authors in [15, 16, 24] supported that if the solution x_0 is sparse enough, it is equal to the solution $\widehat{x}_1$ of the l^1-minimization problem:

$$(l^1): \quad \widehat{x}_1 = \arg\min ||x||_1 \quad s.t. \quad Mx = y. \tag{3.7}$$

In sparse representation classification we define a characteristic function $\delta_i : \mathbb{R}^s \to \mathbb{R}^s$ that has nonzero entries, only if x is associated with class i. Then the function $\hat{y}_i = M\delta_i(\hat{x}_1)$, represents the given sample y using components from class i only. To classify y and determine the class label $\widehat{\omega_i}$, we minimize the residual between y and $\hat{y}_i$ [97]:

$$\widehat{\omega_i} = \arg\min_i r_i(y) \doteq ||y - M\delta_i(\hat{x}_1)||_2. \tag{3.8}$$

This technique also adopts the sparsity concentration index (SCI) to measure the efficiency of class-conditional representation of a sample. The SCI of a coefficient vector $x \in \mathbb{R}^s$ is $SCI(x) = \frac{k \times \max_i ||\delta_i(x)||_1/||x||_1 - 1}{k-1} \in [0,1]$ as defined in [97]. For a solution $\widehat{x}$, if $SCI(\widehat{x})$ is 1, y is only represented by images from a single class, and if $SCI(\widehat{x}) = 0$, the components of β are spread evenly over all classes.

Algorithm 2 Sparse Representation-based Classification (SRC)

1: **Input**: A training samples matrix for k classes
$M = [v_{1,1}, v_{1,2}, ..., v_{k,s_k}] \in \mathbb{R}^{l \times s}$,
A test sample $y \in \mathbb{R}^l$.
2: Solve the l^1-minimization problem:

$$(l^1): \quad \widehat{x}_1 = \arg\min ||x||_1 \quad s.t. \quad Mx = y. \tag{3.9}$$

3: Compute the residuals

$$\min_i r_i(y) \doteq ||y - M\delta_i(\hat{x}_1)||_2. \tag{3.10}$$

for $i = 1, ..., k$.
4: **Output:** Identify $\widehat{\omega_i} = \arg\min_i r_i(y)$.

3.3 Algorithms for solving sparse representation problem

In Sec. 3.1 we mentioned that finding the accurate solution of sparse representation is an NP hard problem, and in Sec. 3.2 we have used one of the common methods to solve the solution in Eq. (3.7). We describe two common methods for the NP hard problem. One is the matching pursuit (MP) method. In [71] the authors proposed a algorithm as orthogonal matching pursuit (OMP). OMP modified MP to achieve full backward orthogonality of residuals (error) at each step, resulting in improved convergence. Another optimization method for this problem is basis pursuit (BP) [2].

3.3.1 Matching Pursuit

Matching pursuit was originally proposed for time-frequency analysis, and now it is employed as a sparse approximation algorithm as well. It attempts to find the best matching solution of a given signal f from Hilbert space H, through the sum of multiple atoms g_{γ_n} that are the components of f on an over-complete dictionary D with their corresponding weight [56],

$$f(t) \approx \widehat{f}_N(t) := \sum_{n=1}^{N} a_n g_{\gamma_n}(t) \tag{3.11}$$

where a_n is the scalar weighting factor.

MP selects atoms one at a time to minimize the approximation error. This is done by finding the atom with the largest inner product of the signal, subtracting the approximation from the signal using only that atom, repeat this step until it finds the satisfying residual,

$$R_{N+1} = f - \widehat{f}_N \tag{3.12}$$

Then if R_{N+1} can converge quickly, only few atoms are needed. This process is solving the problem

$$\min_x \|f - Dx\|_2^2 \quad \text{s.t.} \quad \|x\| \leq N \tag{3.13}$$

and Eq. (3.13) is same as Eq. (3.8). One of the applications in [92] uses the OMP algorithm to solve sparse approximation problem on a redundant dictionary.

3.4 Linear Programming

A linear programming (LP) problem is a constrained optimization problem that seeks the minimizer x of a linear objective function $C^T x = c_1 x_1 + c_2 x_2 + \cdots + c_n x_n$ subject to linear constraints [77],

$$\min_x C^T x \quad \text{subject to} \quad \begin{cases} A \cdot x \leq b \\ Aeq \cdot x = beq \\ lb \leq x \leq ub \end{cases} \tag{3.14}$$

where b and beq are inequality and equality vectors respectively, A is inequality matrix, and Aeq is equality matrix. Here lb denotes the lower bounds vector, and ub denotes the upper bounds vector.

The SRC method uses the $Aeq \cdot x = beq$ to find a linear representation and the function to be minimized is the l^1 norm. The approximated solution is $\hat{x}_1$ for SRC method. In SRC the components of the solution vector x are assigned to their respective object classes. We use the Interior-Point Solver to find the solution of the LP problem.

The KKT Conditions

The Karush-Kuhn-Tucker (KKT) conditions use generalized Lagrangian multipliers to determine if the point x is an optimal solution in a feasible region. Let $f : \mathbb{R}^n \to \mathbb{R}$ be a objective function, smooth constraint functions $\mu_w(x) \geq 0, w = 1, ..., m$, and a collection of Lagrange multipliers $\lambda \geq 0$. Then our optimization problem is equivalent to minimization of $\mathcal{L}(x, \lambda) = f(x) + \sum_{w=1}^{m} \lambda_w \mu_w(x)$ [77].

Interior-Point Linear Programming Algorithm

The interior point method traverses the interior of the feasible region on a path towards the boundary to reach an optimum solution. We seek to minimize the barrier function $F(x, \mu) = C^T x - \mu \sum_{w=1}^{n} \ln x_w$ subject to $Aeq \cdot x = beq$, instead of (3.14), as the solutions produced by the projective algorithm and by use of barrier methods were shown to be equivalent [36]. The Lagrangian is defined by $\mathcal{L}(x, \lambda) = C^T x - \mu \sum_{w=1}^{n} \ln x_w - \lambda^T (Aeq \cdot x - beq)$. To detect the optimal solution, a search direction $d_F = x + \frac{1}{\mu} X^2 (Aeq^T \lambda^* - C)$ can be defined [77], such that $x_{\gamma+1} = x_\gamma + \alpha d$ satisfies $C^T x_{\gamma+1} < C^T x_\gamma$, where $X = diag(x_1, x_2, \cdots, x_n)$ and α is a parameter. If $\mu \to 0$, then the optimal solution to the barrier function will be the optimal solution to the original LP problem [77]. Then we can compute the direction d_F and solve $\min_\alpha F(x, \mu)$.

3.4.1 Basis Pursuit

The basis pursuit (BP) is another method for decomposition of an overcomplete system. BP is a mathematical optimization problem, which decomposes a signal into an optimal superposition of dictionary elements and the optimal mean has the smallest l_1 norm coefficient over all the compositions [18].

Basis pursuit solves the following problem

$$\arg\min_x 1/2\|Ax - y\|_2^2 + \lambda\|x\|_1 \tag{3.15}$$

The relation of BP with fields of ill-posed problem and total variation denoising are interesting. BP leads to a large-scale optimization problem in highly overcomplete dictionary [18].

3.5 Second order cone programming

The second order cone (SOCP) programming problems are convex optimization problems. The SOCP can be used to implement linear programming (LP), convex quadratic programs (QPs) and convex quadratically constrained quadratic programs (QCQPs) [5]. The standard form of SOCP is defined as following:

$$\min \quad \mathbf{u_1}^\top \mathbf{x_1} + \cdots + \mathbf{u_n}^\top \mathbf{x_n} \tag{3.16}$$

$$s.t. \quad A_1\mathbf{x_1} + \cdots + A_n\mathbf{x_n} = \mathbf{b} \tag{3.17}$$

$$\mathbf{x_i} \succeq \mathbf{0} \text{ for } i = 1, 2, ..., n \tag{3.18}$$

The SOCP problems can implement LP problems, where the standard form of LP is

$$\min \quad \sum_{i=1}^{k} c_i x_i \tag{3.19}$$

$$s.t. \quad \sum_{i=1}^{k} x_i \mathbf{a_i} = \mathbf{b} \tag{3.20}$$

$$x_i \geq 0 \text{ for } i = 1, 2, ..., k \tag{3.21}$$

The standard form of LP may be described as a special cased of SOCP standard form. QPs and QCQPs can be implemented by SOCP by substituting some variables or vectors.
SOCP models are widely applied in the fields of engineering, such as filter design and antenna array design [52], robust optimization control, finance [5, 52].

We utilize a method that solves the following problem,

$$\min_x f(x) \text{ s.t. } \begin{cases} c(x) \leq 0 \\ ceq(x) = 0 \\ A \cdot x \leq b \\ Aeq \cdot x = beq \\ lb \leq x \leq ub \end{cases} \tag{3.22}$$

where $f(x)$ is a object function, lb and ub are lower bound and upper bound respectively, A is a matrix and b is a vector for inequality, Aeq is a matrix and beq is a vector for equality, and $c(x)$ and $ceq(x)$ are constraint functions that return vectors. Especially, $f(x), c(x)$ and $ceq(x)$ can be nonlinear functions.

To solve SOCP problems by using the formulation of Eq (3.22), we may utilize interior-point optimization, SQP or active-set optimization algorithms. The implementation parameters for the Eq (3.22) will be introduced in the following sections.

3.5.1 Interior-point Optimization

The interior-point optimization algorithm searches through all the interior of the feasible region to obtain the optimal solution and can be described as,

$$\min_x f(x) \text{ s.t. } g(x) \leq 0 \text{ and } h(x) = 0. \tag{3.23}$$

The corresponding barrier function of (3.23) is

$$B(x, c) = f(x) - \mu \sum_i^m \ln (c_i) \tag{3.24}$$

where $g(x) + c = 0$, $i = 1, ..., m$. We define $c_i > 0$, the logarithmic term of Eq (3.24) is bounded. μ is a small positive scalar, when $\mu \to 0$, the right side term of Eq (3.24) approximates to zero, hence $B(x, c)$ is the value of $f(x)$.

The reason we use Eq (3.24) instead of the standard form Eq(3.23) is because Eq (3.24) is an equation constrained problem that is easier to solve than the standard form Eq (3.23) inequality constrained problem.

The gradient of the barrier function Eq (3.24) is

$$gradB = gradf - \mu \sum_{i=1}^{m} \frac{1}{c_i(x)} \nabla c_i(x) \tag{3.25}$$

where *gradf* is the gradient of f and ∇c_i is the gradient of c_i.

Let $\lambda \in \mathbb{R}^m$ denote the Lagrange multiplier vector associated with constrain function g, such that

$$\forall_{i=1}^{m} \quad g_i(x)\lambda_i = \mu \tag{3.26}$$

Eq (3.26) is similarly as the condition of "complementary slackness" in KKT conditions, and the condition (3.26) is called "perturbed complementarity" sometimes.

We denote the Hessian matrix of the Lagrangian of barrier function by [12, 13, 94],

$$H = \nabla^2 f(x) + \sum_i \lambda_i \nabla^2 g_i(x) + \sum_j \lambda_j \nabla^2 h_j(x) \tag{3.27}$$

Then the Newton step is

$$\begin{pmatrix} H & 0 & J_h^T & J_g^T \\ 0 & S\Lambda & 0 & -S \\ J_h & 0 & I & 0 \\ J_g & -S & 0 & I \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta s \\ -\Delta y \\ -\Delta \lambda \end{pmatrix} = - \begin{pmatrix} \Delta f - J_h^T y - J_g^T \lambda \\ S\lambda - \mu e \\ h \\ g + s \end{pmatrix} \tag{3.28}$$

where J_g and J_h are the Jacobians of the constraint functions g and h, respectively. S and Λ are diagonal matrices of s and λ. y is the Lagrange multiplier vector associated with h and e is the vector of ones that is the same size as g [12, 13, 94].

3.5.2 Active Set Algorithm

The active set algorithm is an iterative algorithm. In the original active set algorithm, all the iterative points are feasible solutions of the problem. The algorithm starts from a initial feasible solution and follows the rules of the iteration, until it reaches the maximum number of steps corresponding to the optimal solution of the problem.

As we know, an equality constraint is much easier to handle than an inequality constraint, and that explains why the concept of active sets is proposed. The active set is a set of subscripts of all the constraint conditions that holds equality constraint and contains all the equality constraints and a subset of inequality constraint. The optimal active set can help to solve the problem fast, since we only need to rewrite the inequality constraint as equality constraint, and not include the other constraint conditions, and then we use Lagrange multiplier to solve the problem. Next, we will introduce how to obtain the optimal active set.

We define a working set that is a subset chosen from all the constraint conditions that includes all the equality constraints and some of the inequality constraints. Assuming this working set is the optimal active set, we solve for the corresponding optimization subproblems of the working set. Then we use the reformulated inequality constraint and equality constraint, not including the other constraints, and then use Lagrange multiplier to solve the problem.

The initial feasible solution x_0 is the starting point, and it can be obtained by the same method or by linear programming. We assume there is a initial feasible solution x_0, and the active set is the working set W_0.

After $k, k = 0, 1, 2, ...$ iterations, we denote the iteration point by x_k, and the working set is W_k. If x_k satisfies the KKT conditions, then it is the optimal solution. Otherwise, we continue to the next iteration.

3.5.3 Sequential Quadratic Programming (SQP)

The SQP algorithm is similar to active-set algorithm. It also attempts to compute the Lagrange multipliers directly and it solves a quadratic programming (QP) problem at each major iteration. At each major iteration, it uses a quasi-Newton updating method to approximate the Hessian matrix of the Lagrangian function, and the solution of QP subproblem is used to determine a direction of line search procedure [31]. To apply the SQP algorithm, the object function and constraints need to be twice continuously differentiable.

The differences of SQP and active-set algorithm are 1) each iterative step of SQP is in the region that is constrained by bounds implying strict feasibility with respect to bounds; 2) in the iterative process, the SQP algorithm may attempt to fail, and then it can take a smaller step; 3) to solve QP subproblems, the SQP uses a different set of linear algebra routines; 4) if the constraints the QP subproblems are not satisfied, the SQP can combine the objective and constraint function into a merit function or the SQP attempts to use second order approximation to obtain feasibility to the constraints [87, 89, 13].

Chapter IV: INTEGRATIVE ENSEMBLE SPARSE ANALYSIS TECHNIQUES

In this chapter we propose a method for finding sparse solutions by reducing the dimensionality of the feature vectors and correcting the bias of estimation using ensembles of Bayesian decision learners. We introduce a classification method that calculates sparse representations of block structures for given ROIs and builds an ensemble model of sparse learners to make a decision on lesion category. We hypothesize that the combination of relative sparsity scores of multiple disjoint sparse representations computed from multiple dictionaries will yield a more robust decision function than the decision function derived from a single dictionary used in conventional sparse representation classification techniques. We also propose a block-based log likelihood (BBLL) decision system and a minimum Bayes error-based approach for determining the decision threshold that will address classification bias. The optimized parameters may be used to define probability decision scores (PDS) in order to determine confidence intervals for prediction. This approach is advantageous in constructing overdetermined linear systems and addressing numerical optimization problems, such as convergence to infeasible solutions. The development of a classifier ensemble learning approach and the introduction of two Bayesian decision functions aim to improve classification accuracy.

4.1 Block Decomposition and Ensemble Classification

Conventional sparse representation techniques may not find a good approximation of the solution vector, if the pattern dimensionality is high and the number of training samples is small. This is a typical case for medical image classification applications that may include lesions of variable types and limitations in the availability of training samples.

The images that we use for lesion characterization are subject to intra-class variability, that cause the samples to depart from the true class prototype. Furthermore, the high

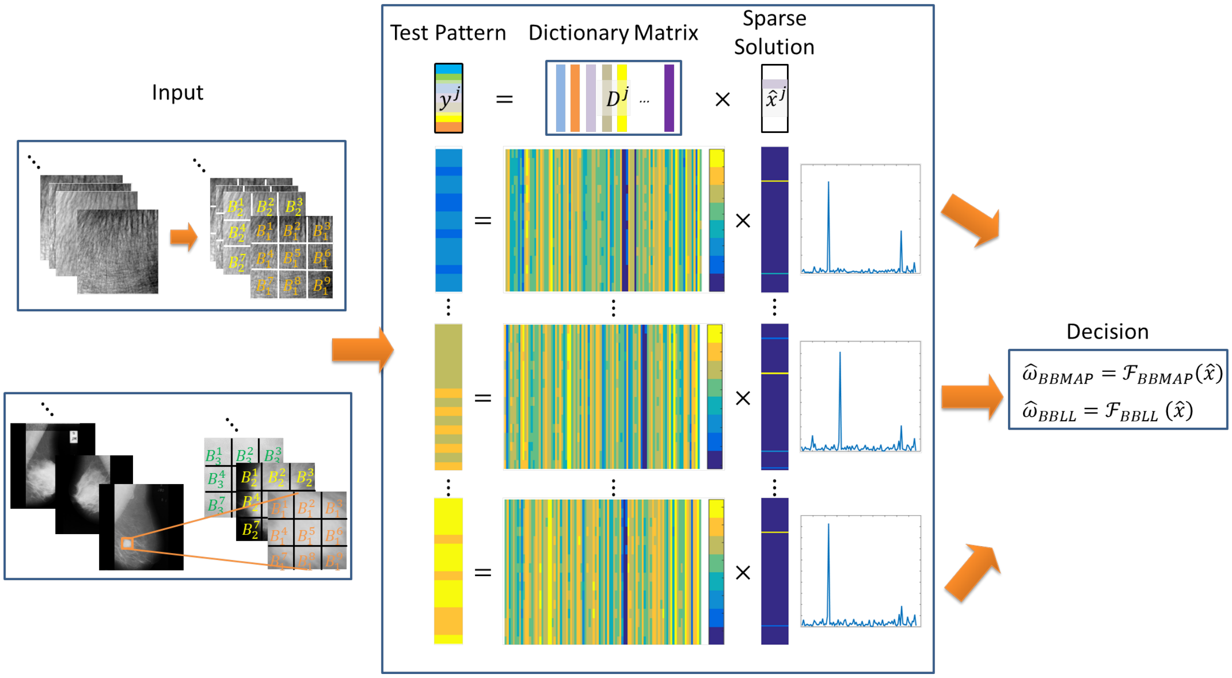

Figure 4.1: Main stages of our integrative sparse modeling system: block-based analysis, sparse solutions, and decision functions.

dimensionality of the feature space complicates the optimization procedure and may lead to infeasible solutions.

We propose to build an ensemble of sparse representation classifiers based on block decomposition of the input ROI to address these shortcomings. Fig. 4.2 summarizes the main stages of our method that may be divided in block-based learning and Bayesian model averaging to form decision functions.

4.1.1 Block Decomposition

We first divide each training ROI into non overlapping blocks of size $m \times n$. Thus, each ROI image is expressed as $I = [B^1, B^2, ..., B^{NB}]$, where NB is the number of blocks in an image. The dictionary D^j, where $j = 1, 2, ..., NB$ corresponds to the block B^j at the same index within the image ROI. The dictionary D^j for all the s images can be represented as follows:

$$D^j = [bv_{1,1}^j, bv_{1,2}^j, \ldots, bv_{k,s_k}^j], \tag{4.1}$$

where $bv_{i,h}^j$ is the column vector denoting the hth sample, ith class, jth block B^j.

4.1.2 Ensemble Classification

We propose to classify each test sample by constructing ensembles of classifiers that solve a set of sparse coding and classification problems, or hypotheses corresponding to the block components. Given a test sample y^j in jth block, we find the solution x^j of the regularized noisy l^1-minimization problem:

$$\widehat{x}^j = \arg\min ||x^j||_1 \text{ subject to } \|D^j x - y^j\|_2 \leq \epsilon \tag{4.2}$$

where $j = 1, 2, ..., NB$. The test sample y^j will be assigned to the class ω_i^j, which has minimum approximation error calculated by (4.3).

$$\omega_i = \arg\min_i r_i(\widehat{x}) \doteq \arg\min_i \|y - \widehat{y}_i\|_2. \tag{4.3}$$

We propose ensemble learning techniques in a Bayesian probabilistic setting as weighted sums of classifier predictions. We propose a function that applies majority voting to individual hypotheses (BBMAP) and an ensemble of log likelihood scores computed from relative sparsity scores (BBLL).

Maximum a Posterior decision function (BBMAP)

The class label for each test sample is determined by voting over the ensemble of NB block-based classifiers. The predicted class label $\widehat{\omega}$ is given by

$$\widehat{\omega}_{BBMAP} = \mathcal{F}_{BBMAP}(\widehat{x}) \doteq \arg\max_i pr(\omega_i|\widehat{x}), \tag{4.4}$$

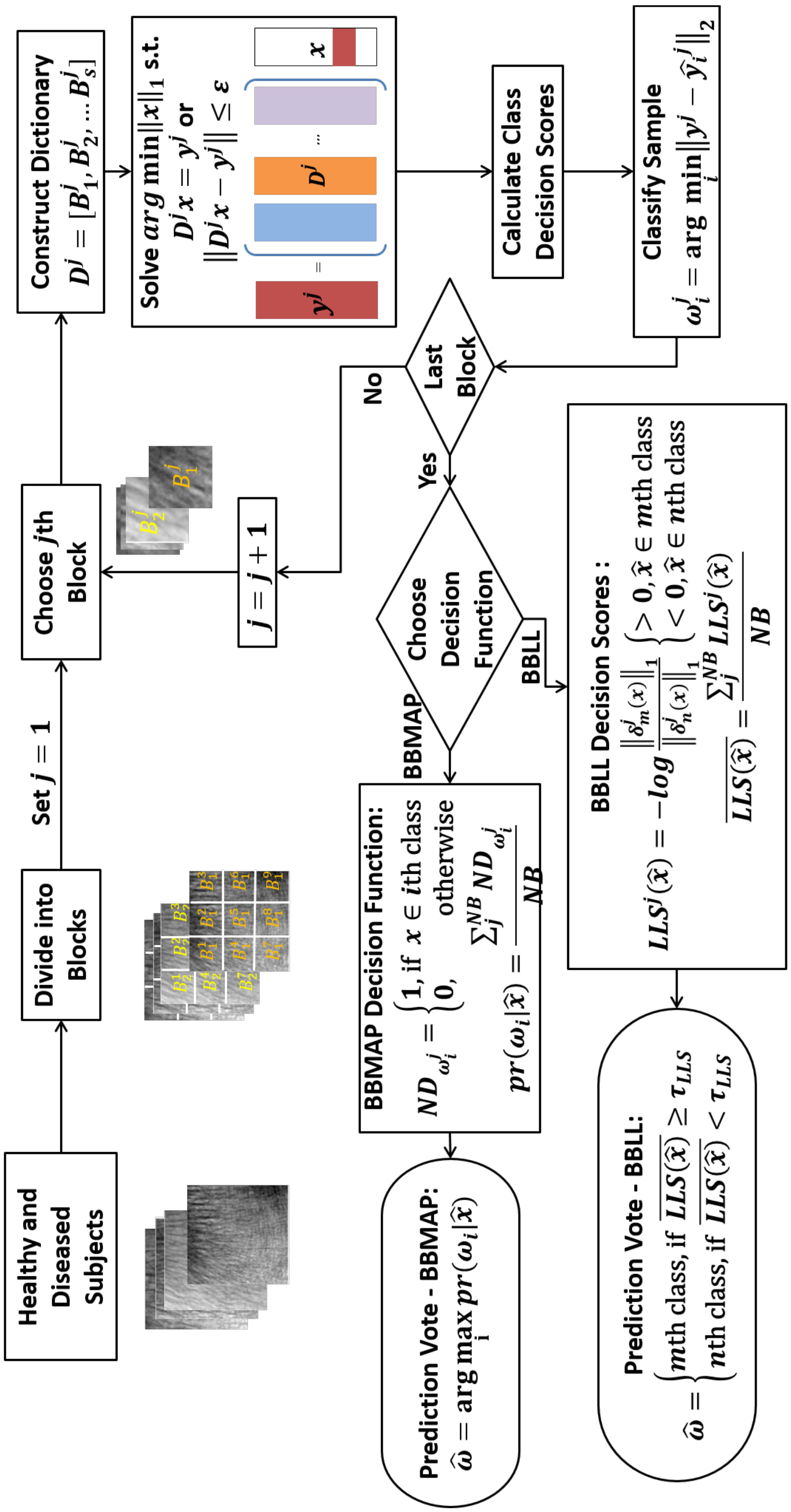

Figure 4.2: Main stages of the proposed system: block decomposition, construction of ensemble of sparse learners, and classification by probabilistic model averaging.

where $\widehat{x}$ is the composite extracted feature from the test sample given by the solution of (4.2). The probability for classifying $\widehat{x}$ into class ω_i is

$$pr(\omega_i|\widehat{x}) = \sum_{j}^{NB} ND_{\omega_i^j}/NB \tag{4.5}$$

$$ND_{\omega_i^j} = \begin{cases} 1, & \text{if } \widehat{x}^j \in i\text{th class} \\ 0, & \text{otherwise} \end{cases}, \tag{4.6}$$

where $ND_{\omega_i^j}$ is an indicator function whose values are determined by the individual classifier decisions.

Log likelihood approximation residual-based decision function (BBLL-R)

We define a likelihood score based on the residuals r_m, r_n calculated in the sparse representation stage of each classifier

$$LLS^j(\widehat{x}) = -\log \frac{r_m^j(\widehat{x})}{r_n^j(\widehat{x})} \begin{cases} > 0, & \widehat{x} \in m\text{th class} \\ < 0, & \widehat{x} \in n\text{th class} \end{cases}. \tag{4.7}$$

, where $r_\omega^j(y)$ is the approximation residual for class ω and j is the block index:

$$r_\omega^j(y) = ||y - M\delta_\omega(\hat{x}_1)||_2 \text{ for } j = 1, ..., k. \tag{4.8}$$

We calculate the expectation of $\overline{LLS(\widehat{x})}$ over all classifiers that is determined by individual classification scores derived from (4.7):

$$\begin{aligned}\overline{LLS(\widehat{x})} &= \sum_{j}^{NB} LLS^j(\widehat{x})/NB \\ &= -\frac{1}{NB}\left[\sum_{j}^{NB} \log r_m^j(\widehat{x}) - \sum_{j}^{NB} \log r_n^j(\widehat{x})\right],\end{aligned} \tag{4.9}$$

Log likelihood sparsity-based decision function (BBLL-S)

We define a likelihood score based on the relative sparsity scores $\|\delta_m(\widehat{x}^j)\|_1$, $\|\delta_n(\widehat{x}^j)\|_1$ calculated in the sparse representation stage of each classifier

$$LLS^j(\widehat{x}) = -\log\frac{\|\delta_m(\widehat{x}^j)\|_1}{\|\delta_n(\widehat{x}^j)\|_1} \begin{cases} > 0, & \widehat{x}^j \in m\text{th class} \\ < 0, & \widehat{x}^j \in n\text{th class}\end{cases}. \tag{4.10}$$

We calculate the expectation of $LLS(\widehat{x})$ over all classifiers that is determined by individual classification scores derived from (4.10):

$$\begin{aligned}\overline{LLS(\widehat{x})} &= \sum_{j}^{NB} LLS^j(\widehat{x}^j)/NB \\ &= -\frac{1}{NB}\left[\sum_{j}^{NB} \log\|\delta_m(\widehat{x}^j)\|_1 - \sum_{j}^{NB} \log\|\delta_n(\widehat{x}^j)\|_1\right],\end{aligned} \tag{4.11}$$

The introduction of the log-likelihood score accommodates the definition of a decision function for the state $\widehat{\omega}$. To determine the class we apply a decision threshold τ_{LLS} to $\overline{LLS(\widehat{x})}$.

$$\widehat{\omega}_{BBLL} = \mathcal{F}_{BBLL}(\widehat{x}) \doteq \begin{cases} m\text{th class}, & \text{if } \overline{LLS(\widehat{x})} \geq \tau_{LLS} \\ n\text{th class}, & \text{otherwise}\end{cases}. \tag{4.12}$$

This threshold is expected to be equal to 0, if there is no estimation bias, but may be experimentally determined as the minimizer of a Bayes-type risk function. Hence the optimal τ^*_{LLS} value can be determined by sampling the domain of τ_{LLS} and calculating true positive and true negative rates. Next, the optimal value is determined by the intersection of TPR and TNR curves. An example of this procedure for determining τ^*_{LLS} is displayed in Figure 4.3.

In the next stage, we aim to convert the log likelihood decision scores to bounded posterior probability values using a sigmoid function. This function is denoted by Probability Decision Score (PDS) and is expressed by

$$PDS(\overline{LLS}) = \frac{1}{1 + \exp(-m(\overline{LLS} - c))} \tag{4.13}$$

To calculate the model parameter c, we require that this function be equal to 50% probability for τ^*_{LLS}, hence $c = \tau^*_{LLS}$. To estimate m, we set a fixed probability level PDS_{min} (e.g., 5%, 10%) for the smallest value $\overline{LLS}_{min}$.

$$m = \frac{1}{\tau^*_{LLS} - \overline{LLS}_{min}} \ln\left(\frac{100 - PDS_{min}}{PDS_{min}}\right) \tag{4.14}$$

In Figure 4.3 we display the graph of PDS versus LLS for one experiment. We can use PDS to express margins of uncertainty for classification in percentiles.

4.2 Optimization Parameters

Here we discuss implementation topics and list options for the SOCP method that may affect convergence to the solution.

4.2.1 Nonlinear Constraints

The SOCP method implements linear and/or nonlinear constraints. We denote by $c(x)$ and $ceq(x)$ the matrices of nonlinear inequality and equality constraints at x. The SOCP

Algorithm 3 Block based sparse representation

tp

1: **Input**: Training and test images.

2: The images presented as

$$I = [B^1, B^2, ..., B^{NB}] \tag{4.15}$$

for NB blocks, and the dictionary is

$$D^j = [bv^j_{1,1}, bv^j_{1,2}, \ldots, bv^j_{k,s_k}]. \tag{4.16}$$

3: We apply each block as a image and apply to SRC method to solve for x,

$$\hat{x}^j = \arg\min_x ||x^j||_1 \quad s.t. \quad ||D^j x - y^j||_2 \leq \varepsilon. \tag{4.17}$$

4: Predict class label by using decision function

5: BBMAP decision function

$$\widehat{\omega}_{BBMAP} = \mathcal{F}_{BBMAP}(\widehat{x}) \doteq \arg\max_i pr(\omega_i|\widehat{x}) \tag{4.18}$$

$$pr(\omega_i|\widehat{x}) = \sum_j^{NB} ND_{\omega_i^j}/NB \tag{4.19}$$

BBLL-R decision function

$$LLS^j(\widehat{x}) = -\log \frac{r^j_m(\widehat{x})}{r^j_n(\widehat{x})} \tag{4.20}$$

BBLL-S decision function

$$LLS^j(\widehat{x}) = -\log \frac{\|\delta_m(\widehat{x}^j)\|_1}{\|\delta_n(\widehat{x}^j)\|_1} \tag{4.21}$$

$$\widehat{\omega}_{BBLL} = \mathcal{F}_{BBLL}(\widehat{x}) \doteq \begin{cases} m\text{th class}, & \text{if } \overline{LLS(\widehat{x})} \geq \tau_{LLS} \\ n\text{th class}, & \text{otherwise} \end{cases}. \tag{4.22}$$

6: Sigmoid function (PDS) to value decision scores to bounded posterior probability,

$$PDS(\overline{LLS}) = \frac{1}{1 + \exp(-m(\overline{LLS} - c))} \tag{4.23}$$

7: **Output:** $\widehat{\omega}$ for each block and voting for decision of each image.

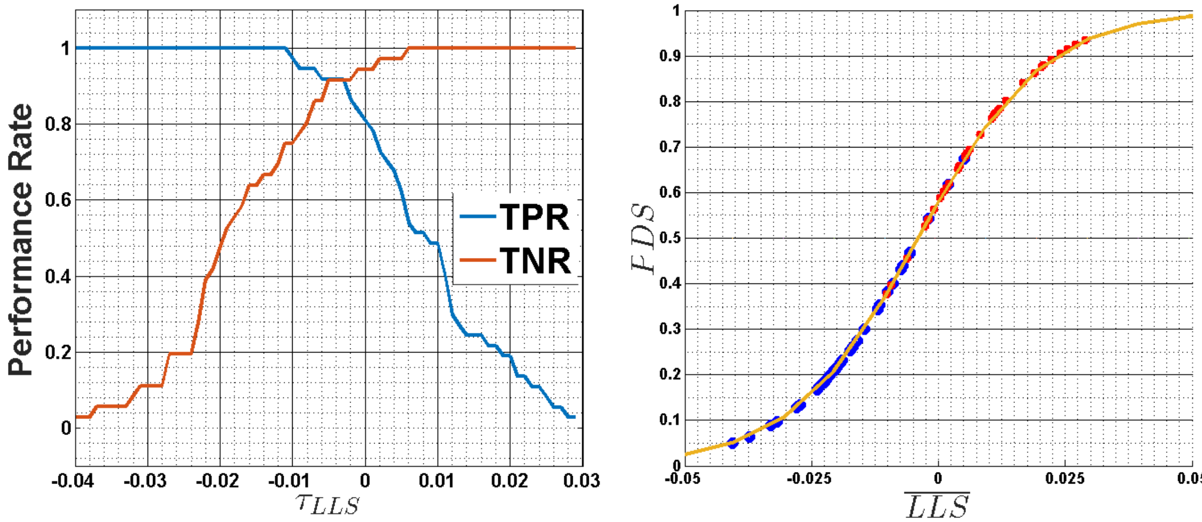

Figure 4.3: An example of TPR and TNR curves versus τ_{LLS} for determining $\tau^*_{LLS} = c$ (left) and the sigmoid probability decision score PDS after calculating the parameters m, c for (4.13) (right).

seeks to satisfy $c(x) \leq 0$, and $ceq(x) = 0$ for all x, respectively.

In our problem, we have the inequality constraints

$$\| Ax - b \|_2 \leq \epsilon \tag{4.24}$$

and we have defined the $c(x)$ as

$$\| Ax - b \|_2 - \epsilon \leq 0 \tag{4.25}$$

In this equation, ϵ expresses the level of uncertainty or noise in the representation. In the imaging context, this may be caused by various types of error in measurements including imaging artifacts. By using the nonlinear inequality constrains, the sparsity of solution is significantly improved.

4.2.2 Lower Bound

The lower bound is a real vector or a real array, for all i, such that $x(i) \geq lb(i)$. We are solving l_1 norm of x, so the lower bound need to be positive vector for linear programming, but not necessary for SOCP.

4.2.3 Stopping Criteria

The following tolerance parameters are stopping criteria for SOCP. The different tolerance parameters may be related.

ConstraintTolerance - TolCon

The ConstraintTolerance is the tolerance on the constraint violation, it is a positive scalar. Constraintrance is an upper bound of constraints' magnitude. If we use ConstraintTolerance in the SOCP and it returns a point with $c(x) > ConstraintTolerance$, then the constraints are violated at point x in SOCP. The iterative attempts will continue even if the ConstraintTolerance is not satisfied, unless there are some other reasons that halt it.

MaxFunctionEvaluations - MaxFunEvals

This stands for the maximum number of function evaluations allowed. F-count is defined as function count, if there has constraints, the F-count is the number of points that function evaluations, that can be smaller than the MaxFunctionEvaluations.

MaxIterations - MaxIter

MaxIterations is the maximum number of iterations allowed. The algorithm may stop before reaching the value of MaxIterations because of the other values of tolerances that may stop the solver before.

OptimalityTolerance - TolFun

OptimalityTolerance is termination tolerance on the first-order optimality. The first-order optimality is a measure of distance of point x to the optimal point. It is a necessary condition but not sufficient condition.

StepTolerance - TolX

The StepTolerance is a positive scalar that is a lower bound for the step of the solver on x at iteration i. So the solver will stop when $\|x_i - x_{i+1}\| < TolX$.

Chapter V: CLINICAL APPLICATION: OSTEOPOROSIS DIAGNOSIS

The main goal of our experiments is to validate the hypothesis that the proposed ensemble of block-based sparse classifiers improves the classification performance of conventional sparse representation. The second goal is to compare the proposed technique with texture-based and Bag of Keypoints-based classification. Finally, we compare the performances of the two decision functions BBMAP and BBLL. We test the predictive and generalization capability of our system for two diverse and significant clinical applications; osteoporosis diagnosis and breast lesion characterization. Here we describe the application of our system to osteoporosis diagnosis, we report the classification results and discuss our findings.

5.1 Background: Osteoporosis Diagnosis

Osteoporosis is a skeletal disorder characterized by decreased bone strength that may lead to susceptibility of fracture [7]. Aerial Bone Mineral Density (BMD) is computed in dual-energy X-ray absorptiometry (DXA) scans to diagnose osteoporosis [41]. However, BMD can predict fracture with only 60% accuracy. Analysis of trabecular bone microarchitecture can significantly improve the prediction rates, but this information requires bone biopsy with histomorphometric analysis. The task of obtaining trabecular bone microarchitecture information by noninvasive methods is a nontrivial scientific problem [53]. Previous approaches to evaluating bone structure on radiographs by 2D texture analysis were reported in [41, 57, 99]. Moreover, in [45, 44] the authors propose to use 2D texture analysis to characterize 3D bone microarchitecture.

Diagnosis of osteoporosis using bone radiograph scans presents some challenges, mainly because images of osteoporotic and healthy subjects are visually very similar. Therefore, early diagnosis can effectively predict fracture risk and prevent the disease [68, 39]. The texture feature computation system produced a classification accuracy of 79.3% and a receiver operating characteristic area of 81% over 116 images. These results are particularly promising when we consider the level of difficulty of the specific dataset.

5.2 Data Description

Our purpose is to distinguish between healthy and osteoporotic subjects. The TCB challenge dataset contains labeled digital radiographs of 87 healthy and 87 osteoporotic subjects for training and testing (available online in http://www.univ-orleans.fr/i3mto/data, last access in 05/2018). The calcaneus trabecular bone images in the dataset have an ROI size of 400×400 pixels. A more detailed description of the dataset is provided in [68, 39]. The experimental procedures involving human subjects were approved by the Institutional Review Board of the institution that provides the data.

5.3 Texture-based Classification

In the performance evaluation of conventional texture-based techniques of Chapter II, we calculated 723 texture-related features. We selected features using correlation-based feature selection with best first search (CFS-BF), correlation-based feature selection with genetic algorithm search (CFS-GA) as described in Section 2.3. We conducted leave-one-out cross-validation and 10-fold cross-validation experiments reported in Tables 5.1 and 5.2. We note that CFS-GA yields an overall better performance than CFS-BF, IG and no-feature selection on leave-one-out cross-validation. This implies that CFS-GA effectively selects distinguishing features from the entire set. Among the tested classifiers, Bagging accomplished the highest performance with an ACC of 67.8% on leave-one-out cross-validation. We performed ROC experiments using CFS-BF feature selection and display the graphs in Fig. 5.1 and 5.2. We note that NB yielded the largest area under the curve for the leave-one-out experiment followed by Bagging. In 10-fold cross-validation, RF reached the top ACC of 66.7% and the top AUC of 67.5% with no dimensionality reduction.

Table 5.1: Bone texture characterization classification performance (leave-one-out cross-validation)

FSM	CL	TPR	TNR	ACC	AUC	Dimension
No	NB	57.5	64.4	60.9	63.5	723
	BN	58.6	65.5	62.1	65.3	
	RF	65.5	64.4	64.9	67.2	
	Bagging	70.1	64.4	67.3	68.1	
CFS-GA	NB	63.2	64.4	63.8	67.3	101
	BN	66.7	62.1	64.4	70.4	
	RF	67.8	65.5	66.7	68.2	
	Bagging	70.1	65.5	**67.8**	65.0	
CFS-BF	NB	71.3	57.5	64.4	70.9	20
	BN	64.4	66.7	65.5	69.9	
	RF	60.9	67.8	64.4	68.4	
	Bagging	66.6	67.8	67.2	70.5	

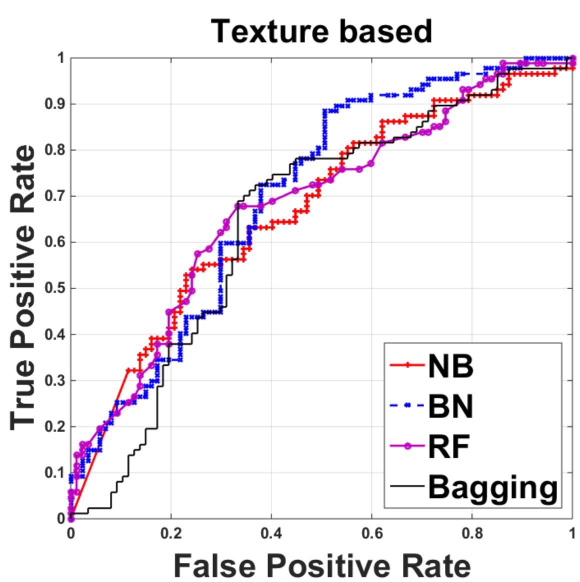

Figure 5.1: ROC curves for bone characterization using conventional (non-sparse) texture-based techniques (Bagging, BN, NB, and RF) with leave-one-out cross-validation.

Table 5.2: Bone texture characterization classification performance (10-fold cross-validation)

FSM	CL	TPR	TNR	ACC	AUC	Dimension
No	NB	57.5	60.9	59.2	63.1	723
	BN	56.3	65.5	60.9	63.6	
	RF	64.4	69	**66.7**	67.5	
	Bagging	63.2	64.4	63.8	66.8	
CFS-GA	NB	64.4	60.9	62.9	67.0	226
	BN	62.1	66.7	64.4	65.5	
	RF	66.7	62.1	64.4	68.8	
	Bagging	62.1	67.80	64.9	68.3	
CFS-BF	NB	69	55.2	62.1	67.1	20
	BN	58.6	59.8	59.2	65.6	
	RF	64.4	59.8	62.1	66.1	
	Bagging	64.4	63.2	63.8	65.8	

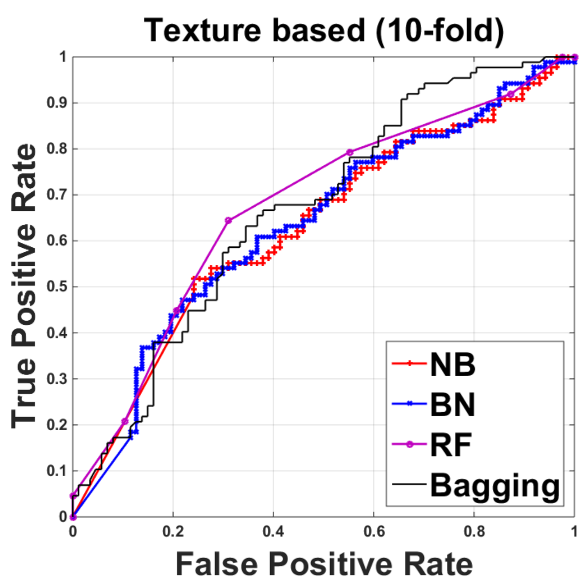

Figure 5.2: ROC curves for bone characterization using conventional (non-sparse) texture-based techniques (Bagging, BN, NB, and RF) with 10-fold cross-validation.

5.4 Bag-of-Keypoints Classifier

We performed cross-validation experiments for the Bag of Keypoints technique. The results showed that BoK was able to separate successfully healthy from osteoporotic subjects with an ACC of 99.3% leave-one-out cross-validation as displayed in Table 5.3. This very high accuracy may be attributed to the extraction of discriminant features from the textured areas. Also, the employed SVM model is known to address data complexity caused by non-linearity and high dimensionality.

Table 5.3: Classification performance for bone characterization using Bag of Keypoints classification (leave-one-out cross-validation)

TPR	TNR	ACC	AUC
98.6	100	99.3	100

5.5 Conventional SRC

We then evaluated the performance of the conventional SRC method described in Chapter III. We utilized multiple undersampling factors to address convergence to infeasible solutions mostly caused by linearly dependent vectors that yielded different classes. In Table 5.4 we show results from the top performing experiments producing 59.2% classification accuracy for resampling of 1/20, corresponding to feature dimensionality of 400 using leave-one-out cross-validation. The ROC curves in Fig. 5.3 indicate that a higher degree of downsampling yields shorter and more numerically tractable feature dimensionality, but it also diffuses the textural information. We also applied conventional SRC to the texture feature set produced in Section 2.3 and the classification accuracy was 71.7%. This result also implies the limited separation capability of a generic texture feature set. In Table 5.5 and Fig. 5.4 we display results using 10-fold cross-validation. The ACC for this experiment was 56.5% and the AUC was 60.1%.

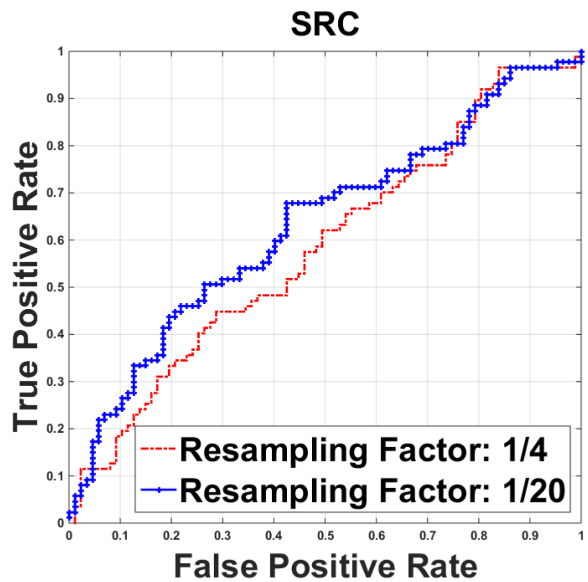

Figure 5.3: ROC curves for bone characterization using conventional SRC classification using LOO CV.

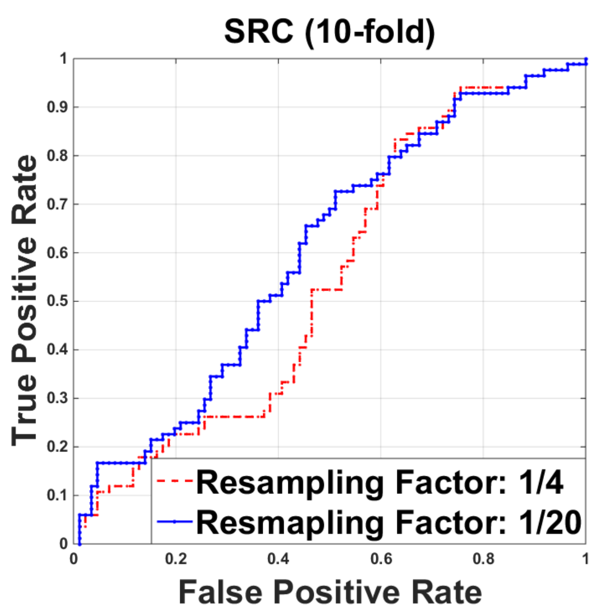

Figure 5.4: ROC curves for bone characterization using conventional SRC classification using 10-fold cross-validation.

Table 5.4: Classification performance for bone texture characterization sparse classifiers using LOO CV.

Size of Block	TPR (%)	TNR (%)	ACC (%)	AUC (%)
400×400 (undersamp. 1/4)	55.2	54	54.6	58.4
400×400 (undersamp. 1/20)	57.5	60.9	59.2	63.4

Table 5.5: Classification performance for bone texture characterization sparse classifiers using 10-fold CV.

Size of Block	TPR (%)	TNR (%)	ACC (%)	AUC (%)
400×400 (undersamp. 1/4)	44	53.5	48.8	54.7
400×400 (undersamp. 1/20)	53.6	59.3	56.5	60.1

5.6 Integrative Sparse Classification

Next, we evaluated the performance of our block-based ensemble of sparse classifiers. We employed block sizes ranging from 100×100 pixels to 10×10 pixels to observe the impact of this variable on the classification performance. We repeated these experiments using the BBMAP and BBLL decision functions in this setting. We show our leave-one-out cross-validation performance in Table 5.6. The experiment with block size 25×25 pixels that led to 256 classifiers performed the best classification of 100% by the BBMAP and BBLL techniques. These results imply 9.5% improvement of our method over the traditional SRC method. The block size with 10×10 also produced 100% accuracy and 100% AUC. Figure 5.5 displays the ROC graphs for varying block sizes using BBMAP and BBLL decision functions. We observe that the largest AUC was obtained by use of25×25 and 10×10 blocks. We also note the improvement in classification performance compared with conventional SRC results that are depicted in Figure 5.3. These results suggest that the block-based approach finds more accurate sparse solutions than the conventional SRC approach and improves the classifier performance. A reason for the improved group separation may be that the block-based ensemble technique employs multiple learners of over-complete dictionaries that are

Table 5.6: Classification performance for bone texture characterization using ensembles of block-based sparse classifiers (LOO CV).

Size of Block	BBMAP-R				BBLL-R ($\tau_{LLS} = 0$)			
	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC
100×100	65.5	67.8	66.7	71.4	85.1	82.8	83.9	87.7
50×50	93.1	81.6	87.4	91.3	98.6	90.8	94.8	97.3
25×25	100	100	100	100	100	100	100	100
10×10	100	100	100	100	100	100	100	100

Size of Block	BBMAP-S				BBLL-S ($\tau_{LLS} = 0$)			
	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC
100×100	51.7	57.5	54.6	59.0	59.8	46.0	52.9	56.9
50×50	97.7	43.7	70.7	76.5	88.5	12.6	50.6	58.7
25×25	100	100	100	100	100	100	100	100
10×10	100	100	100	100	100	100	100	100

MaxIter=10. solver method is LP (top table) and solver method is SOCP (bottom table).

more amenable to sparse coding and representation. In addition, we estimated the statistical significance of the differences between the AUC values of BBLL with optimized threshold τ^*_{LLS} and BBMAP by applying DeLong's statistical test between the ROCs produced by BBMAP and BBLL. The p-values for block sizes of 100×100, 50×50, 25×25 and 10×10 were 0.47, 0.66, 0 and 0 respectively, suggesting significant differences for block sizes of 100×100, 50×50, 25×25 and 10×10.

We also performed 10-fold, 20-fold and 30-fold cross-validation experiments for variable block sizes. We display the classification results in Tables 5.7, 5.8,and 5.9 and the ROC curves in Fig. 5.6, 5.7, and 5.8. For 10-fold CV, the best accuracy of 60.59% was obtained for 25×25 block size, and the area under the curve was 62.46%. In the Tables 5.7, 5.8 and 5.9 we observe that the highest accuracy with block size 25×25, was 70.67% using 30-fold cross-validation. 30-fold cross-validation has 6 test samples in each fold for this data set. The corresponding area under the curve was 74.36%. We estimated the AUC values of BBLL with optimized threshold τ^*_{LLS} and BBMAP by applying DeLong's statistical test between

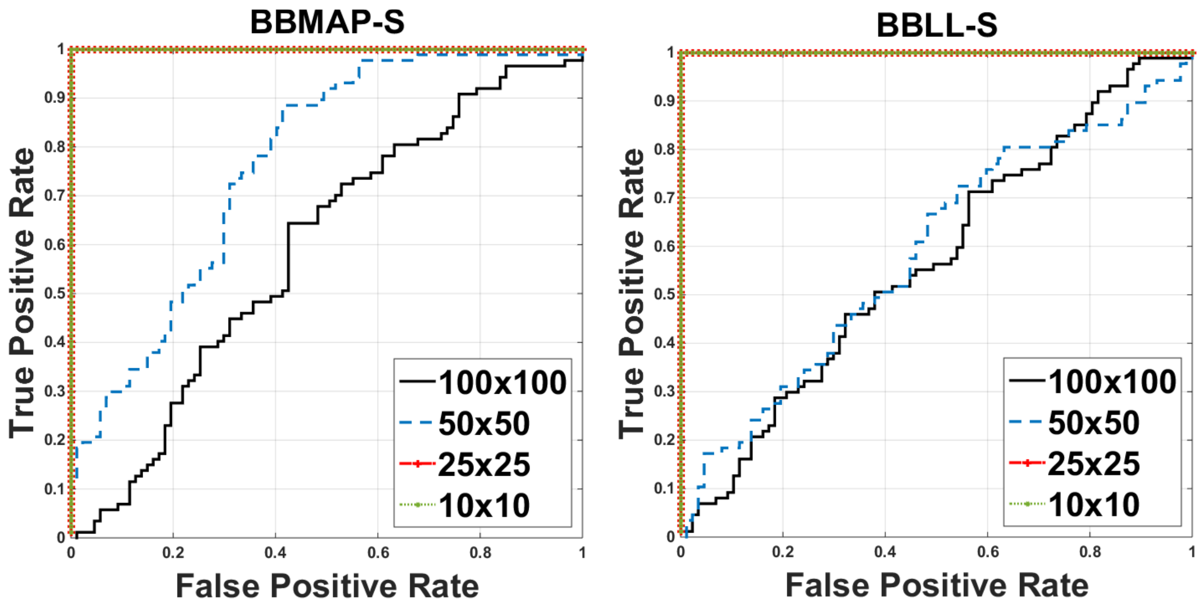

Figure 5.5: ROC curves for bone characterization using the proposed block-based ensemble method with BBMAP (left), and BBLL (right) decision function for leave-one-out cross-validation.

the ROCs produced by BBMAP and BBLL as well for k-fold cross-validation. The p-values for block sizes of 100×100, 50×50, 25×25 and 10×10 were 0.59, 0.52, 0.003 and 0.0016 respectively for 10-fold cross-validation. With 20-fold cross-validation the p-values for block sizes of 100×100, 50×50, 25×25 and 10×10 were 0.086, 0.96, 0.79 and 0.052 respectively. For 30-fold cross-validation, the p-values for block sizes of 100×100, 50×50, 25×25 and 10×10 were 0.69, 0.42, 0.036 and 0.0004 respectively.

Table 5.7: Classification performance for bone texture characterization using ensembles of block-based sparse classifiers (10-fold CV)

Size of Block	BBMAP-S				BBLL-S ($\tau_{LLS} = 0$)				BBLL-S ($\tau_{LLS} = \tau^*_{LLS}$=0.05)			
	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC
100×100	47.62	51.16	49.41	54.43	54.24	45.35	45.29	50.37	19.1	84.88	52.35	49.36
50×50	72.62	41.86	57.06	61.36	83.33	29.07	55.88	62.38	22.62	74.42	48.82	50.01
25×25	59.52	59.3	59.41	62.47	59.52	59.3	59.41	62.47	59.52	61.63	60.59	62.46
10×10	59.52	59.3	59.41	62.47	59.52	59.3	59.41	62.47	0	100	50.59	59.65

Table 5.8: Classification performance for bone texture characterization using ensembles of block-based sparse classifiers (20-fold CV)

Size of Block	BBMAP-S				BBLL-S ($\tau_{LLS} = 0$)			
	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC
100×100	51.85	51.9	51.88	55.98	50.62	46.84	48.75	52.51
50×50	80.25	35.44	58.13	60.4	88.89	13.92	51.88	51.34
25×25	77.78	53.16	65.63	67.29	79.01	53.16	66.25	66.48
10×10	79.01	49.37	64.38	66.32	79.01	49.37	64.38	66.32

Table 5.9: Classification performance for bone texture characterization using ensembles of block-based sparse classifiers (30-fold CV)

Size of Block	BBMAP-S				BBLL-S ($\tau_{LLS} = 0$)				BBLL-S ($\tau_{LLS} = \tau^*_{LLS} = 0.004$)			
	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC
100×100	47.44	54.17	50.67	54.26	60.26	55.56	58.00	61.25	25.64	70.83	47.33	54.81
50×50	84.62	37.5	62.0	64.05	88.46	18.06	54.67	58.44	35.9	72.22	53.33	54.27
25×25	71.79	66.67	69.33	70.23	71.79	66.67	69.33	70.23	71.79	69.44	70.67	74.36
10×10	71.79	66.67	69.33	70.23	71.79	66.67	69.33	70.23	0	100	48	56.52

MaxIter=10

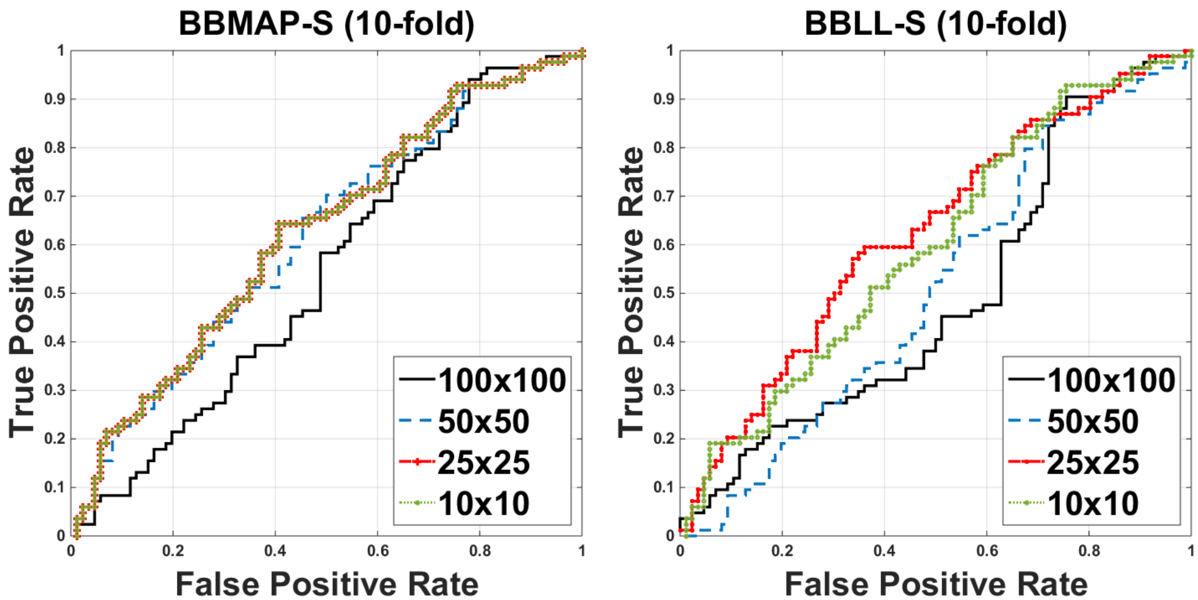

Figure 5.6: ROC curves for bone characterization using the proposed block-based ensemble method with BBMAP (left), and BBLL (right) decision function with 10-fold cross-validation.

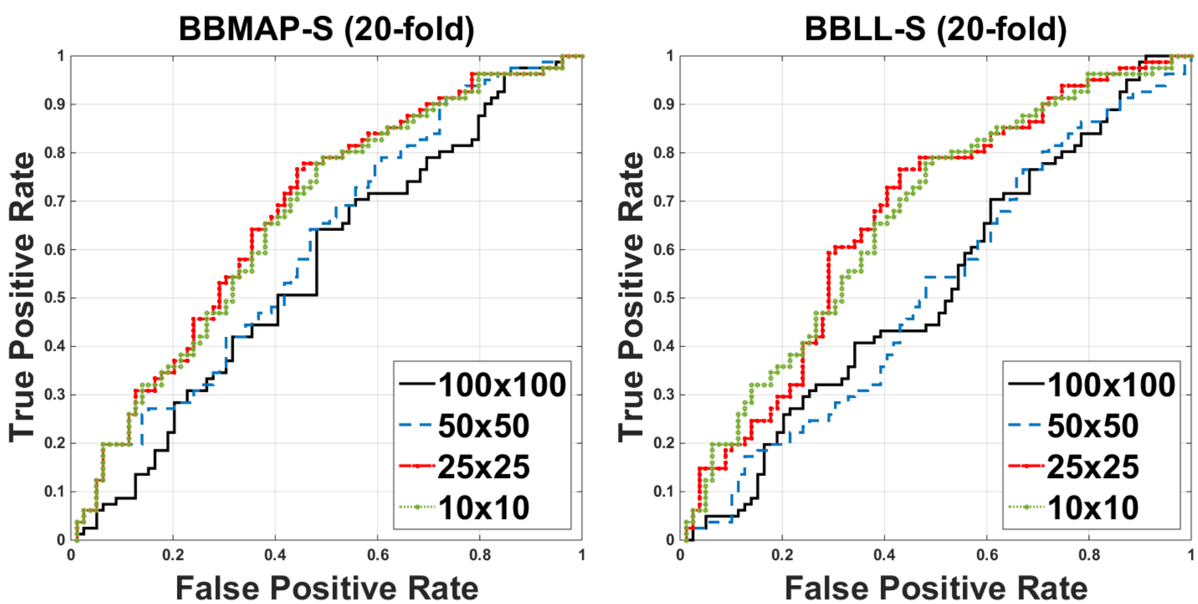

Figure 5.7: ROC curves for bone characterization using the proposed block-based ensemble method with BBMAP (left), and BBLL (right) decision function with 20-fold cross-validation.

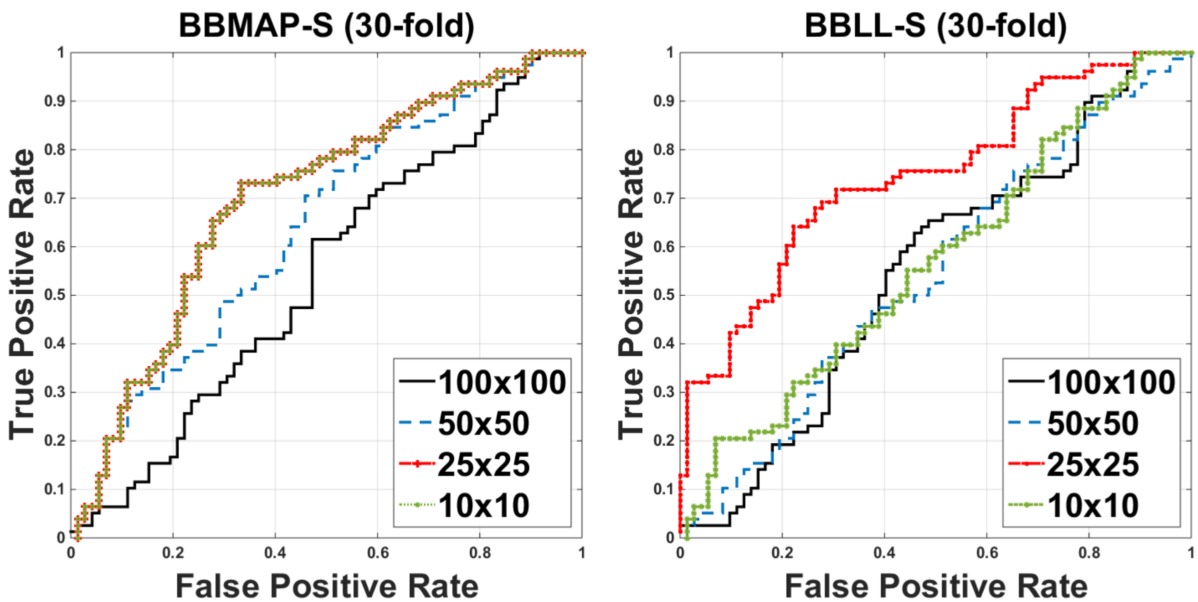

Figure 5.8: ROC curves for bone characterization using the proposed block-based ensemble method with BBMAP (left), and BBLL (right) decision function with 30-fold cross-validation.

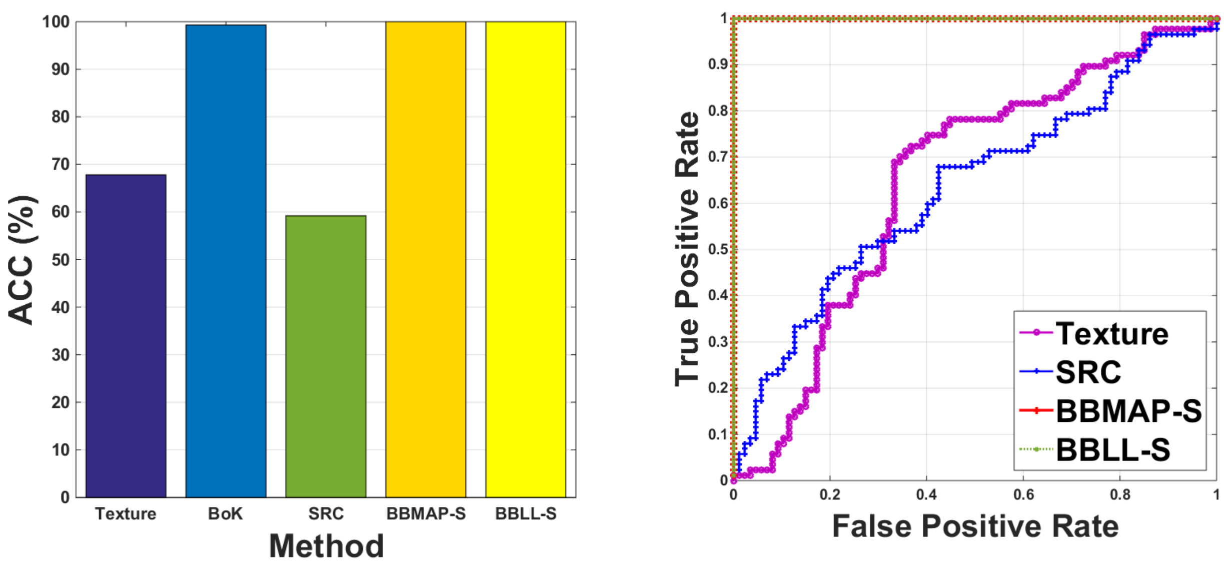

Figure 5.9: Graphs of ACC values versus ROI size (left) and the corresponding average ACC for each method (right) produced by BoK, SRC, BBMAP and BBLL using LOO CV.

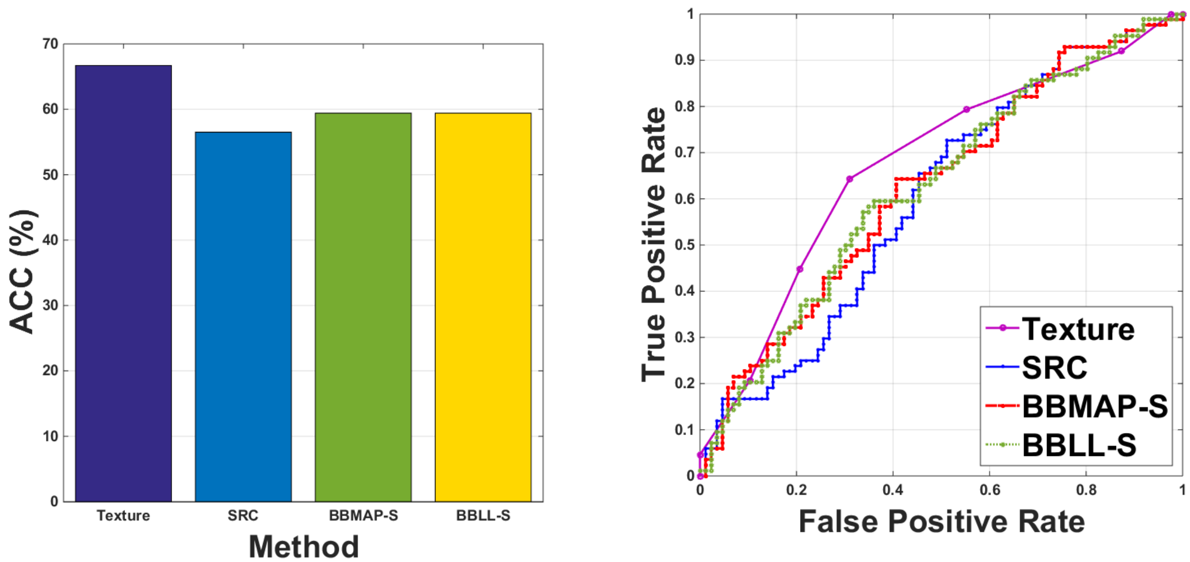

Figure 5.10: Graphs of ACC values versus ROI size (left) and the corresponding average ACC for each method (right) produced by BoK, SRC, BBMAP and BBLL using 10-fold CV.

Chapter VI: CLINICAL APPLICATION: BREAST LESION CHARACTERIZATION

The second clinical application that we developed in this work is breast lesion characterization as benign or malignant using digitized or digital mammograms. We outline the significance and background of this application, then describe the data and experiments. We also discuss the experimental results produced by our approach and other approaches that we use for comparisons as in the previous chapter. We perform analysis for various ROI sizes to explore the relationship of ROI size with classification accuracy.

6.1 Background: Breast Lesion Characterization

Early detection and characterization of breast lesions is important for increasing the life expectancy and quality of health of women. Because of its significance, automated detection and diagnosis of breast cancer is a popular field of research [40, 66, 67, 93, 58, 63, 48, 73, 64]. Mammograms can help to find breast cancer at an early stage. Automated diagnosis is very challenging in this application as well.

6.2 Data Description

One of the most reliable methods for diagnosis and early prediction of breast cancer is using X-ray mammographic test[10, 60]. In general, there are two view for each breast: the craniocaudal (CC) view, this is view from top to bottom of breast; another view is mediolateral oblique (MLO) view, ML is from middle to side and LM is from side to middle view. The images acquired as x-ray films, such as film screen mammogram are converted into TIFF and digital imaging and communications in medicine (DICOM) format. Mammograms show the masses, calcifications, architectural distortion of breast tissue, and symmetries [63].

The MIAS database has 161 cases, and 322 digitized MLO PGM images with benign, malign lesions and normal images. The annotations includes the information of center and radius of the area of interest (ROI). To obtain good quality of a mammographic image high contrast resolution is needed, at least 10 bits, i.e. 1024 gray levels [58]. Although low contrast resolution is not well suited for detection of microcalcifications (MCCs), 100% accuracy has been reached on MIAS data in [51]. Bancoweb has 12 bits (4,096) contrast resolution. For contrast resolution higher than 14 bits (16,384) little differences in the performances of most CAD schemes have been observed [82]. There are 1,400 images from 320 patients, the spatial resolutions of the images are 0.085 mm or 0.150 mm.

We validated the separation of the breast lesions data set into two classes: malignant and benign. The training and testing data were obtained from the Mammographic Image Analysis Society (MIAS) database that is available online [66, 58]. The resolution of the mammograms is 200 micron pixel edge, and size of each image is 1024×1024 px after clipping/padding. MIAS contains 322 MLO scans from 161 subjects. The data is categorized into groups of healthy subjects, subjects with benign, and subjects with malignant lesions. Our goal is to characterize the lesion type, therefore we utilized 68 benign and 51 malignant mammograms for performance evaluation.

Because our proposed method performs block-wise analysis, we need to ensure that the majority of the blocks cover the lesion to improve the accuracy. Hence we designed our system so that the lesion ROI sizes are greater than or equal to the analysis ROI size. In this experiment, we determined from the provided metadata the centroid and radius of each lesion. We used these two values to determine a minimum bounding square ROI for each scan. We trained and tested all classifiers on these ROI patches centered at the lesion centroid. In order to evaluate the classification performance with respect to the lesion size, we performed validation experiments on variable minimum ROI sizes. The selected ROI sizes were $48 \times 48, 56 \times 56$, and 64×64. For each ROI size we selected subsets of the dataset that met the minimum lesion radius criteria described above. The numbers of benign and malignant images for each ROI size are displayed in Table 6.1.

Table 6.1: MIAS Dataset Information by ROI size

ROI	Benign	Malignant
64×64	36	37
56×56	43	42
48×48	48	45

6.3 Texture-based Classification

In Tables 6.2, 6.3, 6.4 and Fig. 6.2 we display texture-based classification results computed for lesions with 48×48, 56×56 and 64×64 pixels minimum ROI size that performed better than the other ROI sizes using leave-one-out cross-validation. The feature dimensionality in this experiment is 451. The dimensionality of texture feature set is different from that of bone characterization experiments because (i) we did not utilize the co-occurrence features due to several ROIs being smaller than the required size, (ii) 14 edge histogram features were always zeros and not used in analysis, (iii) 2 additional features produced numerical errors such as division by zero.

In Table 6.2, display the classification results after on 48×48 ROI size. We observe that the Bagging technique achieves the best performance with an ACC of 63.4% and AUC of 58.4% and 62.1% crosponding to CFS-GA and no feature selection. Bayes Network, Naive Bayes and Random Forest techniques produced lower ACC values than Bagging, at 61.3%, 57.8% and 61.3%, respectively. Fig. 6.1 displays the ROC graphs for the CFS-GA feature selection. This figure confirms that Bagging produced the largest AUC for the leave-one-out experiment, followed by Random Forest.

In Table 6.3, we present results for 56×56 ROI size with leave-one-out cross-validation, and the best performances is 58.8% by using Bagging with no feature selection, and Bayes Network with CFS-GA feature selection method. The corresponding areas under the curve are 59.5%, 33.9% and 36.9%.

Table 6.2: ROI images of size 48 × 48 classification performance (leave-one-out cross-validation)

FSM	CL	TPR	TNR	ACC	AUC	Dimension
No	NB	66.7	22.2	45.2	45.7	451
	BN	100	20	61.3	37.8	
	RF	66.7	46.7	57	53.0	
	Bagging	64.6	62.2	**63.4**	62.1	
CFS-GA	NB	62.5	26.7	45.2	43.4	55
	BN	100	20	61.3	37.8	55
	RF	70.8	51.1	61.3	58.9	49
	Bagging	62.5	64.4	**63.4**	58.4	41
CFS-BF	NB	50	53.3	51.6	50.9	2
	BN	100	20	61.3	37.8	(330,402)
	RF	70.8	44.4	58.1	54.4	
	Bagging	64.6	55.6	60.2	61.3	

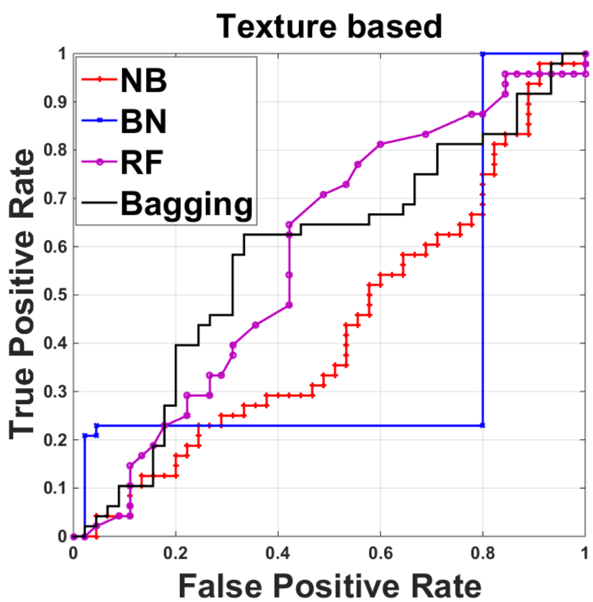

Figure 6.1: ROC curves for breast lesion characterization using conventional (non-sparse) texture-based techniques (Bagging, BN, NB, and RF) with leave-one-out cross-validation.

Table 6.3: ROI images of size 56×56 classification performance (leave-one-out cross-validation)

FSM	CL	TPR	TNR	ACC	AUC	Dimension
No	NB	74.4	28.6	51.8	46.7	451
	BN	90.7	21.4	56.5	33.4	
	RF	58.1	54.8	56.5	54.4	
	Bagging	60.5	57.1	**58.8**	59.5	
CFS-GA	NB	76.7	31	54.1	46.6	55
	BN	93	21.4	57.6	33.9	
	RF	58.1	54.8	56.5	51.7	
	Bagging	55.8	50	52.9	51.7	
CFS-BF	NB	53.5	54.8	54.1	58.8	2
	BN	90.7	21.4	56.5	33.4	(330,402)
	RF	48.8	52.4	50.6	65.7	
	Bagging	55.8	57.1	56.5	63.8	

The same experiments were performed on 64×64 ROI size as well. In table 6.4, the best performances is 58.9% by using Bayes Network with CFS-GA feature selection method, the corresponding area under the curve is 71.9%. Overall the best performance using leave-one-out cross-validation is obtained by 48×48 ROI size and accuracy is 63.4%.

Next, we use 10-fold cross-validation for all the ROI sizes and present the results in Tables 6.5, 6.6, and 6.7 and Fig. 6.3. The best accuracy with 10-fold cross-validation was 71.2% and corresponding area under the curve was 69.8% for 64×64 ROI size, obtained by no feature reduction and Random Forest classifier. This is 7.8% higher accuracy than the best performance using leave-one-out cross-validation. For 56×56 ROI size, the best accuracy was 64.5% and area under the curve was 65.6% and the top performances for 48×48 ROI size were 64.7% accuracy and 65.2% area under the curve.

6.4 Bag-of-Keypoints Classifier

The cross-validation experiments of BoK for each ROI size are displayed in Table 6.8. We note that this approach produces high classification rates for most of the ROI sizes and

Table 6.4: ROI images of size 64 × 64 classification performance (leave-one-out cross-validation)

FSM	CL	TPR	TNR	ACC	AUC	Dimension
No	NB	66.7	35.1	50.7	54.2	451
	BN	75	5.4	39.7	40.2	
	RF	58.3	54.1	56.2	56.5	
	Bagging	61.1	55.8	58.9	59.0	
CFS-GA	NB	75	43.2	**58.9**	71.9	31
	BN	22.2	70.3	46.6	21.6	9
	RF	50	54.1	52.1	47.8	31
	Bagging	52.8	62.2	57.5	50.4	134
CFS-BF	NB	58.3	21.6	39.7	28.2	4
	BN	75	5.4	39.7	40.2	(127,302,
	RF	58.3	62.2	57.5	50.4	406,409)
	Bagging	50	51.4	50.7	44.4	

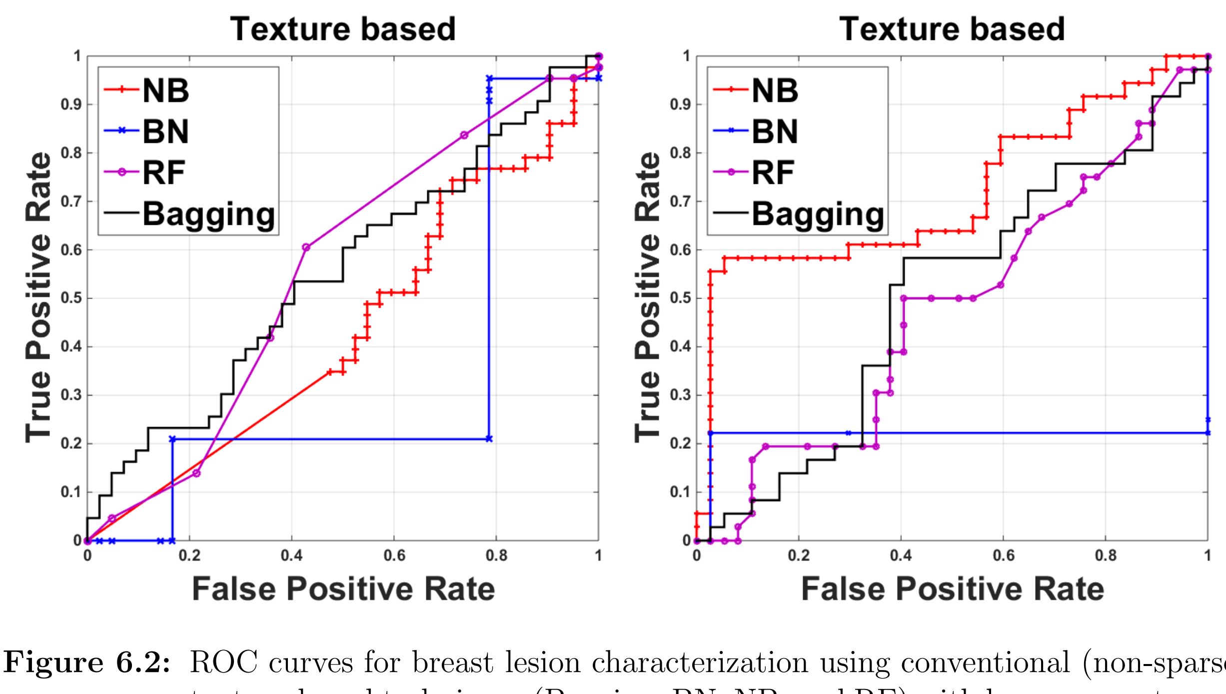

Figure 6.2: ROC curves for breast lesion characterization using conventional (non-sparse) texture-based techniques (Bagging, BN, NB, and RF) with leave-one-out cross-validation. left top is ROI size 48 × 48, left right is ROI size 56 × 56 and bottom is ROI size 64 × 64.

Table 6.5: ROI images of size 48×48 classification performance (10-fold cross-validation)

FSM	CL	TPR	TNR	ACC	AUC	Dimension
No	NB	58.3	20	39.8	46.3	451
	BN	58.3	53.3	55.9	59.1	
	RF	81.3	40	61.3	57.3	
	Bagging	70.8	57.8	**64.5**	65.6	
CFS-GA	NB	68.8	35.6	52.7	51.0	55
	BN	58.3	53.3	55.9	59.1	
	RF	70.8	53.3	62.4	60.0	
	Bagging	64.6	67.8	61.3	64.0	
CFS-BF	NB	33.3	60	46.2	50.9	2
	BN	58.3	53.3	55.9	59.1	(330,402)
	RF	60.4	55.6	58.1	55.0	
	Bagging	60.4	57.8	59.1	62.5	

Table 6.6: ROI images of size 56×56 classification performance (10-fold cross-validation)

FSM	CL	TPR	TNR	ACC	AUC	Dimension
No	NB	67.4	31	49.4	39.9	451
	BN	72.1	28.6	50.6	47.2	
	RF	72.1	57.1	**64.7**	65.2	
	Bagging	72.1	57.1	**64.7**	65.2	
CFS-GA	NB	67.4	38.1	52.9	48.7	29
	BN	69.8	35.7	52.9	48.7	
	RF	62.8	52.4	57.6	54.0	
	Bagging	60.5	64.3	62.4	57.5	
CFS-BF	NB	51.2	40.5	45.9	39.1	4
	BN	72.1	28.6	50.6	47.1	330,452)
	RF	58.1	57.1	57.6	52.1	
	Bagging	62.8	50	56.5	52.9	

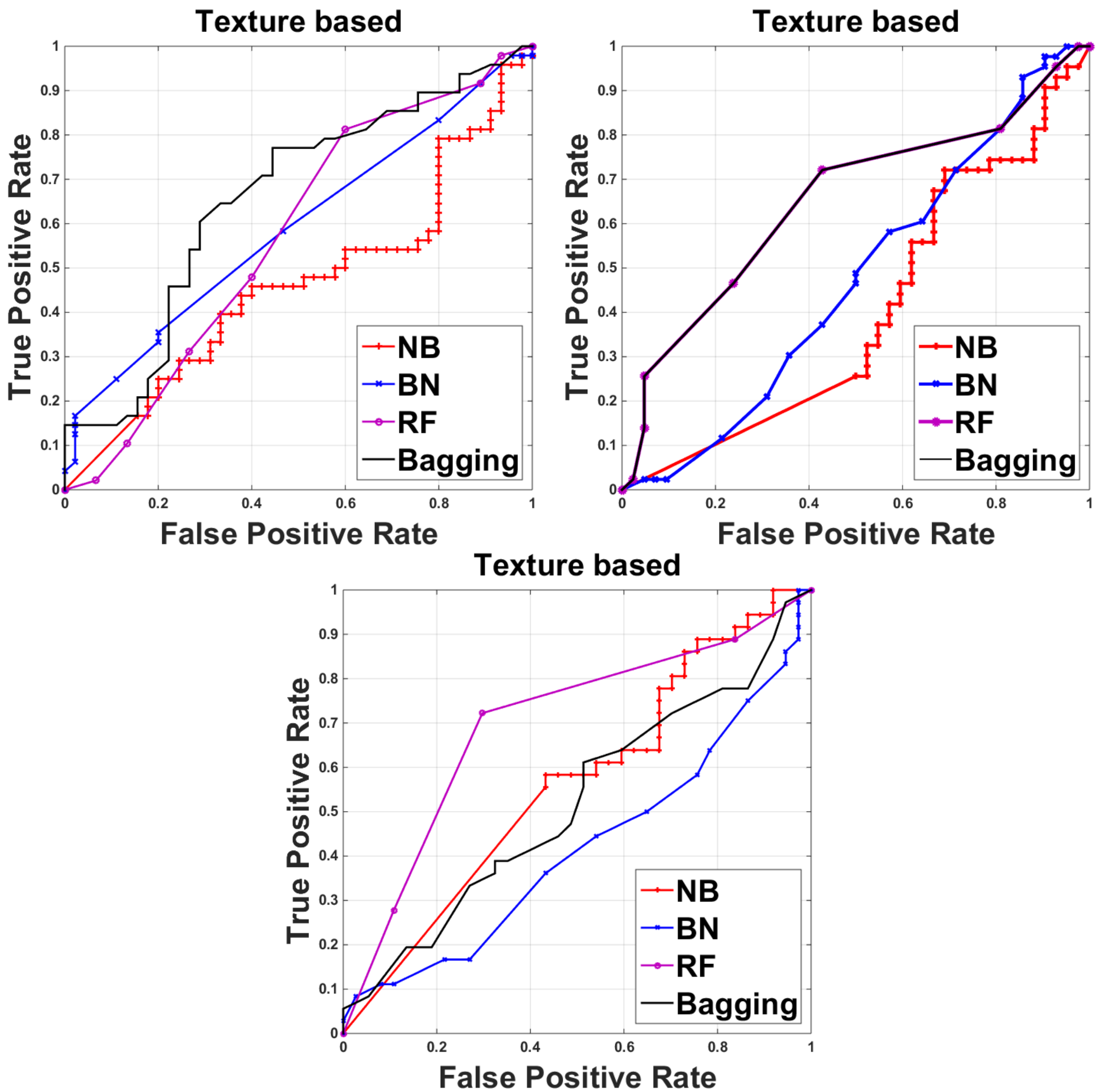

Figure 6.3: ROC curves for breast lesion characterization using conventional (non-sparse) texture-based techniques (Bagging, BN, NB, and RF) with leave-one-out cross-validation. left top is ROI size 48×48, left right is ROI size 56×56 and bottom is ROI size 64×64.

Table 6.7: ROI images of size 64 × 64 classification performance (10-fold cross-validation)

FSM	CL	TPR	TNR	ACC	AUC	Dimension
No	NB	63.9	37.8	50.7	56.3	451
	BN	58.3	24.3	41.1	41.7	
	RF	72.2	70.3	**71.2**	69.8	
	Bagging	61.1	48.6	54.8	51.8	
CFS-GA	NB	69.4	45.9	57.5	55.6	29
	BN	38.9	59.5	49.3	48.9	
	RF	50	67.6	58.9	62.0	
	Bagging	61.1	62.2	61.6	55.9	
CFS-BF	NB	63.9	32.4	47.9	43.2	4
	BN	61.1	24.3	42.5	44.2	330,452)
	RF	58.3	45.9	52.1	47.5	
	Bagging	58.3	48.6	53.4	47.7	

Table 6.8: Classification performance for breast lesion characterization using Bag of Key-points classification on ROIs (LOO CV).

ROI size	TPR	TNR	ACC	AUC
72×72	100	71.3	86.7	99.8
64×64	77.3	100	88.5	99.6
60×60	92.7	48.3	69.6	85.8
48×48	**100**	**98.3**	**99.1**	**100**
Summary	92.5± 10.7	79.5± 24.6	86.0± 12.2	96.3± 7.0

the top class separation with ACC of 99.1 was accomplished for ROI size of 48 × 48. We deduce that the extraction of discriminant features and use of SVM classification drives the very good results.

6.5 Conventional SRC

Furthermore, Table 6.9 lists the results produced by the conventional SRC method using leave-one out cross-validation. The top performance was obtained for ROI size of 56 × 56 at 65.9%. After comparing the Tables 6.2, 6.3, 6.4 and 6.9, 6.5, 6.6, 6.7 and Figs. 6.2 and 6.4, we conclude that texture-based classification produces more accurate classification rates than conventional SRC. Similarly to our bone characterization experiments, here we applied conventional SRC to the texture feature set produced in subsections 2.2, 2.3 and 2.4 and the top classification accuracy was 56.7% that indicates the limited separation capability of generic texture features.

Table 6.9: Classification performance for breast lesion characterization using conventional SRC on ROIs (LOO CV).

ROI size	TPR	TNR	ACC	AUC
64×64	32.4	72.2	52.1	50.2
56×56	47.6	51.2	49.4	47.3
48×48	53.3	41.7	47.3	46.4
Summary	44.4±10.8	55.0±15.6	49.6±2.4	48.0±2.0

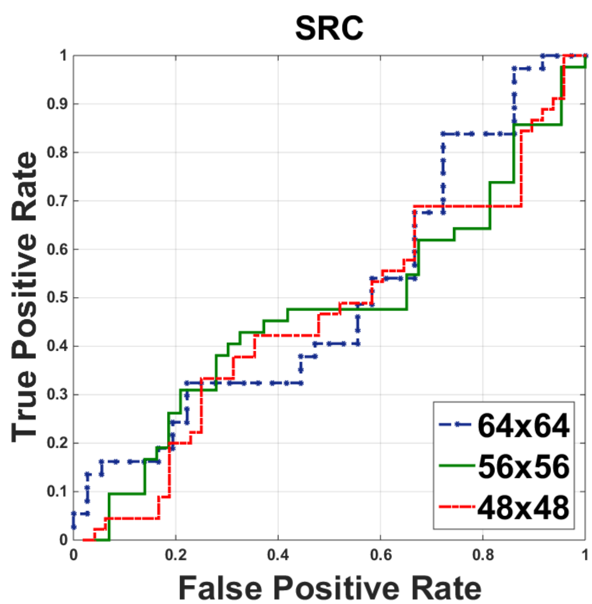

Figure 6.4: ROC curves for breast lesion characterization using conventional SRC classification (LOO CV).

In Table 6.10 and Fig 6.5 we display 10-fold cross-validation results produced by conventional SRC. This method yields 55% ACC and 51.8% AUC, indicating that this approach does not provide separation between the classes.

Table 6.10: Classification performance for breast lesion characterization using conventional SRC on ROIs (10-fold CV).

Size of Block	TPR	TNR	ACC	AUC
64×64	51.35	33.33	42.86	39.23
56×56	53.66	56.41	55	51.84
48×48	58.14	42.55	50.0	50.92

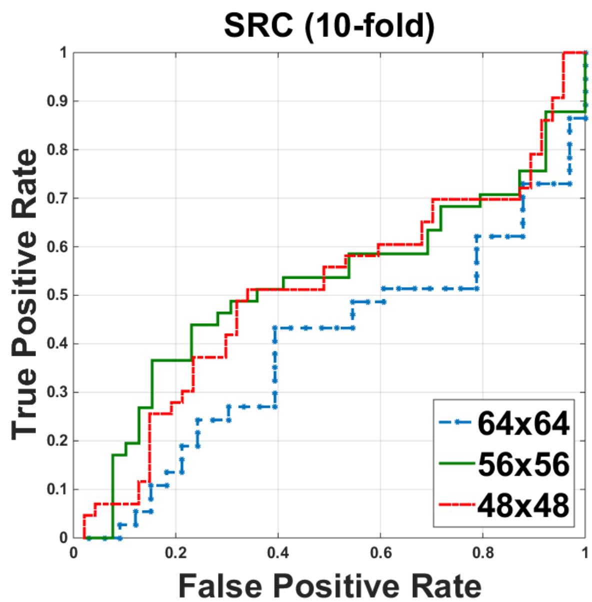

Figure 6.5: ROC curves for breast lesion characterization using conventional SRC classification (10-fold CV).

6.6 Integrative Sparse Classification

In the last part of this experiment we validated our block-based ensemble classification system. In Tables 6.11, 6.12 and 6.13 we present the results with minimum ROI size of 48×48, 56×56 and 64×64 pixels that include 48 benign and 45 malignant lesions, 43 benign and 42 malignant lesions and 36 benign and 37 malignant lesions, respectively using leave-one-out cross-validation. In these tables the rows correspond to classifier ensembles. Overall, the best accuracy achieved by our system using the BBLL approach was 100% for block size of 8×8 and 6×6 using $\tau^*_{LLS} = 0.025$ of 48×48 ROI size, as well as 56×56 with block size 8×8 and using $\tau^*_{LLS} = 0.025$. The ROI size 64×64 with block size 8×8 using $\tau^*_{LLS} = 0.01$ also preformed high accuracy 97.3%. We note that our method improved the accuracy by 56.2%, 34.1% and 37.6% for the corresponding ROI sizes compared to the traditional SRC method. This indicates that the block decomposition and sampling combined with classifier decision fusion yields more accurate solutions than SRC. The ROC graphs in Fig. 6.6 and 6.7 confirm that the BBLL decision function using 8×8 blocks yielded the largest AUC. The BBLL approach contributes to reduction of potential prediction bias. In addition, we applied DeLong tests between the ROC curves produced by BBLL and BBMAP to find whether their differences are statistically significant. For breast lesion characterization, the DeLong's test p-values for minimum ROI size of 64×64 and block sizes of 32×32, 16×16, 8×8 and 4×4 were 0.1, 0.46, 0.39 and 0.009 respectively, suggesting significant differences for block sizes of 4×4. For ROI size 56×56, the p-values for block sizes of 14×14, 8×8 and 4×4 were 0.0078, 0.34 and 0.15 respectively. Applied to ROI size 48×48, the p-values for block size of 16×16, 12×12, 8×8 and 6×6 were 0.46, 0.81, 0.82 and 0.38 respectively. We also note that comparisons between Tables 6.2, 6.3, 6.4 and 6.11, 6.12 and 6.13 indicate that the proposed BBLL ensemble learning approach outperformed the top performing non-sparse texture-based classifier by 36.6% of leave-one-out cross validation. In addition, Fig. 6.8 displays a graph of the classification rates produced by BoK, SRC, BBMAP and BBLL ensemble learners with respect to ROI size (left), and the average ACC for each method over the ROI sizes (right). The summarized ($\mu \pm \sigma$) classification of highest rates over multiple ROI sizes for BoK, SRC, BBMAP and BBLL are 86.0 ± 12.2, 55.0 ± 15.6, 71.6 ± 13.3 and 97.6 ± 3.1 respectively. From these experiments we observe that BBLL and BoK are the top performing approaches, and BBLL yields more consistent classification rates than BoK with respect to the ROI size for leave-one-out cross-validation.

We also performed 10- and 30-fold cross-validation experiments. In Tables 6.14, 6.16 and 6.18 we present results from 10-fold cross-validation. Tables 6.15, 6.17 and 6.19, show

Table 6.11: Classification performance for breast lesion characterization using ensembles of block-based sparse classifiers (ROI size: 48×48, LOO CV)

Size of Block	BBMAP-S				BBLL-S ($\tau_{LLS}=0$)				BBLL-S ($\tau_{LLS}=\tau^*_{LLS}=0.025$)			
	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC
16×16	100	0	48.4	47.6	93.3	2.1	46.2	46.6	88.9	91.7	90.3	89.1
12×12	100	0	48.4	47.6	100	0	48.4	47.6	95.6	100	97.9	97.8
8×8	100	0	48.4	47.6	100	0	48.4	47.6	100	100	100	100
6×6	100	0	48.4	47.6	100	0	48.4	47.6	100	100	100	100
Summary	100	0	48.4	47.6	98.3±3.4	0.5±1.1	47.9±1.1	47.4±0.5	96.1±5.2	97.9±4.2	97.1±4.6	96.1±5.8

Table 6.12: Classification performance for breast lesion characterization using ensembles of block-based sparse classifiers (ROI size: 56×56, LOO CV)

Size of Block	BBMAP-R				BBLL-R ($\tau_{LLS}=0$)				BBLL-R ($\tau_{LLS}=\tau^*_{LLS}=0.025$)			
	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC
14×14	88.1	67.4	77.7	71.8	97.6	55.8	76.5	72.5	73.8	90.7	82.4	81.6
8×8	88.1	55.8	71.8	66.5	97.6	53.5	75.3	70.6	81.0	79.1	80.0	77.2
4×4	71.4	51.2	61.2	60.6	73.8	46.5	60	59.2	69.0	60.5	64.7	65.3

Size of Block	BBMAP-S				BBLL-S ($\tau_{LLS}=0$)				BBLL-S ($\tau_{LLS}=\tau^*_{LLS}=0.006$)			
	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC
14×14	100	74.4	87.1	82.8	90.5	72.1	81.2	81.2	90.5	97.7	94.1	88.4
8×8	100	30.2	64.7	65.2	100	27.9	63.5	63.6	100	100	100	100
4×4	100	0	49.4	46.6	97.6	7.0	51.8	47.3	97.6	100	98.8	97.6
Summary	100	34.9±37.4	67.1±19.0	64.9±18.1	96.0±4.9	35.7±33.2	65.5±14.8	64.0±17.0	96.0±4.9	99.2±1.3	97.6±3.1	97.8±3.5

Table 6.13: Classification performance for breast lesion characterization using ensembles of block-based sparse classifiers (ROI size: 64×64, LOO CV)

Size of Block	BBMAP-S				BBLL-S ($\tau_{LLS}=0$)				BBLL-S ($\tau_{LLS}=\tau^*_{LLS}$=-0.01)			
	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC
32×32	2.7	97.2	49.3	49.6	21.6	83.3	52.1	53.3	24.3	77.8	50.7	49.6
16×16	59.5	100	79.5	77.9	62.2	97.2	79.5	76.6	86.5	97.2	91.8	90.2
8×8	97.3	100	98.6	97.8	89.2	100	94.5	94.8	94.6	100	97.3	94.9
4×4	97.3	100	98.6	97.8	70.3	100	84.9	86	94.6	86.1	90.4	96.8
Summary	60.2±7.1	83.3±22.3	71.6±13.3	70.7±13.7	60.8±28.5	95.1±8.0	77.8±18.2	77.7±17.9	75.7±32.7	90.1±10.3	82.9±20.7	83.2±20.2

TolCon=$1e-6$,TolX=[],TolFun=$1e-8$,MaxIter=6,solver method=2(LP) (4th table);

TolCon=$1e-6$,TolX=$1e-6$,TolFun=$1e-6$,MaxIter=10,solver method=4(SOCP) (5th table);

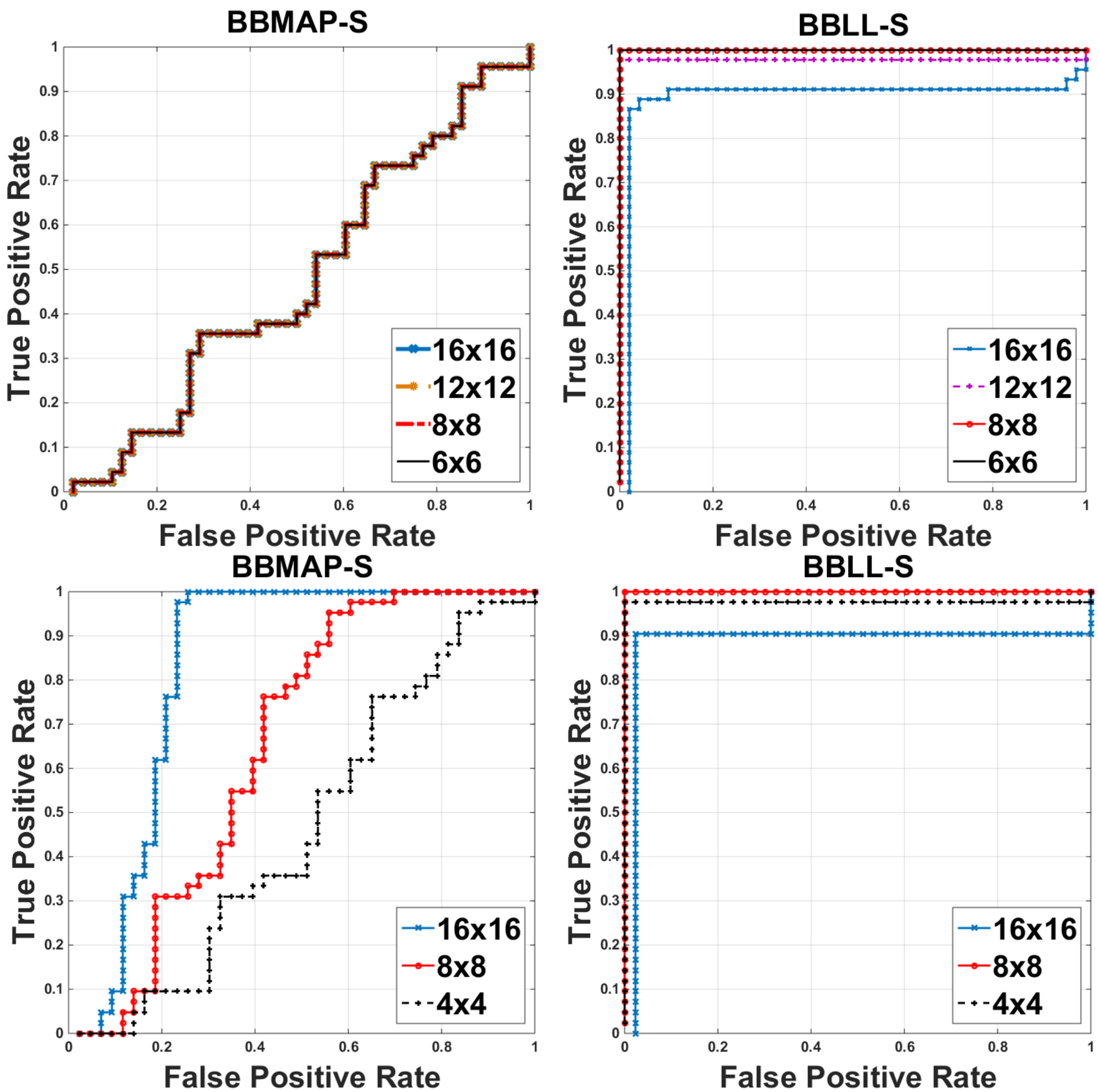

Figure 6.6: ROC curves for 48×48 (top row), 56×56 (bottom row) ROI size breast lesion characterization using the proposed block-based ensemble method with BBMAP (left), and BBLL (right) decision functions with leave-one-out cross-validation.

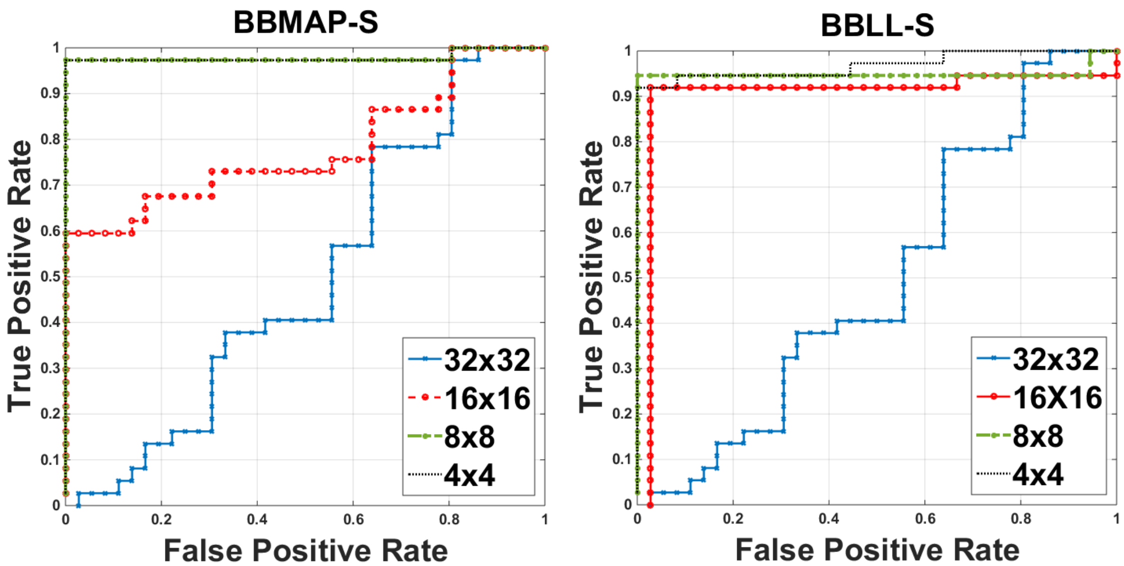

Figure 6.7: ROC curves for 64×64 ROI size breast lesion characterization using the proposed block-based ensemble method with BBMAP (left), and BBLL (right) decision functions with leave-one-out cross-validation.

results obtained using 30-fold cross-validation. Furthermore, Fig. 6.6, 6.6,and 6.6 display the ROC graphs for 10- and 30-fold cross-validation. We note that the accuracy increases when the number of folds increases for the same ROI size. Also, ACC increases for the same number of folds cross-validation when the ROI size increases. The highest accuracy by using 10-fold cross-validation is 68.89% and corresponding area under the curve is 73.73% for 48×48 ROI size with 8×8 block size. For 20-fold cross-validation, the best accuracy is 75% and AUC is 74.42% for 64×64 ROI size with 8×8 block size. The best performance over all the ROI size experiments for k-fold cross-validation is obtained for 30-fold cross validation. The highest accuracy is 86.67% and corresponding area under the curve is 88.21% for 64×64 ROI size with 16×16 block size. There are 2 or 3 testing samples for ROI size 64×64 when $k = 30$. We estimated the AUC values of BBLL with optimized threshold τ^*_{LLS} and BBMAP by applying DeLong's statistical test between the ROCs produced by BBMAP and BBLL as well for k-fold cross-validation. The p-values for ROI size 64×64 with block sizes of 32×32, 16×16, 8×8 and 4×4 were 0.78, 0.49, 0.24 and 0.21 respectively for 10-fold cross-validation,

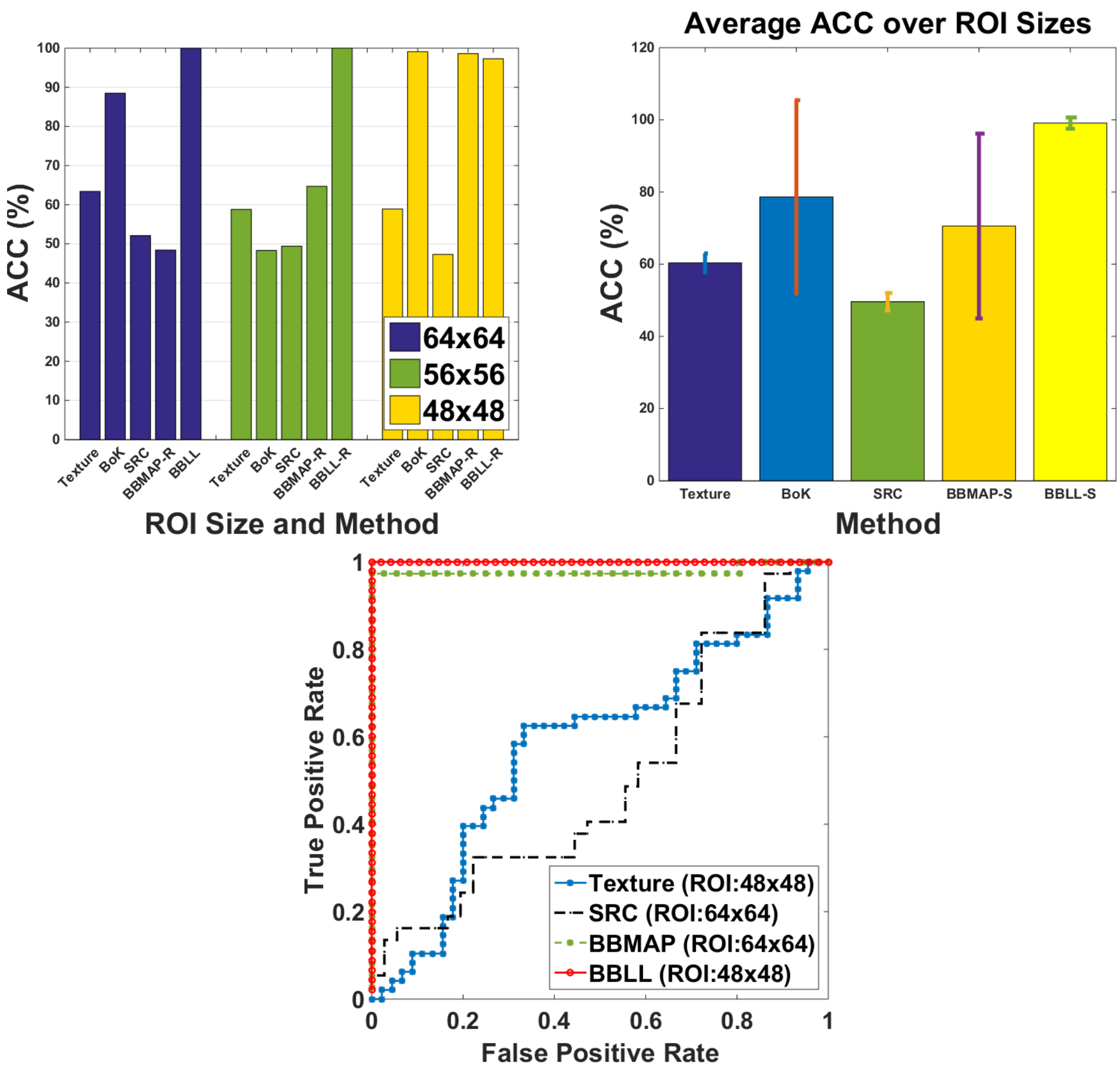

Figure 6.8: Graphs of ACC values versus ROI size produced by BoK, SRC, BBMAP and BBLL (left) and the corresponding average ACC for each method over all ROI sizes (right), the corresponding AUC of the best ACC from each method (bottom) using leave-one-out cross-validation.

Table 6.14: Classification performance for breast lesion characterization using ensembles of block-based sparse classifiers (ROI size: 48×48), 10-fold CV.

Size of Block	TPR	TNR	ACC	AUC
48×48(SOCP-S)	58.14	42.55	50.0	50.92

Size of Block	BBMAP-S				BBLL-S ($\tau_{LLS} = 0$)				BBLL-S ($\tau_{LLS} = \tau^*_{LLS} = 0.02$)			
	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC
24×24	76.74	31.91	53.33	56.61	76.74	31.91	53.33	56.56	60.47	70.21	65.56	60.42
16×16	83.72	38.30	60.0	62.1	79.07	44.68	61.11	63.63	60.47	74.47	67.78	69.47
12×12	79.07	51.06	64.44	67.19	79.07	53.19	65.56	68.78	62.79	74.47	68.89	73.73
8×8	67.44	63.83	65.56	69.52	69.77	63.83	66.67	70.51	58.14	76.6	67.78	72.09
6×6	67.44	65.96	66.67	70.95	72.09	63.83	67.78	71.40	58.14	76.6	67.78	70.86
4×4	65.12	65.96	65.56	69.22	65.12	65.96	65.56	69.22	55.81	76.6	66.67	69.57

Size of Block	TPR	TNR	ACC	AUC
48×48(SOCP-S)	60.47	48.94	54.44	54.68

Size of Block	BBMAP-S				BBLL-S ($\tau_{LLS} = 0$)			
	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC
24×24	9.3	97.87	55.56	52.25	13.95	95.74	56.67	53.19
16×16	2.33	100	53.33	49.98	2.33	97.87	52.22	49.28
12×12	2.33	100	53.33	50.72	0	100	52.22	48.69
8×8	46.51	76.60	62.22	61.75	6.98	93.62	52.22	47.9
6×6	62.79	68.09	65.56	68.33	27.91	85.11	57.78	51.56
4×4	9.30	97.87	55.56	53.29	0	100	52.22	48.69

MaxIter = 10 (top two tables) and MaxIter = 20 (bottom two tables)

and p-values for 30-fold cross-validation were 0.72, 0.54, 0.16 and 0.0086 respectively. The p-values for ROI size 56×56 with block sizes of 14×14, 8×8 and 4×4 were 0.96, 0.33 and 0.61 respectively for 10-fold cross-validation, and p-values for 30-fold cross-validation were 0.72, 0.12, 0.00044 and 0.00064 respectively. The p-values for ROI size 48×48 with block sizes of 16×16, 12×12, 8×8 and 6×6 were 0.61, 0.26, 0.33 and 0.66 respectively for 10-fold cross-validation, and p-values for 30-fold cross-validation were 0.12, 0.00048, 0.00044 and 0.0052 respectively.

We also measured the standardized execution times of our BBLL method versus the ROI size and the block size. For each method we applied cross-validation experiments and we divided the total execution time by the number of experiments and the number

Table 6.15: Classification performance for breast lesion characterization using ensembles of block-based sparse classifiers (ROI size: 48×48), 30-fold CV.

Size of Block	TPR	TNR	ACC	AUC
48×48 (SOCP-S)	46.51	40.43	43.33	42.8

Size of Block	BBMAP-S				BBLL-S ($\tau_{LLS} = 0$)				BBLL-S ($\tau_{LLS} = \tau^*_{LLS} = 0.03$)			
	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC
24×24	81.4	23.4	51.11	54.38	79.07	12.77	44.44	47.11	67.44	68.09	67.78	63.29
16×16	100	21.28	58.89	60.17	95.35	25.53	58.89	60.86	72.09	74.47	73.33	77.04
12×12	100	31.91	64.44	66.25	97.67	31.91	63.33	65.26	74.42	78.72	76.67	85.6
8×8	100	36.17	66.67	68.09	100	31.91	64.44	65.41	79.07	76.60	77.78	85.01
6×6	100	29.79	63.33	63.58	100	29.79	63.33	63.58	79.07	76.60	77.78	84.66
4×4	100	29.79	63.33	63.58	100	29.79	63.33	63.58	79.07	76.60	77.78	85.16

Size of Block	TPR	TNR	ACC	AUC
48×48 (SOCP-S)	58.14	42.55	50.0	49.48

Size of Block	BBMAP-S				BBLL-S ($\tau_{LLS} = 0$)			
	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC
24×24	6.98	95.74	53.33	50.47	6.98	93.62	52.22	48.39
16×16	0	100	52.22	48.69	0	97.87	51.11	47.5
12×12	11.63	100	57.78	55.71	0	100	52.22	48.69
8×8	58.14	61.70	60.0	58.98	13.95	97.87	57.78	53.09
6×6	86.05	42.55	63.33	64.42	32.56	82.98	58.89	54.53
4×4	2.33	100	53.33	50.22	0	100	52.22	48.69

MaxIter = 10 (top two tables) and MaxIter = 20 (bottom two tables)

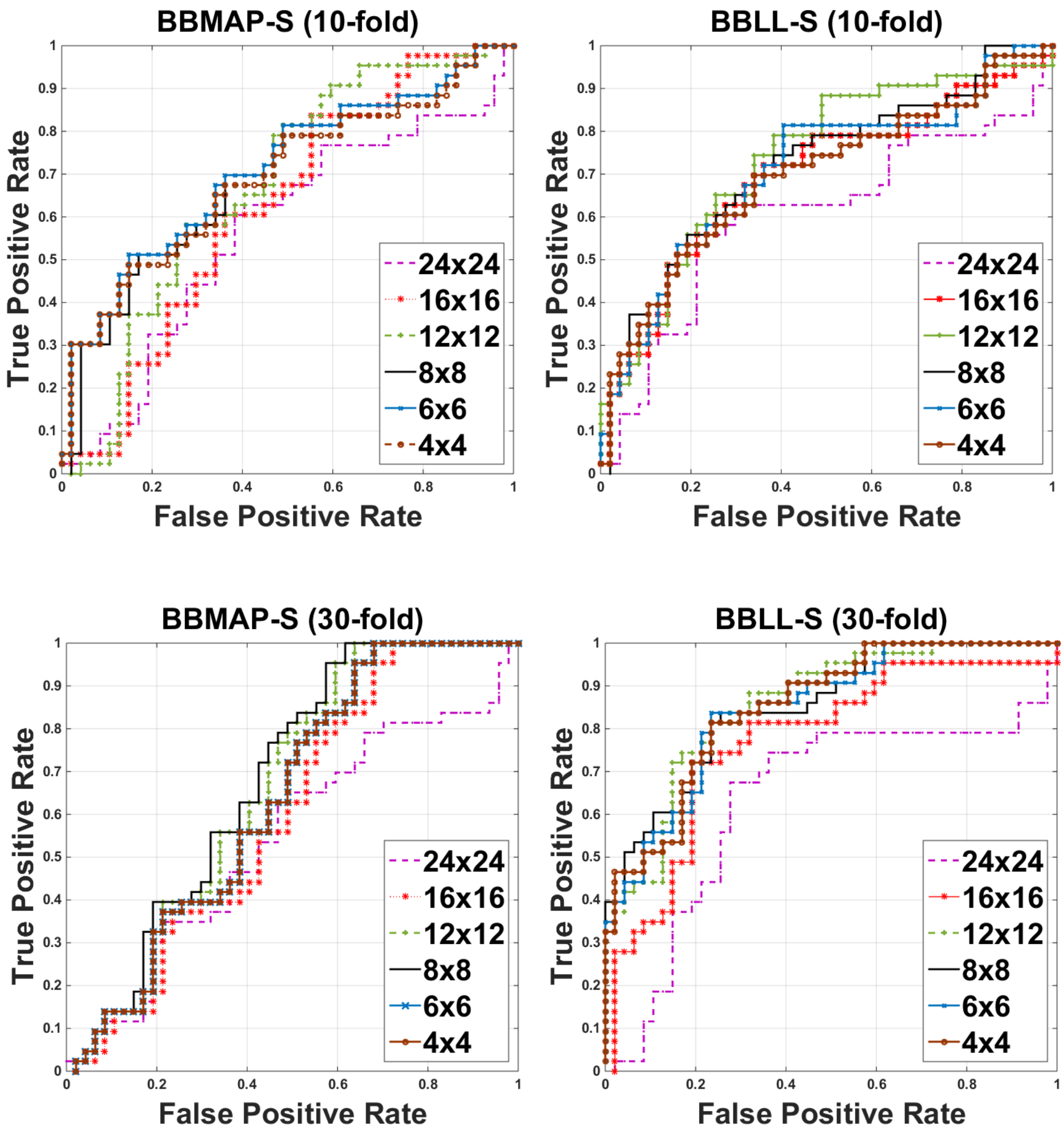

Figure 6.9: ROC curves for 48×48 ROI size breast lesion characterization using the proposed block-based ensemble method with BBMAP-S (left), and BBLL-S (right) decision functions with 10- (top row) and 30-fold (bottom row) cross-validation.

Table 6.16: Classification performance for breast lesion characterization using ensembles of block-based sparse classifiers (ROI size: 56×56, 10-fold CV).

Size of Block	TPR	TNR	ACC	AUC
56×56(SOCP-S)	53.66	56.41	55	51.84

Size of Block	BBMAP-S				BBLL-S ($\tau_{LLS} = 0$)				BBLL-S ($\tau_{LLS} = \tau^*_{LLS} = 0.01$)			
	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC
28×28	65.85	56.41	61.25	60.73	65.85	46.15	56.25	54.16	60.98	58.97	60.0	58.72
14×14	90.24	30.77	61.25	63.35	82.93	33.33	58.75	59.35	73.17	48.72	61.25	61.41
8×8	90.24	30.77	61.25	63.35	90.24	33.33	62.5	65.35	63.41	64.10	63.75	67.85

Table 6.17: Classification performance for breast lesion characterization using ensembles of block-based sparse classifiers (ROI size: 56×56), 30-fold CV.

Size of Block	TPR	TNR	ACC	AUC
56×56(SOCP-S)	56.25	57.14	56.67	47.99

Size of Block	BBMAP-S				BBLL-S ($\tau_{LLS} = 0$)				BBLL-S ($\tau_{LLS} = \tau^*_{LLS} = 0.02$)			
	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC
28×28	62.5	78.57	70	62.72	71.88	60.71	66.67	58.59	71.88	60.71	66.67	60.71 (0)
14×14	100	64.29	83.33	77.46	87.5	64.29	76.67	70.2	62.5	96.43	78.33	82.59
8×8	100	64.29	83.33	77.46	100	64.29	83.33	77.46	68.75	100	83.33	94.87

MaxIter = 10

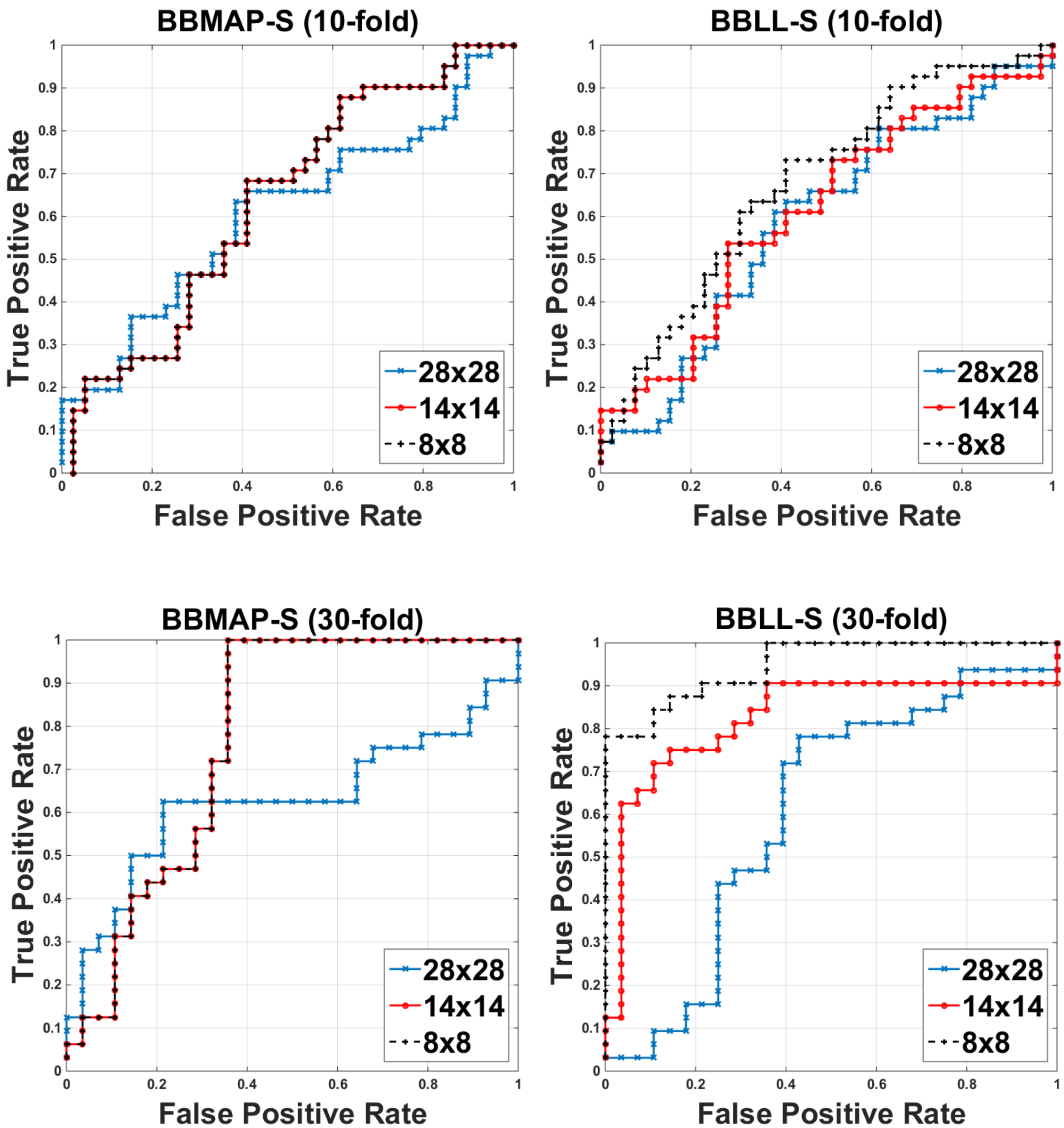

Figure 6.10: ROC curves for 56×56 ROI size breast lesion characterization using the proposed block-based ensemble method with BBMAP-S (left), and BBLL-S (right) decision functions with 10- (top row) and 30-fold (bottom row) cross-validation.

Table 6.18: Classification performance for breast lesion characterization using ensembles of block-based sparse classifiers (ROI size: 64×64, 10-fold CV.)

Size of Block	TPR	TNR	ACC	AUC
64×64(SOCP-S)	51.35	33.33	42.86	39.23

Size of Block	BBMAP-S				BBLL-S ($\tau_{LLS}=0$)				BBLL-S ($\tau_{LLS}=\tau^*_{LLS}$=-0.02)			
	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC
32×32	40.54	90.91	64.29	64.29	32.43	75.76	52.86	51.68	45.95	69.70	57.14	60.44
16×16	59.46	81.82	70.0	69.69	54.05	78.79	65.71	65.93	54.05	78.79	65.71	70.84
8×8	48.65	81.82	64.29	63.64	40.54	87.88	62.86	62.49	67.57	66.67	67.14	71.42

Table 6.19: Classification performance for breast lesion characterization using ensembles of block-based sparse classifiers (ROI size: 64×64), 30-fold CV.

Size of Block	TPR	TNR	ACC	AUC
72×72(sparsity)	20.59	76.92	45.0	40.05

Size of Block	BBMAP-S				BBLL-S ($\tau_{LLS}=0$)				BBLL-S ($\tau_{LLS}=\tau^*_{LLS}$=-0.02)			
	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC	TPR	TNR	ACC	AUC
32×32	9.68	100	53.33	48.83	29.03	82.76	55	48.39	75.86	61.67	68.33	65.41
16×16	70.97	100	85	85.65	64.52	96.55	80	80.98	93.55	79.31	86.67	88.21
8×8	74.19	93.1	83.33	82.09	70.97	93.10	81.67	80.65	93.55	72.41	83.33	89.1

MaxIter=10

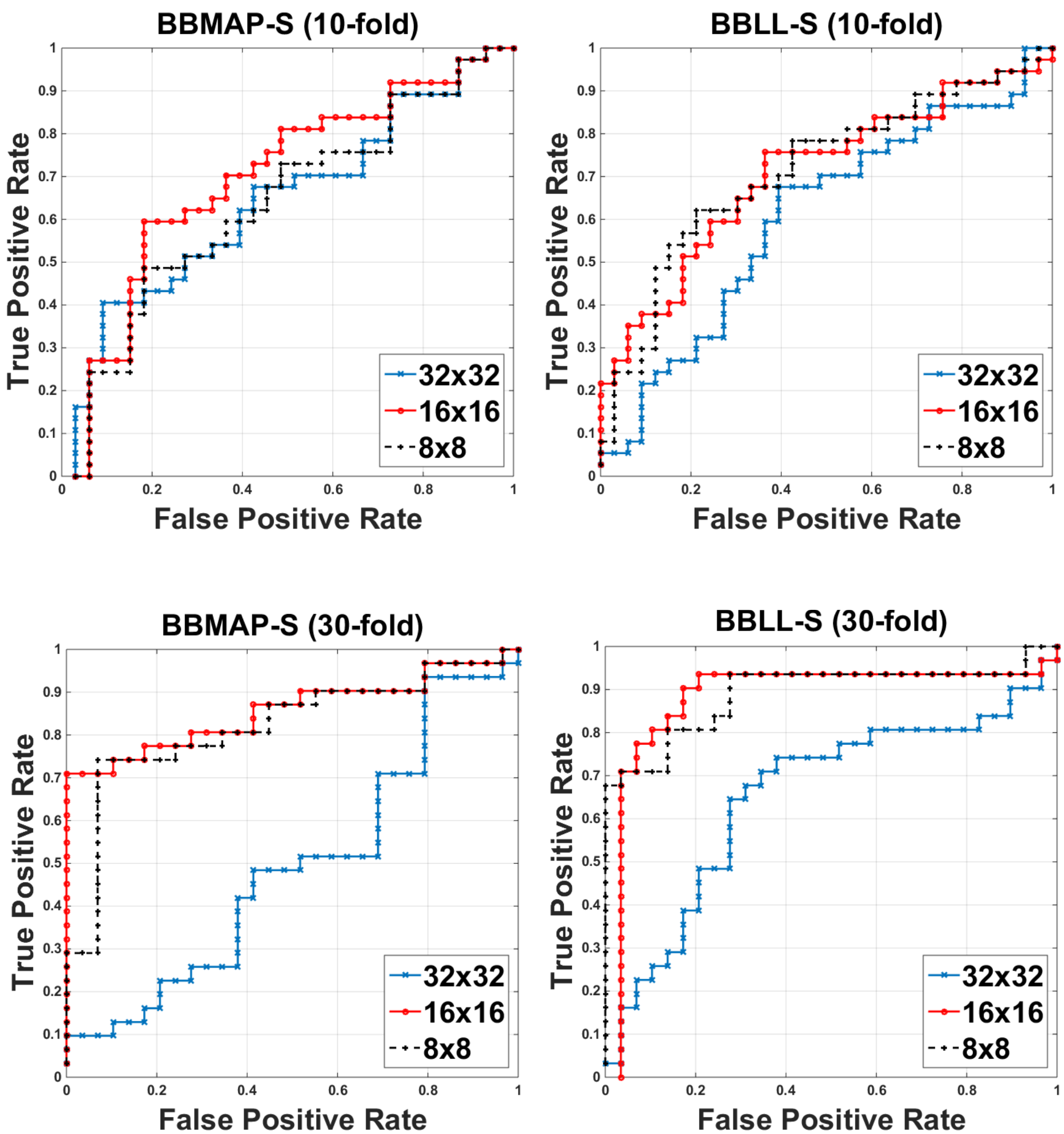

Figure 6.11: ROC curves for 64 × 64 ROI size breast lesion characterization using the proposed block-based ensemble method with BBMAP-S (left), and BBLL-S (right) decision functions with 10- (top row) and 30-fold (bottom row) cross-validation.

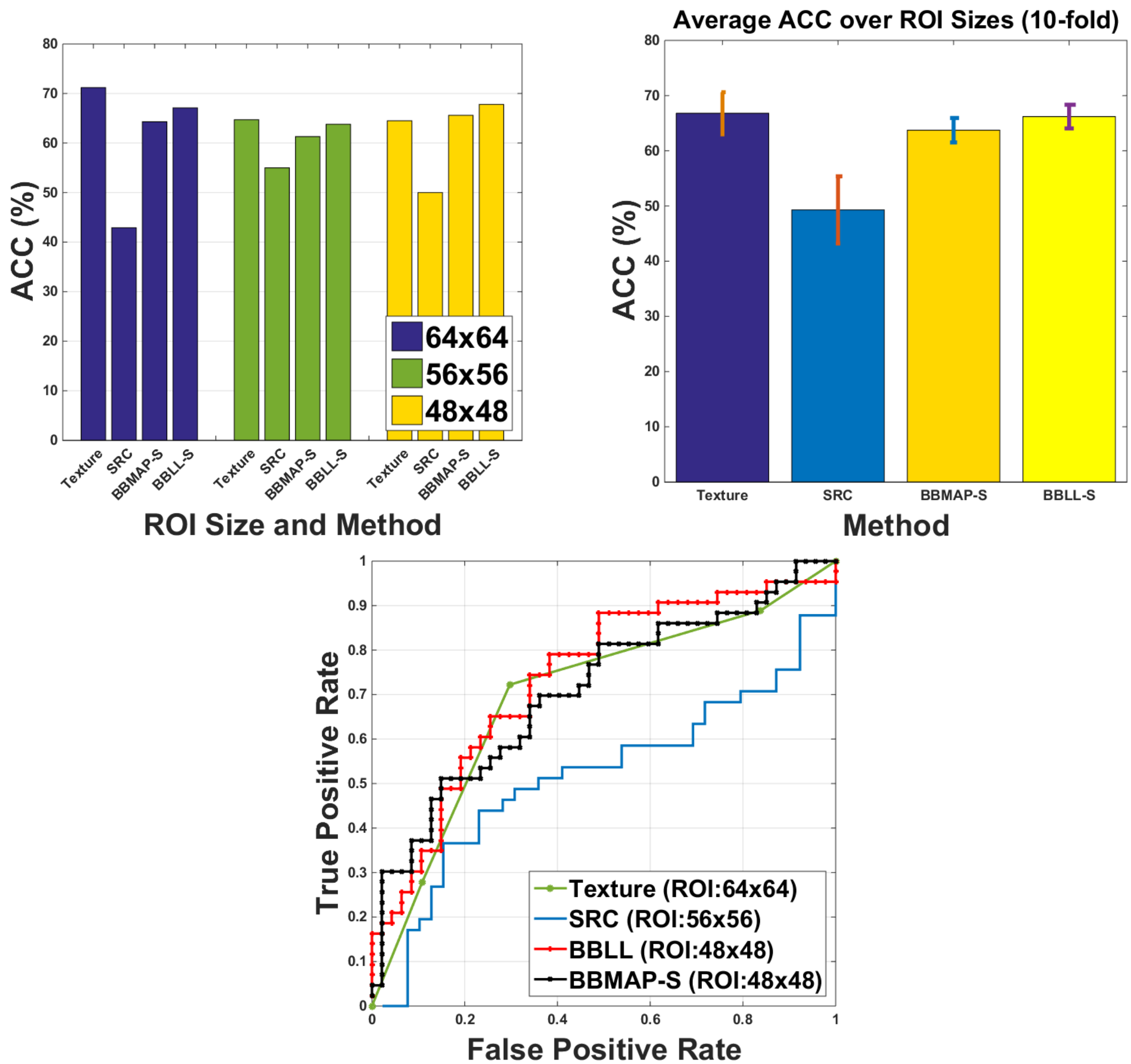

Figure 6.12: Graphs of ACC values versus ROI size produced by BoK, SRC, BBMAP and BBLL (left) and the corresponding average ACC for each method over all ROI sizes (right), and the corresponding AUC of the best ACC from each method (bottom) using 10-fold CV.

of subjects. Then we identified all values by the maximum execution time. Overall the average standardized execution time of conventional SRC for the MIAS dataset using ROI sizes 64×64, 56×56, 48×48 were all approximately equal to 0.015 indicating that the execution times of conventional SRC are not dependent on the ROI size of the lesion. When we applied to 64×64 ROIs classifier ensembles with block sizes of 32×32, 16×16, 8×8, we measured execution times of 0.038, 0.128, 0.473 respectively. These results suggest that the computational time for BBLL increases linearly with the number of blocks. We also calculated the execution times for the BoK method and ROI sizes of 64×64, 56×56, 48×48. The standardized execution times were all approximately equal to 0.422. BoK applies the keypoint-feature extraction stage, therefore the execution time depends mostly on the number of keypoints and not very much on the ROI size. We observe that the top performing BBLL method for 64×64 ROI size and 8×8 block size requires about the same execution time as the top performing BoK for 48×48 ROI size.

Chapter VII: CONCLUSION

This dissertation studies the use of sparsity and integrative classification techniques for characterization of biomedical imaging patterns and separation of healthy from diseased subjects. Sparse analysis provides an elegant and theoretically sound foundation for representation and recognition of patterns. Sparsity-based techniques have been successfully applied to image reconstruction, signal processing, denoising and classification problems.

We first implemented and tested texture-based descriptors and classification techniques to evaluate the difficulty of separation and use these techniques as benchmarks for performance evaluation of sparse analysis approaches. Despite the fact that there were little to no visual differences between the two classes, the top performing techniques yielded 67.8% ACC and 70.9% area-under-the-curve of ROC for bone characterization and 71.2% ACC and 69.8% AUC for breast lesion characterization. These results support the hypothesis that 2D texture analysis can contribute to identification of changes in trabecular bone microarchitecture.

In the next stage we proposed integrative block-based sparse classification techniques for automated lesion characterization. We introduced two Bayesian decision functions based on maximum a posteriori (MAP) and log likelihood (LL) estimates. We compared our ensemble of sparse classifiers to conventional SRC, texture-based, and Bag of Keypoints approaches.

We applied our method to diagnosis of osteoporosis in digital radiographs and breast lesion characterization in mammograms. The integrative sparse-based method (BBLL-S) produced classification rates of 100% for bone characterization and as well as 100% for breast lesion characterization with leave-one-out cross-validation. For 30-fold cross-validation, BBLL-S yielded 70.7% ACC and 74.4% AUC for bone characterization and 86.7% ACC and 88.2% AUC for breast lesion characterization. For 10-fold cross-validation, BBLL-S produced 60.6% ACC and 62.5% AUC for bone characterization and 68.9% ACC and 73.7% AUC for breast lesion characterization. Our results indicate that the introduction of patch analysis yields more accurate solutions than the compared methods. Our block-based approach produced better performance than SRC and texture-based classifiers. The performance of Bag of Key-points was very high, albeit slightly less consistent than BBLL with respect to ROI size in breast lesion characterization and slightly lower for bone characterization. Also, the BoK method may be slower than the BBLL learners especially if the block sizes are relatively large in BBLL. Our results also indicate that BBLL produces more accurate classification than BBMAP. Another advantage of this method is that it calculates the types of features to be used without being dependent on the imaging modality or the disease pattern. Therefore it is expected to be applicable to various clinical applications for identification of subjects with higher risk of disease and computer-aided diagnosis.

REFERENCE LIST

[1] S Agatonovic-Kustrin and R Beresford. Basic concepts of artificial neural network (ann) modeling and its application in pharmaceutical research. *Journal of pharmaceutical and biomedical analysis*, 22(5):717–727, 2000.

[2] Michal Aharon and Michael Elad. Sparse and redundant modeling of image content using an image-signature-dictionary. *SIAM Journal on Imaging Sciences*, 1(3):228–247, 2008.

[3] Michal. Aharon, Michael. Elad, and Alfred. Bruckstein. K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, 54(11):4311–4322, November 2006.

[4] Timo Ahonen, Abdenour Hadid, and Matti Pietikainen. Face description with local binary patterns: Application to face recognition. *IEEE transactions on pattern analysis and machine intelligence*, 28(12):2037–2041, 2006.

[5] F. Alizadeh and D. Goldfarb. Second-order cone programming. *MATHEMATICAL PROGRAMMING*, 95:3–51, 2001.

[6] Edoardo Amaldi and Viggo Kann. On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems. *Theoretical Computer Science*, 209(1):237 – 260, 1998.

[7] Reiner Bartl and Bertha Frisch. *Osteoporosis: diagnosis, prevention, therapy*. Springer Science & Business Media, 2009.

[8] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006.

[9] S Blanco, A Figliola, R Quian Quiroga, OA Rosso, and E Serrano. Time-frequency analysis of electroencephalogram series. iii. wavelet packets and information cost function. *Physical Review E*, 57(1):932, 1998.

[10] Peter Boyle, Bernard Levin, et al. *World cancer report 2008*. IARC Press, International Agency for Research on Cancer, 2008.

[11] Stan Brown. *Measures of Shape: Skewness and Kurtosis*, 20082017. https://brownmath.com/stat/shape.htm.

[12] Richard H Byrd, Jean Charles Gilbert, and Jorge Nocedal. A trust region method based on interior point techniques for nonlinear programming. *Mathematical Programming*, 89(1):149–185, 2000.

[13] Richard H Byrd, Mary E Hribar, and Jorge Nocedal. An interior point algorithm for large-scale nonlinear programming. *SIAM Journal on Optimization*, 9(4):877–900, 1999.

[14] Deng Cai, Xiaofei He, and Jiawei Han. Spectral regression: A unified approach for sparse subspace learning. In *Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on*, pages 73–82. IEEE, 2007.

[15] Emmanuel J. Cands, Justin K. Romberg, and Terence Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics*, 59(8):1207–1223, 2006.

[16] Emmanuel J. Cands and Terence Tao. Near-optimal signal recovery from random projections: Universal encoding strategies? *IEEE Transactions on Information Theory*, 52(12):5406–5425, Dec 2006.

[17] Olivier Chapelle, Bernhard Scholkopf, and Alexander Zien. Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]. *IEEE Transactions on Neural Networks*, 20(3):542–542, 2009.

[18] Scott Shaobing Chen, David L. Donoho, and Michael A. Saunders. Atomic decomposition by basis pursuit. *SIAM Rev.*, 43(1):129–159, January 2001.

[19] A. F. Costa, G. Humpire-Mamani, and A. J. M. Traina. An efficient algorithm for fractal analysis of textures. In *2012 25th SIBGRAPI Conference on Graphics, Patterns and Images*, pages 39–46, Aug 2012.

[20] Gabriella Csurka, Christopher R. Dance, Lixin Fan, Jutta Willamowski, and Cdric Bray. Visual categorization with bags of keypoints. In *In Workshop on Statistical Learning in Computer Vision, ECCV*, pages 1–22, 2004.

[21] Christos Davatzikos, Dinggang Shen, Ruben C Gur, Xiaoying Wu, Dengfeng Liu, Yong Fan, Paul Hughett, Bruce I Turetsky, and Raquel E Gur. Whole-brain morphometric study of schizophrenia revealing a spatially complex set of focal abnormalities. *Archives of general psychiatry*, 62(11):1218–1227, 2005.

[22] G. Davis, S. Mallat, and M. Avellaneda. Adaptive greedy approximations. *Constructive Approximation*, 13(1):57–98, Mar 1997.

[23] Thomas G Dietterich and Eun Bae Kong. Machine learning bias, statistical bias, and statistical variance of decision tree algorithms. Technical report, Technical report, Department of Computer Science, Oregon State University, 1995.

[24] David L. Donoho. For most large underdetermined systems of linear equations the minimal 1-norm solution is also the sparsest solution. *Comm. Pure Appl. Math*, 59(6):797–829, 2004.

[25] David L. Donoho and Michael Elad. Optimally sparse representation in general (nonorthogonal) dictionaries via 1 minimization. *Proceedings of the National Academy of Sciences*, 100(5):2197–2202, 2003.

[26] Richard O. Duda, Peter E. Hart, David G. Stork, C R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification, 2nd Ed.* Wiley-Interscience, 2001.

[27] M.C. Erlandson, A.L. Lorbergs, S. Mathur, and A.M. Cheung. Muscle analysis using pqct, dxa and mri. *European Journal of Radiology*, 85(8):1505 – 1511, 2016.

[28] FDA. Guidelines for preclinical and clinical evaluation of agents used in the prevention or treatment of postmenopausal osteoporosis. *Division of Metabolic and Endocrine Drug Products, Food and Drug Administration; Rockville, MD*, 17(4):S125–S133, Apr, 1994.

[29] Jacques Ferlay, Clarisse Héry, Philippe Autier, and Rengaswamy Sankaranarayanan. *Global Burden of Breast Cancer*, pages 1–19. Springer New York, New York, NY, 2010.

[30] M. A. T. Figueiredo, R. Nowak, and S. Wright. Gradient projection for sparse reconstruction: application to compressed sensing and other inverse problems. *IEEE Journal on Selected Topics in Signal Processing*, 1(4):586–597, December 2007.

[31] Roger Fletcher. *Practical methods of optimization*. John Wiley & Sons, 2013.

[32] Committee for Proprietary Medicinal Products. Note for guidance on postmenopausal osteoporosis in women. *European Agency for the Evaluation of Medicinal Products, London*, 2001. (CPMP/EWP/552/95 rev. 1.

[33] Jerome H Friedman. On bias, variance, 0/1loss, and the curse-of-dimensionality. *Data mining and knowledge discovery*, 1(1):55–77, 1997.

[34] Jerome H Friedman and Lawrence C Rafsky. Multivariate generalizations of the wald-wolfowitz and smirnov two-sample tests. *The Annals of Statistics*, pages 697–717, 1979.

[35] Alison L Gibbs and Francis Edward Su. On choosing and bounding probability metrics. *International statistical review*, 70(3):419–435, 2002.

[36] Philip E Gill, Walter Murray, Michael A Saunders, John A Tomlin, and Margaret H Wright. On projected newton barrier methods for linear programming and an equivalence to karmarkars projective method. *Mathematical programming*, 36(2):183–209, 1986.

[37] Robert M Haralick. Statistical and structural approaches to texture. *Proceedings of the IEEE*, 67(5):786–804, 1979.

[38] Khaled Harrar, Rachid Jennane, Karima Zaouchi, Thomas Janvier, Hechmi Toumi, and Eric Lespessailles. Oriented fractal analysis for improved bone microarchitecture characterization. *Biomedical Signal Processing and Control*, 39:474–485, 2018.

[39] Mohammed El Hassouni, Abdessamad Tafraouti, Hechmi Toumi, Eric Lespessailles, and Rachid Jennane. Fractional brownian motion and rao geodesic distance for bone x-ray image characterization. *IEEE J. Biomedical and Health Informatics*, 21(5):1347–1359, 2017.

[40] M. Heath, K. Bowyer, Daniel B. Kopans, P. Kegelmeyer Jr., Richard H. Moore, K. Chang, and S. Munishkumaran. Current status of the digital database for screening mammography. In Nico Karssemeijer, Martin Thijssen, Jan H. C. L. Hendriks, and Leon van Erning, editors, *Digital Mammography / IWDM*, volume 13 of *Computational Imaging and Vision*, pages 457–460. Springer, 1998.

[41] S. Hough. Fast and slow bone losers. relevance to the management of osteoporosis. *Drugs and aging*, 12(Suppl. 1):1–7, 1998.

[42] Guang-Bin Huang, Hongming Zhou, Xiaojian Ding, and Rui Zhang. Extreme learning machine for regression and multiclass classification. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 42(2):513–529, 2012.

[43] A. K. Jain, R. P. W. Duin, and Jianchang Mao. Statistical pattern recognition: a review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1):4–37, Jan 2000.

[44] R. Jennane, W. J. Ohley, S. Majumdar, and G. Lemineur. Fractal analysis of bone x-ray tomographic microscopy projections. *IEEE Transactions on Medical Imaging*, 20(5):443–449, May 2001.

[45] Rachid Jennane, Rachid Harba, Grald Lemineur, Stphanie Bretteil, Anne Estrade, and Claude Laurent Benhamou. Estimation of the 3d self-similarity parameter of trabecular bone from its 2d projection. *Medical Image Analysis*, 11(1):91 – 98, 2007.

[46] John A Kanis, L Joseph Melton, Claus Christiansen, Conrad C Johnston, and Nikolai Khaltaev. The diagnosis of osteoporosis. *Journal of bone and mineral research*, 9(8):1137–1141, 1994.

[47] Sotiris B Kotsiantis, I Zaharakis, and P Pintelas. Supervised machine learning: A review of classification techniques. *Emerging artificial intelligence applications in computer engineering*, 160:3–24, 2007.

[48] Pradnya Kulkarni, Andrew Stranieri, Siddhivinayak Kulkarni, Julien Ugon, and Manish Mittal. Hybrid technique based on ngram and neural networks for classification of mammographic images. In *Second International Conference on Signal, Image Processing and Pattern Recognition*, pages 297–306, 2014.

[49] Deena Lala, Angela M. Cheung, Cheryl L. Lynch, Dean Inglis, Chris Gordon, George Tomlinson, and Lora Giangregorio. Measuring apparent trabecular structure with pqct: a comparison with hr-pqct. *J Clin Densitom*, 17(1):47–53, 2014.

[50] San-Kan Lee, Chien-Shun Lo, Chuin-Mu Wang, Pau-Choo Chung, Chein-I Chang, Ching-Wen Yang, and Pi-Chang Hsu. A computer-aided design mammography screening system for detection and classification of microcalcifications. *International journal of medical informatics*, 60 1:29–57, 2000.

[51] Rafael Llobet, Roberto Paredes, and Juan C Pérez-Cortés. Comparison of feature extraction methods for breast cancer detection. In *Iberian Conference on Pattern Recognition and Image Analysis*, pages 495–502. Springer, 2005.

[52] Miguel Sousa Lobo, Lieyen Vandenberghe, Stephen Boyd, and Herv Lebret. Applications of second-order cone programming. *Linear Algebra and its Applications*, 284:193–228, 1998.

[53] Norma J Macintyre and Amanda L Lorbergs. Imaging-based methods for non-invasive assessment of bone properties influenced by mechanical loading. *Physiotherapy Canada*, 64(2):202215, 2012.

[54] J. Mairal, M. Elad, and G. Sapiro. Sparse representation for color image restoration. *Trans. Img. Proc.*, 17(1):53–69, January 2008.

[55] S. Makrogiannis, Ragini Verma, and Christos Davatzikos. Anatomical equivalence class: A morphological analysis framework using a lossless shape descriptor. *IEEE Trans. Med. Imaging*, 26(4):619–631, 2007.

[56] S.G. Mallat and Zhifeng Zhang. Matching pursuits with time-frequency dictionaries. *Trans. Sig. Proc.*, 41(12):3397–3415, December 1993.

[57] E Martın-Badosa, A Elmoutaouakkil, S Nuzzo, D Amblard, L Vico, and F Peyrin. A method for the automatic characterization of bone architecture in 3d mice microtomographic images. *Computerized Medical Imaging and Graphics*, 27(6):447–458, 2003.

[58] Bruno Roberto Nepomuceno Matheus and Homero Schiabel. Online mammographic images database for development and comparison of cad schemes. *Journal of digital imaging*, 24(3):500–506, 2011.

[59] Anke Meyer-Baese. *Pattern recognition for medical imaging.* Academic Press, 2004.

[60] Subhasis Misra, Naveenraj L Solomon, Frederick L Moffat, and Leonidas G Koniaris. Screening criteria for breast cancer. *Advances in surgery*, 44(1):87–100, 2010.

[61] Tom M Mitchell. *The need for biases in learning generalizations.* Department of Computer Science, Laboratory for Computer Science Research, Rutgers Univ. New Jersey, 1980.

[62] Tom M. Mitchell. *Machine learning*. McGraw Hill series in computer science. McGraw-Hill, 1997.

[63] Inês C Moreira, Igor Amaral, Inês Domingues, António Cardoso, Maria João Cardoso, and Jaime S Cardoso. Inbreast: toward a full-field digital mammographic database. *Academic radiology*, 19(2):236–248, 2012.

[64] Radhakrishnan Nagarajan and Meenakshi Upreti. An ensemble predictive modeling framework for breast cancer classification. *Methods*, 131:128 – 134, 2017. Systems Approaches for Identifying Disease Genes and Drug Targets.

[65] Ilia Nouretdinov, Sergi G Costafreda, Alexander Gammerman, Alexey Chervonenkis, Vladimir Vovk, Vladimir Vapnik, and Cynthia HY Fu. Machine learning classification with confidence: application of transductive conformal predictors to mri-based diagnostic and prognostic markers in depression. *Neuroimage*, 56(2):809–813, 2011.

[66] Arnau Oliver, Jordi Freixenet, Joan Marti, Elsa Perez, Josep Pont, Erika RE Denton, and Reyer Zwiggelaar. A review of automatic mass detection and segmentation in mammographic images. *Medical image analysis*, 14(2):87–110, 2010.

[67] Arnau Oliver, Xavier Lladó, Elsa Pérez, Josep Pont, Erika R. E. Denton, Jordi Freixenet, and Joan Martí. A statistical approach for breast density segmentation. *Journal of Digital Imaging*, 23(5):527–537, Oct 2010.

[68] Hind Oulhaj, Mohammed Rziza, Aouatif Amine, Hechmi Toumi, Eric Lespessailles, Mohammed El Hassouni, and Rachid Jennane. Anisotropic discrete dual-tree wavelet transform for improved classification of trabecular bone. *IEEE Trans. Med. Imaging*, 36(10):2077–2086, 2017.

[69] Hind Oulhaj, Mohammed Rziza, Aouatif Amine, Hechmi Toumi, Eric Lespessailles, Rachid Jennane, and Mohammed El Hassouni. Trabecular bone characterization using circular parametric models. *Biomedical Signal Processing and Control*, 33:411–421, 2017.

[70] Emanuel Parzen. On estimation of a probability density function and mode. *The annals of mathematical statistics*, 33(3):1065–1076, 1962.

[71] Yagyensh Chandra Pati, Ramin Rezaiifar, and Perinkulam Sambamurthy Krishnaprasad. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In *Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on*, pages 40–44. IEEE, 1993.

[72] A. P. Pentland. Fractal-based description of natural scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6(6):661–674, Nov 1984.

[73] Danilo Cesar Pereira, Rodrigo Pereira Ramos, and Marcelo Zanchetta Do Nascimento. Segmentation and detection of breast cancer in mammograms combining wavelet analysis and genetic algorithm. *Computer methods and programs in biomedicine*, 114(1):88–101, 2014.

[74] Vasileios K Pothos, Christos Theoharatos, George Economou, and Spiros Fotopoulos. Robust classification of texture images using distributional-based multivariate analysis. In *Tools in Artificial Intelligence*. InTech, 2008.

[75] Lishan Qiao, Songcan Chen, and Xiaoyang Tan. Sparsity preserving projections with applications to face recognition. *Pattern Recognition*, 43(1):331 – 341, 2010.

[76] J. RamíRez, J. M. GóRriz, D. Salas-Gonzalez, A. Romero, M. LóPez, I. ÁLvarez, and M. GóMez-RíO. Computer-aided diagnosis of alzheimer's type dementia combining support vector machines and discriminant set of features. *Inf. Sci.*, 237:59–72, July 2013.

[77] Robert Robere. Interior point methods and linear programming. *University of Toronto, Ontario, Canada*, 2012.

[78] F. Rosenblatt. *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*. Spartan, 1962.

[79] Murray Rosenblatt. Remarks on some nonparametric estimates of a density function. *The Annals of Mathematical Statistics*, pages 832–837, 1956.

[80] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

[81] Alberto Santamaria-Pang, Sandeep Dutta, Sokratis Makrogiannis, Amy Hara, William Pavlicek, Alvin Silva, Brian Thomsen, Scott Robertson, Darin Okerlund, David A. Langan, and Rahul Bhotika. Automated liver lesion characterization using fast kvp switching dual energy computed tomography imaging. volume 7624, pages 76240V–76240V–10, 2010.

[82] H Schiabel, FLS Nunes, MC Escarpinati, and RH Benatti. Performance of a processing scheme for clustered microcalcifications detection with different images database. In *Engineering in Medicine and Biology Society, 2000. Proceedings of the 22nd Annual International Conference of the IEEE*, volume 2, pages 1199–1202. IEEE, 2000.

[83] L.G. Shapiro and G.C. Stockman. *Computer Vision*. Prentice-Hall, Upper Saddle River, NJ, 2001.

[84] Anushikha Singh, Malay Kishore Dutta, Rachid Jennane, and Eric Lespessailles. Classification of the trabecular bone structure of osteoporotic patients using machine vision. *Computers in biology and medicine*, 91:148–158, 2017.

[85] Robert A. Smith, Vilma Cokkinides, and Harmon J. Eyre. American cancer society guidelines for the early detection of cancer, 2003. *CA: A Cancer Journal for Clinicians*, 53(1):27–43, 2003.

[86] Milan Sonka, Vaclav Hlavac, and Roger Boyle. *Image processing, analysis, and machine vision*. Cengage Learning, 2014.

[87] P. Spellucci. A new technique for inconsistent qp problems in the sqp method. *Mathematical Methods of Operations Research*, 47(3):355–400, Oct 1998.

[88] John Suckling, J Parker, D Dance, S Astley, I Hutt, C Boggis, I Ricketts, E Stamatakis, N Cerneaz, S Kok, et al. The mammographic image analysis society digital mammogram database. In *Exerpta Medica. International Congress Series*, volume 1069, pages 375–378, 1994.

[89] Kaoru Tone. Revisions of constraint approximations in the successive qp method for nonlinear programming problems. *Mathematical Programming*, 26(2):144–152, Jun 1983.

[90] Simon Tong and Daphne Koller. Support vector machine active learning with applications to text classification. *Journal of machine learning research*, 2(Nov):45–66, 2001.

[91] J Touvier, R Winzenrieth, H Johansson, JP Roux, J Chaintreuil, H Toumi, R Jennane, D Hans, and E Lespessailles. Fracture discrimination by combined bone mineral density (bmd) and microarchitectural texture analysis. *Calcified tissue international*, 96(4):274–283, 2015.

[92] Joel A Tropp. Greed is good: Algorithmic results for sparse approximation. *IEEE Transactions on Information theory*, 50(10):2231–2242, 2004.

[93] Brijesh Verma, Peter McLeod, and Alan Klevansky. Classification of benign and malignant patterns in digital mammograms for the diagnosis of breast cancer. *Expert systems with applications*, 37(4):3344–3351, 2010.

[94] Richard A Waltz, José Luis Morales, Jorge Nocedal, and Dominique Orban. An interior algorithm for nonlinear optimization that combines line search and trust region steps. *Mathematical programming*, 107(3):391–408, 2006.

[95] World Health Organization (WHO). Assessment of fracture risk and its application to screening for postmenopausal osteoporosis. report of a who study group. pages 1–129, 1994.

[96] World Health Organization (WHO). Who scientific group on the assessment of osteoporosis at primary health care level. summary meeting report. 2007.

[97] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(2):210–227, Feb 2009.

[98] Jianchao Yang, John Wright, Thomas S. Huang, and Yi Ma. Image super-resolution as sparse representation of raw image patches. *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8, 2008.

[99] Florian Yger. Challenge ieee-isbi/tcb: Application of covariance matrices and wavelet marginals. 2014.

[100] Charles T Zahn. Graph-theoretical methods for detecting and describing gestalt clusters. *IEEE Transactions on computers*, 100(1):68–86, 1971.

[101] Joaquin Zepeda, Ewa Kijak, and Christine Guillemot. Sift-based local image description using sparse representations. In *Multimedia Signal Processing, 2009. MMSP'09. IEEE International Workshop on*, pages 1–6. IEEE, 2009.

[102] J. Zhang, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: a comprehensive study. *International Journal of Computer Vision*, 73:2007, 2007.

[103] Shu-Huan Zhao and Zheng-Ping Hu. Occluded face recognition based on block-label and residual. *International Journal on Artificial Intelligence Tools*, 25(03):1650019, 2016.

[104] Wei Zhao, Rui Xu, Yasushi Hirano, Rie Tachibana, and Shoji Kido. A sparse representation based method to classify pulmonary patterns of diffuse lung diseases. *Computational and Mathematical Methods in Medicine*, 2015, 2015. Article ID 567932.

[105] Xiaojin Zhu. Semi-supervised learning literature survey. Technical Report 1530, Computer Sciences, University of Wisconsin-Madison, 2005. $http://pages.cs.wisc.edu/ jerryzhu/pub/ssl_survey.pdf$.