A COMPUTATIONAL INTELLIGENCE AND MACHINE LEARNING-BASED FRAMEWORK FOR IMPROVING CASE-BASED COMPUTER-AIDED MEDICAL DECISION SYSTEMS WITH APPLICATION TO MAMMOGRAPHY

By

Maciej A. Mazurowski
M.S., Wroclaw University of Technology, 2005
M.A., University of Wroclaw, 2005

A Dissertation
Submitted to the Faculty of the
Graduate School of the University of Louisville
in Partial Fulfillment of the Requirements
for the Degree of

Doctor of Philosophy

Electrical Engineering Program
Department of Electrical and Computer Engineering
University of Louisville

December 2008
INFORMATION TO USERS

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleed-through, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

UMI Microform 3352062
Copyright 2009 by ProQuest LLC.
All rights reserved. This microform edition is protected against unauthorized copying under Title 17, United States Code.
A COMPUTATIONAL INTELLIGENCE AND MACHINE LEARNING-BASED FRAMEWORK FOR IMPROVING CASE-BASED COMPUTER-AIDED MEDICAL DECISION SYSTEMS WITH APPLICATION TO MAMMOGRAPHY

Submitted by

Maciej A. Mazurowski

A Dissertation Approved on

December 5, 2008
(Date)

by the Following Reading and Examination Committee:

Jacob Zurada, Dissertation Director

Amir Amini

Hichem Frigui

Tamer Inanc

Georgia D. Tourassi
ACKNOWLEDGMENTS

I would like to thank Dr. Jacek Zurada and Dr. Georgia Tourassi for guiding me in my research efforts. I would also like to thank Dr. Amir Amini, Dr. Hichem Frigui and Dr. Tamer Inanc for agreeing to serve on my committee.

I am very grateful to all my colleagues for their help and inspiration. I would like to especially acknowledge Dr. Piotr Habas, Dr. Kerem Muezzinoglu and Dr. Robert Saunders.

I would like to thank my parents Andrzej and Miroslawa for their constant support. Finally, I thank my friends for good times spent together and everlasting inspiration.
ABSTRACT

A COMPUTATIONAL INTELLIGENCE AND MACHINE LEARNING-BASED FRAMEWORK FOR IMPROVING CASE-BASED COMPUTER-AIDED MEDICAL DECISION SYSTEMS WITH APPLICATION TO MAMMOGRAPHY

Maciej A. Mazurowski
November 17, 2008

In case-based computer-aided decision systems (CB-CAD), previously acquired examples associated with different disease states are used to classify new cases. The case-based paradigm is becoming increasingly popular in computer-aided diagnosis. The primary reason is that large digital medical databases are now available, mainly of medical images. This incurs a need for efficient storage and use of the cases for medical decision making. Case-based systems are up to the task since new examples can be simply added to the system without the need for retraining. Further, the operating principles of case-based systems resemble the decision making process of physicians. Thus, case-based CAD systems are accepted more easily in the medical environment. Such systems, however, face certain limitations. Since all examples are stored in the system's database, the system induces large storage requirements. Furthermore, individual comparisons of a query case to all case-base examples increase the system's response time per query.

In this dissertation, a comprehensive computational intelligence and machine learning-based framework is proposed for optimization of case-based medical decision systems. It applies to two crucial components of case-based systems: decision algorithm and
case base. The study is performed in the context of a computer-aided decision system for detecting breast cancer in screening mammograms that has been previously published. Although the study is based on the specific CAD system, efforts were made to ensure that the proposed techniques will be easily applicable to other case-based CAD systems.

In the first stage of the research, an improvement of the decision function is proposed for the CAD system. The study hypothesis at this stage is that differentiating the importance of each case in the knowledge database may improve the system's performance. A problem of finding an optimal vector of importance weights is formalized as an optimization problem and a genetic algorithm is applied to solve it. The initial experimental results show that the proposed technique results in a statistically significant improvement of the classification performance.

In the second stage of the research some computational intelligence and machine learning techniques were used to optimize the case base of the CAD system by removing superfluous/detrimental examples. This type of optimization is of great significance since it can reduce storage requirements of the system, decrease response time of the system, and possibly improve system classification performance by removing misleading patterns. The results show that using computational intelligence and machine learning techniques allows for the database of examples to be significantly reduced while increasing performance of the system.

The third part of the dissertation research is devoted to building ensemble classifiers for improving the classification performance and reducing storage requirements of case-based systems. Two methods are proposed that automatically adapt the ensemble size to the problem. The new methods are compared to more traditional approaches. Experimental results show that the ensemble techniques provide a significant improvement in the classification performance of the CAD system while at the same time reducing the total number of examples used for classification.
The last part of this dissertation is devoted to comparison of all the proposed techniques, an extension of the ensemble approach and application of the ensemble approach to evaluate case-specific reliability of classifier decisions. Some insight into combining the proposed techniques to further improve performance is also offered.
<table>
<thead>
<tr>
<th>TABLE OF CONTENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACKNOWLEDGMENTS</td>
</tr>
<tr>
<td>ABSTRACT</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CHAPTER</th>
</tr>
</thead>
<tbody>
<tr>
<td>I. INTRODUCTION</td>
</tr>
<tr>
<td>A. CAD systems in screening mammography</td>
</tr>
<tr>
<td>B. A typical system for detection of masses in screening mammograms</td>
</tr>
<tr>
<td>C. Case-based computer-aided medical support systems</td>
</tr>
<tr>
<td>D. Information-theoretic CAD system</td>
</tr>
<tr>
<td>1. IT-CAD System Overview</td>
</tr>
<tr>
<td>2. Image similarity assessment algorithm</td>
</tr>
<tr>
<td>3. Decision algorithm</td>
</tr>
<tr>
<td>E. Clinically relevant classifier evaluation criteria</td>
</tr>
<tr>
<td>F. Motivation and goals of the dissertation research</td>
</tr>
<tr>
<td>II. DECISION OPTIMIZATION USING GENETIC ALGORITHMS</td>
</tr>
<tr>
<td>A. Introduction</td>
</tr>
<tr>
<td>B. Proposed decision algorithm</td>
</tr>
<tr>
<td>C. Genetic algorithm to optimize decision</td>
</tr>
<tr>
<td>1. Definition of the optimization problem</td>
</tr>
<tr>
<td>2. Genetic algorithm to find an optimal solution</td>
</tr>
<tr>
<td>D. Databases and experimental design</td>
</tr>
<tr>
<td>1. Databases</td>
</tr>
</tbody>
</table>
III. CASE BASE OPTIMIZATION
   A. Introduction .................................................. 28
   B. Methods: case base optimization ............................. 31
      1. Genetic algorithm based case selection ................. 33
      2. Greedy example selection .................................. 36
      3. Random mutation hill climbing ........................... 36
      4. Random selection ........................................... 37
   C. Databases and experimental design .......................... 37
      1. Databases .................................................... 37
      2. Experimental design ........................................ 38
   D. Experimental results ........................................ 39
      1. Satisfying limited storage requirements ................. 40
      2. Maximizing diagnostic performance ....................... 45
      3. Additional validation ..................................... 46
      4. Comparison to previously reported case base reduction techniques in CAD .................................................. 49
   E. Discussion ..................................................... 49

IV. DECISION AND CASE BASE OPTIMIZATION USING ENSEMBLE TECHNIQUES ................................. 54
   A. Introduction .................................................. 54
   B. Methods: constructing the ensemble ....................... 57
      1. Constructing sub-classifiers ............................... 58
      2. Combining sub-classifiers ................................ 59
      3. An incremental strategy .................................... 59
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>FIGURE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 1. An example of (a) a mammographic image and (b) the same image with regions of interest indicated by automated detection system</td>
<td>2</td>
</tr>
<tr>
<td>Figure 2. A block diagram of typical system for detection of masses in screening mammograms</td>
<td>3</td>
</tr>
<tr>
<td>Figure 3. Examples of ROIs depicting masses and normal tissue</td>
<td>5</td>
</tr>
<tr>
<td>Figure 4. A block diagram of information-theoretic CAD system</td>
<td>6</td>
</tr>
<tr>
<td>Figure 5. A block diagram of a genetic algorithm</td>
<td>17</td>
</tr>
<tr>
<td>Figure 6. Averaged ROC curves</td>
<td>23</td>
</tr>
<tr>
<td>Figure 7. Averaged ROC curves for high sensitivities</td>
<td>24</td>
</tr>
<tr>
<td>Figure 8. A typical histogram of weights assigned to images when system is optimized with AUC and [0,1] weight range</td>
<td>25</td>
</tr>
<tr>
<td>Figure 9. Diagram illustrating steps of the genetic algorithm</td>
<td>35</td>
</tr>
<tr>
<td>Figure 10. Average IT-CAD performance obtained with each case base selection method at various desired case base sizes</td>
<td>41</td>
</tr>
<tr>
<td>Figure 11. The most frequently selected examples by RMHC in Database 1 when the resulting case base size was 20</td>
<td>43</td>
</tr>
<tr>
<td>Figure 12. ROC curves for best IT-CAD performance obtained with each case base selection method</td>
<td>46</td>
</tr>
<tr>
<td>Figure 13. ROC curves for best IT-CAD performance obtained with each case base selection method for high sensitivities</td>
<td>47</td>
</tr>
</tbody>
</table>
Figure 14. A diagram showing the general idea of building an ensemble with a case-based system ................................................. 57
Figure 15. A diagram showing different kinds of ensemble techniques used in this study .......................................................... 61
Figure 16. A relation between AUC performance of the system and number of sub-classifiers ..................................................... 66
Figure 17. Validation and test ensemble performance at each step of building the ensemble for 'add-if-better' method ......................... 71
Figure 18. Validation and test ensemble performance at each step of building the ensemble for 'pick-best' method .......................... 72
Figure 19. Comparison of the best techniques evaluated in the dissertation .................................................................................. 77
Figure 20. ROC performance when GA-generated weights are used to reduce the case base compared with the performance of random mutation hill climbing ..................................................... 79
Figure 21. ROC performance of the original IT-CAD system, the featured-based LDA, and the ensemble IT-CAD system for the two grouping techniques (i.e., random division and feature-based, k-means clustering) 84
Figure 22. ROC performance of the original IT-CAD system, the featured-based LDA, and the ensemble IT-CAD system for the two grouping techniques (i.e., random division and feature-based, k-means clustering) for high classifier sensitivities ................................................................. 85
Figure 23. ROC performance of the two ensemble approaches (random division and feature-based, k-means clustering) for various numbers of groups ........................................................................... 87
Figure 24. Average ROC area performance for five groups of examples stratified by their variability (L - low, ML - medium-low, M - medium, MH - medium high, H - high) for different numbers of ensemble elements (a) 10, b) 20, c) 50) ................................................................. 89
Figure 25. Relation between the final output of the ensemble and standard deviation of the ensemble elements' outputs for different numbers of ensemble elements (a) 10, b) 20, c) 50). Red circles represent negative and blue crosses represent positive test samples. 

Figure 26. Relation between decision values for the RMHC-improved and the importance weights-improved system for one 10-fold crossvalidation split.

Figure 27. Relation between decision values for the the importance weights-improved system and the original system for one 10-fold crossvalidation split.

Figure 28. Relation between decision values for the RMHC-improved system and the original system for one 10-fold crossvalidation split.
LIST OF TABLES

TABLE                      PAGE

Table 1. Comparison of AUC and $a_{0.9}$AUC values for the original and the optimized IT-CAD system. "Range" indicates the range of change of the vector $w$ elements. ............................................ 21

Table 2. Two-tail p-values for the AUC pairwise comparison of techniques for three representative case base sizes in Database 1 ................. 42

Table 3. Best performance obtained by the system developed with different case base selection algorithms. For each algorithm, the numbers in brackets indicate the number of cases for which the best performance was obtained. ................................................. 45

Table 4. Maximum AUC test performance for the examined ensemble techniques ....................................................... 65

Table 5. Total number of distinct examples selected for the random selection approach ................................................. 69
CHAPTER I
INTRODUCTION

A. CAD systems in screening mammography

According to the American Cancer Society (ACS) estimate, in 2008, 182,460 women in the US will be diagnosed with breast cancer and 40,480 will die from this disease (ACS, 2008). According to this estimate, breast cancer is the most common form of cancer in women and second cancer-related cause of deaths in women. Currently the most effective way of detecting breast cancer is screening mammography.

In screening mammography, two X-ray images of each breast are taken: mediolateral oblique (MLO) view and cranio-caudal (CC) view. This results in four images per case. An example MLO view is shown in Figure 1a. Then, the images are analyzed by a radiologist who attempts to detect and diagnose signs of breast cancer such as masses, microcalcifications and architectural distortions. If there is a suspicion of cancer, the patient is sent to biopsy which provides a final confirmation of the detected abnormality being benign or malignant.

To assist a radiologist in interpretation of mammographic images so called computer-aided decision (CAD) systems are developed. Multiple successful applications of CAD systems have been presented in the literature (Friedman et al., 1999; Sampat et al., 2005; Doi, 2006; Kawamoto et al., 2005). Current research in CAD for mammography goes into two major directions. The first is detection of abnormalities in mammograms. The second is diagnosis of abnormalities regarding their malignancy status. The main focus of this dissertation is on CAD systems for detection of masses in
mammograms. However, the proposed techniques could be easily generalizable to diagnosis.

B. A typical system for detection of masses in screening mammograms

Computerized detection of masses in screening mammograms typically consists of two steps (see Figure 2). In the first step, the entire mammographic view is presented to the system and by scanning through the image, so called regions of interest (ROIs) are identified. Figure 1b presents a mammographic view with regions of interest identified by a detection system developed at Duke Advanced Imaging Laboratories.

Figure 1. An example of (a) a mammographic image and (b) the same image with regions of interest indicated by automated detection system.
Figure 2. A block diagram of typical system for detection of masses in screening mammograms

(Saunders et al., 2007). Typically, the first step is designed to operate at a high sensitivity. It means that the main focus of the first step is to detect as many masses as possible without worrying about a high false alarm rate. In the second step the ROIs are analyzed in more detail and they are classified as true mass or normal tissue. Some examples of positive and negative ROIs are presented in Figure 3. The second step is also called false positive reduction since it attempts to distinguish true masses from false positive marks and improve the overall specificity. The second step of the mass detection task will be the main testbed for the methods presented in this dissertation.

Several approaches have been proposed for false positive reduction. Typically they involve feature extraction and classification using supervised learning principles (Sahiner et al., 1996a), (te Brake et al., 2000). However, featureless techniques were developed as well (Tourassi et al., 2003b; Tourassi et al., 2007a).
C. Case-based computer-aided medical support systems*

The task of false positive reduction is a classification task, i.e. one where a query object (i.e. ROI) has to be assigned to one of predefined classes (i.e. mass/normal tissue) based on previously acquired knowledge. Two main approaches to classification exist in machine learning: rule-based and case-based. In rule-based systems, the acquired examples are used to construct decision rules. These rules are further used to make decisions regarding new, unknown cases. Popular examples of this approach are artificial neural networks (Zurada, 1992) where knowledge about the previous cases is incorporated into the weights of the inter-neuron connections. Other examples are systems based on statistical principles (Duda et al., 2000) or decision trees (Mitchell, 1997). In rule-based systems, the training examples are used exclusively for training the CAD system. As soon as the system is trained, it is applied independently to the new, incoming cases using only the rules found during training.

In contrast, there is virtually no training in case-based systems (Aha et al., 1991), (Mitchell, 1997). Instead, the available examples are stored in the database of the system to serve as established knowledge during the classification stage. When a new query case is presented to a case-based system, it is compared with known examples stored in the case base. Then, classification of the query case is performed based on these pair-wise comparisons. This type of decision making is also known as evidence-based since it relies on prior evidence of established cases.

Recently, case-based CAD (CB-CAD) systems (Chang et al., 2001; Schmidt et al., 2001; Tourassi et al., 2003b; El-Naqa et al., 2004a; El-Naqa et al., 2004b; Tourassi et al., 2007a; Park et al., 2007), have been gaining popularity in the medical domain. These systems can be divided into two general types: feature-based and featureless. In feature based systems, various features are extracted from images. Then, incoming

*Parts of this section have been included in (Mazurowski et al., 2008e) and (Mazurowski et al., 2008b).
queries are classified based on the class membership of its neighbors in the feature space. Typically, classical measures such as Euclidean metric are used to assess distance between cases. In featureless systems (Tourassi et al., 2003b; Tourassi et al., 2007a) no features are extracted from images and similarity between cases is evaluated using statistical measures. Information-theoretic CAD (IT-CAD) system belongs in the second group. A false positive reduction problem with IT-CAD system will serve as the main testbed for the algorithms examined in this dissertation.

Figure 3. Examples of ROIs depicting masses and normal tissue
D. Information-theoretic CAD system*

1. Information-theoretic CAD System Overview

Figure 4 presents an overview of the IT-CAD system. The decision process is composed of two separate steps. First, an incoming query image is compared to all the images in the knowledge database. The result of this comparison is a vector of similarity indices between the query image and each of the images in the database. This vector, along with the ground truth for each of the images in the database is utilized in the second step to make a binary decision, whether a query image depicts a mass or a normal tissue.

The decision process of case-based CAD systems and IT-CAD in particular aims at mimicking the radiologist's approach to decision making. Typically, the radiologist

* This section with some modifications has been included in (Mazurowski et al., 2008b).
recalls the mammograms seen before along with their ground truth to make a proper classification for the examined query case. To implement this process in a CAD system, the CAD designer needs to select (1) an algorithm for evaluating similarities between the mammograms and (2) an algorithm to make a decision based on those similarities. The algorithms implemented in IT-CAD are described in detail below.

2. Image similarity assessment algorithm

To quantify the similarity between images, a so called similarity measure has to be used. A wide variety of similarity assessment methods are available in the literature. A comprehensive review of the topic can be found in (Smeulders et al., 2000). Typically the similarity is assessed using a set of features previously extracted from images. Here, a featureless, approach is utilized. Namely, the classical statistical concept of mutual information (MI) is used (Cover, 1991; Maes et al., 1997; Tourassi et al., 2001). It has been shown to be a very efficient measure of similarity between images (Maes et al., 1997; Viola and Wells, 1997) and was successfully applied in CAD systems (Tourassi et al., 2007a; Tourassi et al., 2003b). In information theory mutual information, also called mutual entropy, describes a statistical dependence between two random variables. For discrete random variables, mutual information $I$ is defined as

$$I(X; Y) = \sum_x \sum_y P_{XY}(x, y) \log_2 \frac{P_{XY}(x, y)}{P_X(x) P_Y(y)}$$

(1)

The mutual information measure can be applied to image similarity assessment by substituting random variables with image histograms. To find the value of $I$ between two images $X$ and $Y$, $P(X)$ and $P(Y)$ in (1) have to be replaced by the intensity histograms of images $X$ and $Y$, respectively, and $P(X,Y)$ has to be replaced by the 2D joint histogram of the two images (Maes et al., 1997). In this study, the normalized mutual information (NMI) index was used, defined as
\[
NMI = \frac{2I(X,Y)}{H(X) + H(Y)}
\]

where \(H(X)\) and \(H(Y)\) are the entropies of the individual images. The normalization results in \(NMI = 1\) for identical images and \(NMI = 0\) for two completely independent (i.e., unrelated) images.

3. Decision algorithm

After similarities between the query image and the images in the case base are quantified using the NMI measure, a decision has to be made whether the query image depicts a mass or a normal case. Several approaches to this problem have been presented (Mitchell, 1997; Chang et al., 2001; Tourassi et al., 2003b). Tourassi et al. (Tourassi et al., 2003b) proposed the Decision Index (DI) defined as

\[
DI_1(Q) = \frac{1}{m} \sum_{i=1}^{m} NMI(Q, M_i) - \frac{1}{n} \sum_{j=1}^{n} NMI(Q, N_j)
\]

where \(NMI(X,Y)\) is the normalized mutual information between images \(X\) and \(Y\), \(Q\) is a query image, \(M_i\) are images corresponding to positive examples (masses), \(N_j\) are images corresponding to negative examples (normal), \(m\) and \(n\) are the number of positive and negative examples from the databank, respectively. The two parts of this index express sums of similarity indices to images depicting masses and to ones depicting normal tissues. If the query image is in general more similar to images depicting masses, the normalized sum of NMIIs with those images will be higher than the normalized sum of NMIIs with images depicting normal tissue which will result in a high value of the DI. In the opposite situation, when according to NMI, the query image is more similar to the images depicting normal tissues, the value of DI will be small. Finally, a threshold on the decision index has to be applied, such that when the decision index for a given query case is larger than the threshold value, the query
E. Clinically relevant classifier evaluation criteria*

Traditionally, the accuracy has been used to evaluate classifier performance. It is defined as the total number of misclassified example divided by the total number of available examples for a given operating point of a classifier. For instance, in a 2-class classification problem with two predefined classes (e.g., positive diagnosis, negative diagnosis) the classified test cases are divided into four categories:

- true positives (TP) – correctly classified positive cases,
- true negatives (TN) – correctly classified negative cases,
- false positives (FP) – incorrectly classified negative cases, and
- false negatives (FN) – incorrectly classified positive cases.

Therefore, the accuracy is

\[
\frac{FP + FN}{TP + TN + FP + FN}
\]

This evaluation criterion is of a limited use in clinical applications for many reasons. First, misclassification error varies dramatically depending on class prevalence and it can be very misleading in clinical applications where the most important class is typically underrepresented. For example, if the presence of cancer is 5% in the test dataset (a typical clinical scenario), a classifier that detects 100% of cancer-free cases and 0% of cancer cases achieves a seemingly low 5% misclassification error. From a clinical perspective though, this is an unacceptable performance since all cancer patients are misdiagnosed and thus left untreated. Second, in medical decision, different kinds of misclassifications have different cost. For example in breast cancer diagnosis, a false positive classification translates into an unnecessary breast biopsy, associated with

* This section has been included in (Mazurowski et al., 2008a)
both emotional and financial cost. A false negative classification, however, means a missed cancer which in turn can be deadly. Such differences are not taken into account by the average classification error. Finally, the average classification error depends on the classifier's operating threshold. Since many classification systems (such as neural networks) provide a decision variable of multiple possible values, choosing the optimal decision threshold can be challenging. It also makes it impossible to make direct comparisons among CAD systems that are designed to operate with different decision thresholds.

To account for these issues, Receiver Operator Characteristic (ROC) analysis is commonly used in the clinical CAD community (Obuchowski, 2003). There are two underlying assumptions in ROC analysis. First, the decision problem is binary. Thus, ROC analysis is applicable to 2-class classification problems. Second, the two classes are mutually exclusive. Thus, the clinical scenario where a patient may belong to both classes (e.g., both diseases coexist in the patient) is not allowed in the ROC framework. ROC curve describes the relation between two indices: (i) true positive fraction (TPF) and (ii) false positive fraction (FPF) defined as follows:

\[
TPF = \frac{TP}{TP + FN}
\]

\[
FPF = \frac{FP}{TN + FP}
\]

A conventional ROC curve plots TPF (or sensitivity) vs. FPP (or [1 − specificity]) for every possible decision threshold imposed on the decision variable. By providing such complete picture, ROC curves are often used to select the optimal decision threshold by maximizing any pre-selected measure of clinical efficacy (e.g., accuracy, average benefit, etc.).

In ROC analysis, the most commonly used FOM is the area under the ROC curve (AUC). The AUC index for useful classifiers is constrained between 0.5 (represent-
ing chance behavior) and 1.0 (representing perfect classification performance). CAD algorithms are typically designed to maximize the area index. However, in cancer screening applications, it is often critical to achieve sufficiently high sensitivity (also known as detection rate). In such applications, it is often expected that the CAD algorithm will need to achieve at least 90% sensitivity. Accordingly, researchers have proposed the partial AUC index ($\mu$AUC, where $\mu$ indicates the lowest acceptable sensitivity level) as a more meaningful FOM (Jiang et al., 1996). Detailed description of ROC analysis and its utilization for CAD evaluation can be found in (Bradley, 1997; Metz et al., 1998b; Jiang et al., 1996).

F. Motivation and goals of the dissertation research

With growing databases of examples, CB-CAD systems seem adequate for the automated medical decision task. Significant effort, however, is needed to adapt and improve the existing case-based techniques to meet very strict requirements in medical decision making. The goal of this work is to improve two basic components of CB-CAD:

1. decision algorithm

2. case base management

The dissertation is organized as follows. In Chapter II a novel decision optimization algorithm is proposed that finds optimal set of importance weights for images stored in the case base. In Chapter III, several computational intelligence and machine learning techniques are applied to reduce size of the reference database while at the same time maximizing the system's performance. Chapter IV presents and extensive evaluation of ensemble techniques as applied to case-based systems. Two incremental adaptive methods are proposed for building ensembles that provide a significant improvement in the system classification performance as well as a notable reduction of the database.
of examples. In Chapter V an initial investigation of some additional directions of research that emerge from the dissertation research is presented and some future directions are suggested.

The study is based on the featureless IT-CAD system as an example of CB-CAD facing typical difficulties of this kind of system. A particular effort, however, has been made to ensure that the proposed and analyzed techniques are applicable to other CB-CAD systems as well. The overall goal of this work is to propose and evaluate a comprehensive set of techniques for construction and optimization of case-based systems for analysis of mammograms.
CHAPTER II
DECISION OPTIMIZATION USING GENETIC ALGORITHMS*

A. Introduction

The CB-CAD systems currently reported in the literature are often based on the assumption that the examples stored in the knowledge database are equally important in the decision making process or their importance is dependent on their order of retrieval. The goal of this study is to challenge this practice and test whether assigning different importance to the knowledge cases may improve the CB-CAD performance. The study hypothesis is based on the common sense observation that when a decision is being made about a query case, its similarity to certain previous cases may be more important than similarity to others. Furthermore it may occur that similarity to certain previous cases is simply misleading. The proposed framework accommodates both scenarios.

Following the study hypothesis, a new decision index is proposed that involves a vector of weights, each of which is a measure of importance of the particular image in the knowledge databank. Then, a genetic algorithm (GA) (Eiben and Smith, 2003; Michalewicz, 1999) is utilized to find the optimal weight vector. GAs have been widely used to construct and optimize CAD tools for many clinical applications (Pena-Reyes and Sipper, 2000), including breast cancer detection (Campanini and Lanceonelli, 2006). These efforts aimed mainly towards optimizing parameters of CAD systems (Gurcan et al., 2002; Bevilacqua et al., 2001) optimizing the selection of features.

* This chapter, with some modifications, has been included in (Mazurowski et al., 2008b).
for classification (Anastasio et al., 1998; Sahiner et al., 1998; Sahiner et al., 1996b; Boroczky et al., 2006), directly training the classifiers (Fogel et al., 1998; Pena-Reyes and Sipper, 1999), as well as other applications (Peng et al., 2006).

The framework proposed here is tailored to optimize the performance of a system measured by clinically relevant criteria. The Receiver Operator Characteristic and two related performance indices are used to optimize and test the system. Such practice will allow a better fit of a classifier to the particular clinical task.

The proposed optimization scheme is tested with respect to the featureless IT-CAD system presented previously in (Tourassi et al., 2003b). The system was developed for the automated detection of breast masses in screening mammograms. The proposed approach, however, is not limited to this system and may be applied to virtually any case-based CAD.

B. Proposed decision algorithm

A more general decision function is introduced below. It offers an advantage that no assumption about equal importance of each image in the database is made a priori. It is defined as

$$D_{I_2}(Q) = \frac{1}{m} \sum_{i=1}^{m} w_{pos,i} NMI(Q, M_i) - \frac{1}{n} \sum_{j=1}^{n} w_{neg,j} NMI(Q, N_j) \tag{7}$$

where $w_{pos,i}$ is the importance weight associated with the $i$-th positive example, $w_{neg,j}$ is the importance weight associated with the $j$-th negative example. Low values of $w_{pos,i}$ or $w_{neg,j}$ diminish the contribution of similarity between a query image $Q$ and image $M_i$ or $N_j$, respectively, to the decision index. In the case of the weight equal to 0, a corresponding image will not play any role in the classification. Note also that $D_{I_1}$ can be treated as a special case of $D_{I_2}$, where all the weights are equal to 1 (equal importance of all images in the database). Having a new definition of the
decision function, the question arises of how to find the importance of each of the images in the database. Here, the problem of finding optimal weights is formalized as an optimization problem and a genetic algorithm is proposed to find an optimal solution.

C. Genetic algorithm to optimize decision

1. Definition of the optimization problem

To approach the presented problem as an optimization task, a suitable objective function (also called a cost function) needs to be defined. Any classification performance measure applied to a resulting CAD system is an appropriate choice for the objective function. In medical diagnosis, however, the Receiver Operator Characteristic curve (Bradley, 1997; Fawcett, 2006; Obuchowski, 2003) is of particular interest and has been widely used to assess classifiers in CAD. In the proposed framework, both AUC and 0.9AUC indices are utilized for optimization to show that the system can be optimized according to a selected, clinically relevant figure of merit. In this study, the nonparametric Wilcoxon approach (Bradley, 1997) of estimating AUC and pAUC was used. Note that, for the given set of examples in the database, the decision index for each query, and the AUC and pAUC calculated based on those indices, depends solely on the vector \( w \) (see (7)). Therefore, the objective function is a function of \( w \) and will be denoted simply as \( f^{AUC}(w) \) or \( f^{0.9AUC}(w) \). The \( w \) denotes a vector of weights. Since there is one weight assigned to each example (\( w_{pos,i} \) to positive examples and \( w_{neg,i} \) to negative examples, the length of the vector \( w \) is equal to the number of cases in the database of the system \( (n + m) \).

The presented problem can now be defined as the problem of finding a vector \( w^* \) such that \( \forall w, f^{AUC}(w) \leq f^{AUC}(w^*) \) and formally, finding
\[ w^* = \arg \max_w [f^{AUC} (w)] \]  

(8)

In the case of optimizing \(_{0.9}AUC\), the corresponding problem is

\[ w^* = \arg \max_w [f^{_{0.9}AUC} (w)] \]  

(9)

A large variety of optimization methods have been proposed for such optimization problems. Selection of the optimization technique depends on the properties of the objective function. Here, the objective function is not given by an arithmetic expression and can be only sampled for each value of \( w \). Furthermore, the objective function is not continuous and thus not differentiable. For these reasons the traditional optimization techniques such as the gradient descent (Bertsekas, 1999) are not applicable. As a consequence, a so-called global optimization algorithm, namely a genetic algorithm, was used to find a solution.

2. Genetic algorithm to find an optimal solution

The genetic algorithm is an optimization technique loosely inspired by the theory of natural selection and genetics (Eiben and Smith, 2003; Michalewicz, 1999; Campanini and Lancoconelli, 2006). The basic concept of the genetic algorithm is to treat different solutions to a problem as competing individuals. Solutions are represented in the chromosomes (vectors of numbers) of these individuals. The best solution evolves among the individuals by means of crossection, mutation and natural selection. When selecting parents or individuals to survive, each individual is evaluated according to its fitness (the better the fitness is, the better the chances for survival and becoming a parent are), which is simply a value of an objective function (in this case \( f^{AUC} (w) \) or \( f^{_{0.9}AUC} (w) \)) in the point represented by its chromosome (in this case \( w \)). A diagram of a simple genetic algorithm is presented in Figure 5 (different variations can be
When using a genetic algorithm, one must make a decision regarding the following properties: representation of the solutions in chromosome, population initialization, algorithm to select parents, recombination technique, mutation algorithm and algorithm of selection individuals to survive. The following is a short description of all these steps in the algorithm used in this study (for a more comprehensive overview of these steps, see (Eiben and Smith, 2003) and (Michalewicz, 1999)).

- **Representation of solutions in chromosomes.** Since a candidate solution is a vector of weights, each chromosome has to represent a vector of real numbers. Therefore a real-valued representation is used such that there is a real number on each position of the chromosome (each gene).

- **Population initialization.** Each position of each chromosome was initialized with a random value in a given range with uniform probability distribution.

- **Parent selection.** This operation selects the parents for the next generation.
Here, the stochastic universal sampling (Gen and Cheng, 1997) is used for this purpose. In this algorithm all the individuals are assigned intervals proportional to their fitness. The intervals are placed on a line in random order. Then, starting from the beginning of the line parents are picked by moving along the line in equal steps. This way each individual will be selected the expected number of times (this condition is not satisfied for example for the roulette rule).

- **Crossover.** Crossover operation is used to generate new individuals by combining chromosomes of selected parents. Here, the uniform (also called random or scattered) crossover is used. It means that each gene of a child comes unaltered from one of the parents with equal probability for both parents.

- **Mutation.** Mutation operation alters selected genes in the population to introduce an additional random factor into the search. In this study a uniform mutation is used which replaces the randomly selected genes with random values uniformly distributed in the range of the particular position.

Additionally in the GA used in this study an elite selection technique was applied. This technique assured that the best individuals always survive to a next generation.

D. Databases and experimental design

1. Databases

The study was based on a set of mammograms extracted from the Digital Database for Screening Mammography (DDSM) (Heath and et al., 1998), collected at the University of South Florida. The mammographic films were digitized using the LUMISYS scanner at 50 microns per pixel. From the mammograms depicting true masses,
512 x 512 pixel regions of interests (ROI) were extracted. The resulting 901 mass ROIs (489 malignant and 412 benign masses) were centered on the physicians annotation. From the normal mammograms and from abnormal mammograms without any annotations for one of the breasts, 919 normal ROIs were extracted randomly. The normal ROIs were also 512 x 512 pixels in size. The prevalence of mass ROIs in the database was roughly 0.5.

2. Experimental design

To assess the effectiveness of the proposed optimization framework, the genetically optimized CAD system (GA-IT-CAD) was compared to the original IT-CAD system with all the patterns in the databank having equal importance.

To evaluate the performance of the systems, a 10-fold crossvalidation scheme was used. First, the entire dataset was divided into 10 equal-size, mutually exclusive parts (folds). Then, each fold was used once as a testing set leaving the remaining nine folds for CAD system development. This resulted in ten splits with 1638 examples in the development dataset and 182 examples in the testing dataset for each split.

For each split the development dataset was used to construct the GA-IT-CAD system while the test set was left for the final testing. Construction of the system consisted of two separate tasks. The first task was choosing images from the development dataset to be included in the knowledge databank of the GA-IT-CAD system for further use in the decision making process. The second task was to find the importance weights for the images in the database.

To complete these tasks, the training set can be utilized in various ways. One way is to include only some of the images from the training dataset in the knowledge databank of the GA-IT-CAD system. In such a case the remaining images can be utilized to calculate the objective function \( f^{AUC}(w) \) in a GA run in order to find an optimal set of weights for the images stored in the knowledge databank. This
approach, however, has significant disadvantages. Namely, not all images available from the development dataset are included in the knowledge databank of the CAD and since the excluded images may be important for the problem at hand, the performance of the resulting system could be significantly compromised. This approach should be used only for very large development datasets.

Since data availability is often limited in medical applications, more efficient data handling schemes should be applied to capitalize on the available clinical cases. Consistent with many CAD-related studies published before, the leave-one-out data handling scheme was used in the development process of this study (Efron and Tibshirani, 1993).

Initially, each of the $N$ images from the training dataset is included in the knowledge database of the GA-IT-CAD. Then, in the optimization process using GA, the objective function for each individual ($f^{AUC}(w)$ or $f^{95\%AUC}(w)$) is computed in the following way: first, $N$ decision indices are obtained, each of them by using one of the images from the knowledge databank as a query. When $image_k$ is provided as a query to the CAD, its corresponding weight is temporarily set to 0 and the decision index is calculated according to (7). This way self-similarity ($NMI(image_k, image_k)$) of the image does not affect the resulting decision index. Such calculation is repeated for all images from the databank, which results in $N$ decision indices. The resulting decision indices along with the ground truth of the corresponding images are used to calculate $f^{AUC}(w)$ or $f^{95\%AUC}(w)$ using the Wilcoxon approach.

After the optimization, for each of the data splits, the "trained" GA-IT-CAD system was tested on the testing set (i.e., the second part of the available database).

To compare the optimized CAD system with the original, a paired student t-test was used without an assumption about equal variances of the two populations to assure the conservativeness of the results. A very similar scheme for statistical comparison of two models based on 10-fold crossvalidation was presented in (Bradley, 1997). Such
Table 1. Comparison of AUC and 0.9AUC values for the original and the optimized IT-CAD system. "Range" indicates the range of change of the vector \( w \) elements.

<table>
<thead>
<tr>
<th>Objective:</th>
<th>AUC</th>
<th>0.9AUC</th>
<th>Original</th>
</tr>
</thead>
<tbody>
<tr>
<td>Range:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[0,1]</td>
<td>0.907±0.024</td>
<td>0.905±0.026</td>
<td>0.905±0.022</td>
</tr>
<tr>
<td>[-1,1]</td>
<td>0.905±0.026</td>
<td>0.533±0.104</td>
<td>0.532±0.100</td>
</tr>
</tbody>
</table>

A scheme allows for incorporating in the analysis the uncertainty of performance estimation due to a finite test set as well as the random factor due to a stochastic nature of the GA.

E. Experimental results

The GA optimization was performed using Matlab programming environment and Matlab Genetic Algorithm and Direct Search Toolbox. The following standard parameters of the GA algorithm were used for the optimization:

- number of individuals in the population: 100
- number of generations: 100
- mutation ratio: 0.01
- crossover fraction: 0.8

The number of individuals in the population and the number of generations were selected to assure extensive search without compromising the time complexity of the algorithm. Some experimentation was performed to select the optimal mutation ratio. The crossover fraction of 0.8 was a default value of this parameter. Table 1 shows
the average test performance of the original and optimized CAD systems across all ten splits. The optimization was performed with two different objective functions: $f^{AUC}(w)$ and $f^{0.9AUC}(w)$ and two different ranges of $w$: $[-1,1]$ and $[0,1]$ resulting in four separately analyzed scenarios (each combination of an objective function and the range of $w$). The $w$ range of $[0,1]$ corresponds to a situation where the minimum importance of a pattern mean that the pattern is not considered in the decision process. In the case of $w$ in the range $[-1,1]$ a situation where a pattern is misleading is taken into account. Thus, it is possible through a negative weight that an example will serve as an example of the opposite class in the decision process.

It is apparent from Table 1 that the proposed optimization framework improved the overall performance of the CAD system from AUC=0.865±0.030 to AUC=0.907±0.024. The improvement was statistically significant (two sided $p < 0.0005$). The best performance was obtained when the CAD system was optimized using $f^{AUC}(w)$ with $w$ ranging in $[0,1]$. The overall performance with other optimization parameters (objective function and range of $w$) was also statistically significantly better than the performance of the original CAD system. The obtained results depended on the choice of the $w$ range and generally better performance was obtained for the range $[0,1]$. Applying $f^{AUC}(w)$ and $f^{0.9AUC}(w)$ had similarly beneficial effect on the CAD overall performance.

It can also be seen that the proposed optimization framework allowed for a large and consistent (nine of ten splits) improvement of the system in terms of highly clinically relevant partial area under the ROC curve. The mean $0.9AUC$ index improved from 0.466±0.091 for the original CAD system to 0.540±0.094 for the CAD system optimized with $f^{0.9AUC}(w)$ and $[0,1]$ range for $w$. The improvement was statistically significant (two sided $p < 0.001$). Applying other objective function and other $w$ range also offered a statistically significant improvement as compared to the original CAD system (the largest p-value was less than 0.05).
Figure 6. Averaged ROC curves: (A) GA-IT-CAD optimized with AUC and [0,1] weight range, (B) GA-IT-CAD optimized with 0.9AUC and [0,1] weight range, (C) GA-IT-CAD optimized with AUC and [-1,1] weight range, (D) GA-IT-CAD optimized with 0.9AUC and [-1,1] weight range and (E) original IT-CAD system.

The ROC curves for the original CAD system and the optimized CAD systems were reconstructed by averaging the parameters of ROC curves for particular splits. Parametric estimations of ROC curves were used for that purpose with the ROCKIT software developed at the University of Chicago (Metz et al., 1998a; Metz et al., 1998b). The reconstructed curves are shown in Figure 6 and Figure 7. It can be seen that the proposed optimization framework improved the performance of the system in the regions of high specificity (for FPF=0.1, TPF improved from 0.583 to 0.738) as well as in the regions of high sensitivity (for TPF = 0.9, specificity improved from 0.653 to 0.738 and for TPF = 0.95, specificity improved from 0.536 to 0.612). The values of TPF and FPF were read directly from the reconstructed ROC curves. A
Figure 7. Averaged ROC curves for high sensitivities: (A) GA-IT-CAD optimized with AUC and [0,1] weight range (---), (B) GA-IT-CAD optimized with 0.9AUC and [0,1] weight range (--), (C) GA-IT-CAD optimized with AUC and [-1,1] weight range (- -), (D) GA-IT-CAD optimized with 0.9AUC and [-1,1] weight range (---), and (E) original IT-CAD system (-).

Linear interpolation was used when the estimation was not given for a particular TPF of FPF.

Figure 8 shows a typical histogram of weights assigned to examples when the system is optimized with AUC as a fitness function and weights vary in the range [0,1]. The distribution of weights is nearly uniform which means almost equal number of images with large (weight close to one), marginal (weight close to zero) and other levels of importance in the decision process. The distributions were similar when the system was optimized with 0.9AUC as a fitness function. In the case when weights varied in the range [-1,1] the number of positive weights generally exceeded the number of negatives weights. The number of examples with negative weights, however, was
Figure 8. A typical histogram of weights assigned to images when system is optimized with AUC and [0,1] weight range

large. This can be explained partially by the existence of misleading examples in the database of images and partially by inability of the GA to find good solutions when the weights are uniformly initialized in the range [-1,1]. A CAD system designer may incorporate a prior knowledge about a problem at hand by introducing different initializations of weights.

As described in the previous section, the optimization was repeated for each of the ten splits of the data. Such data handling allowed for analysis of consistency in weights assigned to each particular image (each image appeared in nine out of ten development datasets). It was observed that the weights assigned to a particular image can vary significantly including cases when a particular example has a large importance (a weight close to one) for one split and marginal importance (a weight close to zero) for other. This can be explained by several factors. Most importantly
there may be more than one example of a particular group that has to play an important role in making a decision. If only one representative of one group is needed, then different representatives may be selected (weight close to one) in each optimization process. In term of the optimization problem at hand it means that there are multiple equally good solutions to the problem often corresponding to very different weights. Furthermore, importance of particular images may vary with presence or absence of other images in the database. Since different data splits altered the development dataset it may be considered as an important factor influencing the resulting weights for particular images. Finally, GA is not guaranteed to find an optimal solution for a given problem. Thus, for certain algorithm runs, non-optimal weights are likely to occur, further increasing variability of weights for particular images among runs.

F. Discussion

In this chapter a new optimization framework for case-based CAD systems was presented. The study hypothesis was that each image stored in the knowledge databank of CB-CAD system has different importance in the diagnostic process. A new decision index was proposed, that incorporates different importance weights for each stored example. Then, a genetic algorithm approach was employed to find an optimal set of weights according to clinically relevant objectives. It was shown experimentally that the modified CAD system performs significantly better as evaluated by the area under ROC curve and partial area under the ROC curve indices. As a result, the optimized CAD system is characterized by a higher mass detection rate for a given specificity as well as a reduced false positive rate for a given sensitivity.

The technique presented in this chapter has some similarities to the importance evaluation based on the order of retrieval (Tourassi et al., 2003a). A significant difference, however, must be stressed. In the approach based on the order of retrieval, the importance weights of patterns in the database are query-dependent, while in the
approach presented here, they are universal among all the incoming queries.

There are limitations to the technique presented in this chapter. Namely, an optimization process has to be performed each time, the database of images is modified (e.g. by adding new examples). The optimization process, however, is run offline and does not affect a response time of the constructed system.

Although the proposed GA approach was demonstrated with respect to CAD system for a breast cancer detection, it is certainly applicable to any type of case-based CAD system (feature-based and featureless). As CB-CAD systems become more popular for clinical applications, sophisticated techniques are expected to emerge for more effective construction and use of knowledge databases. The proposed technique is a step in this direction.
CHAPTER III
CASE BASE OPTIMIZATION*

A. Introduction

A focus of this study is on one of the distinct issues tied with case-based systems, namely managing the case base (i.e., database of known examples). Storing the examples directly in the database of the system has advantages over the rule-based approach. The major one is that the case base can be updated without retraining the system. Such need arises when newly acquired examples need to be included. On the other hand, there are some drawbacks as well such as the ever increasing storage requirement. Data storage becomes a major problem when a large number of examples needs to be stored in the system to maintain case base variety while each example requires a lot of storage space. This is a typical scenario when storing clinical images. Another problem arising in a large case-based system is the computational cost of classification due to the need for comparing the query case to many examples from the case base. This problem is intensified when a single comparison incurs a large computational cost due to an elaborate (dis)similarity measure. An example of a case-based medical decision support system facing such difficulties is the IT-CAD system previously proposed (Tourassi et al., 2003b; Tourassi et al., 2007a) for mass detection in screening mammograms. The system uses the information theoretic concept of mutual information to assess the similarity between two mammographic cases. The first practical difficulty is the fact that the knowledge cases are stored in

* This chapter, with some modifications, has been included in (Mazurowski et al., 2008e). The preliminary results were published in (Mazurowski et al., 2007).
the database in the form of full images and not feature vectors causing large storage requirements. The second difficulty is the longer calculation time of assessing the mutual information between two mammographic images instead of just comparing the extracted image features.

The disadvantages of case-based systems can be alleviated by proper management of the system's case base. In fact, careful selection of the examples included in the database may not only decrease the decision time and storage requirements but possibly improve CAD performance by removing unnecessary or even misleading examples.

Case base optimization is a long-standing problem in artificial intelligence (Blum and Langley, 1997; Wilson and Martinez, 2000). The problem can be stated in several different ways depending on the ultimate optimization goal. In medical CAD systems, two main properties should be taken into account: (i) system performance and (ii) response time per query. For a case-based CAD system, both properties are directly tied to the size of the case base. These two properties can be of different importance in different environments. Priority is typically given to system performance, measured often by ROC analysis (Bradley, 1997; Metz et al., 1998b; Obuchowski, 2003), system sensitivity or positive predictive value for a specific decision threshold, etc. In such scenario, certain minimal conditions on system performance may have to be met and the system designer chooses the smallest subset of available examples that meets such conditions. In another scenario, system performance needs to be maximized at any expense. Then, the case base size is of no concern. In most situations, the case base size should be reasonably controlled while ensuring that the diagnostic performance of the system is not compromised. In this study, all of these scenarios in the formalization of the problem, the proposed techniques, and the experimental evaluation are accounted for.

The existing case base reduction algorithms can be classified into three general groups. The first group is comprised by algorithms based on the nearest neighbor
editing rule. These algorithms rely on the distances between examples and their class membership to remove those on the borders of classes, noisy examples, etc. (Hart, 1968; Aha et al., 1991; Aha, 1992). The algorithms from the first group are well suited to feature based systems and metric spaces. The second group are iterative algorithms that evaluate the classification performance of the case-based system when relying on subsets of examples. A typical algorithm falling into this group is random mutation hill climbing (Skalak, 1994) where the subsets are randomly modified and the old set is replaced if the performance obtained by the new set is strictly better. This group of algorithms is easily applicable with feature-based and featureless systems as well as suitable for metric/nonmetric spaces (Pekalska et al., 2006). Finally, the algorithms belonging in the third group operate under very different principles. Instead of selecting the most useful examples from the available pool, they modify the existing examples or create new, more representative ones. Such an algorithm is one called the prototypes (Chan, 1974) which merges examples that are close to each other using weighted averaging. Since these algorithms rely on feature-based representation of cases, they are not applicable to featureless systems. Furthermore, this third group of case selection algorithms seems inappropriate for clinical evidence-based systems since it alters clinical evidence. Regardless, case base selection algorithms are typically developed and evaluated using classification accuracy (number of correctly classified queries vs. total number of queries) as the figure of merit.

Recently case base optimization gained some interest in the CAD community. For example, Park et al (Park et al., 2007) proposed a variation of the edited nearest neighbor rule (Wilson and Martinez, 2000) for case base reduction. The authors evaluated the technique with a feature-based CAD system for false positive reduction in screening mammograms. The method uses leave-one-out technique to assign a decision variable to each example in the case base. Then, two thresholds $T_1$ and $T_2$ are set. Given these thresholds the reduction is performed such that an example is
removed if it depicts a mass and the corresponding value of the decision variable is lower than $T_1$ as well as if it depicts a normal tissue and the corresponding value of the decision variable is higher than $T_2$. This way, nontypical examples are removed from the case base. In their experiments, the authors examine some thresholds such that $T_1 + T_2 = 1$. The second study on case base reduction for a CAD system was the one reported by Tourassi et al. (Tourassi et al., 2007b) for the same clinical problem as the previous study. This technique implements an entropy-based selection scheme where only examples with the highest entropy (i.e., highest information content) are preserved in the case base. The technique was proposed and investigated strictly within the context of IT-CAD system and as such it is not necessarily generalizable to feature-based CAD systems.

Choosing a case base selection algorithm for a particular clinical problem is not trivial. As discussed before, each of the previously proposed algorithms has its own advantages and disadvantages and many of them are limited to specific types of case representation. The aims of this study were to investigate algorithms previously proposed in machine learning that are suitable to a variety of CAD systems, adapt them for optimization based on clinically relevant objectives, and evaluate them with respect to evidence-based IT-CAD system for false positive reduction in screening mammograms. Specifically, three different intelligent techniques that fall into the second group of case selection algorithms described before are investigated. The criteria of choosing the techniques are discussed in more detail in the next section. Overall, the focus is on intelligent techniques that are not only well suited to feature-less CAD system but also easily generalizable to other types of evidence-based CAD systems.

B. Methods: case base optimization

To formalize the problem, it is assumed that assume that $T$ is an initial set of
available examples and $S$ is a subset of $T$. Then, given a desired number of examples in the final case base $k$, the problem is to find such $S^*$ that among all the subsets of $T$ containing $k$ examples, $S^*$ provides the best performance. Formally, find

$$S^* = \arg \max_{\{S: S \subseteq T, |S| = k\}} A(S)$$

where $A(S)$ is a measure of the classification accuracy of the system given $S$ as its case base. Note that the desired number $k$ is imposed by the hypothesized case base storage restrictions or by the response time requirement for efficient, real-time application of the system. Three different intelligent methods of case base reduction were selected. The selection criteria used were as follows:

- applicability to both feature-based and featureless case-based CAD systems, independently of their (dis)similarity measure and decision function,
- adaptability of the selection algorithm to optimize clinically relevant performance measures such as area under the ROC curve, partial ROC area, positive and negative predictive values etc.
- adaptability of the selection algorithm to specific storage limitations
- simplicity of implementation.

Consequently, the intelligent algorithms selected for comparative investigation in this study were: genetic algorithm selection (GAS), greedy selection (GREEDY), and random mutation hill climbing (RMHL). A random selection (RANDOM) technique was also applied as the default strategy to establish whether a more sophisticated strategy is indeed necessary.

All intelligent algorithms analyzed here are based on the same principle. In each algorithm the selected case base subsets of a given size are evaluated using the chosen figure of merit. Each algorithm, however, applies a different exploration technique to
find the most diagnostically useful case base subset. Although different metrics could be used as the figure of merit for performance evaluation, ROC based assessment was used. Namely, the AUC is chosen as the evaluation index since it is widely used with CAD systems. Specifically, the Wilcoxon approach (Bradley, 1997) was utilized to calculate the AUC. Thus, the problem was to find

$$S^* = \arg \max_{\{ S : S \subseteq T, |S| = k \}} \text{AUC}(S)$$

(11)

where AUC(T) is the area under the ROC curve for the system with the case base T.

More detailed description of the selected case base selection algorithms follows.

1. Genetic algorithm-based example selection

Genetic algorithm-based selection (GAS) is a technique utilizing evolutionary computation to find an optimal case base subset. Genetic algorithms have been applied to select a subset of features in feature-based systems as well as optimization of the case base (Cano et al., 2003; Llora and Garrell, 2003; Mazurowski et al., 2007). Initial investigations on the applicability of GAs to the case weighting (Mazurowski et al., 2008b) and case selection (Mazurowski et al., 2007) problem with IT-CAD system have been presented before for discrimination of true masses from normal breast parenchyma. Here, a more extensive study is presented by applying several case base reduction algorithms and by using a more elaborate data handling scheme as well as a new, more clinically challenging dataset.

In GAs, each candidate solution for a problem is coded in a chromosome of one individual. The algorithm starts with typically randomly generated population of individuals (typically 50-200). Then, the best solutions are evolved by means of crossover, mutation of genes, and natural selection. More details about the mechanics of the GA can be found elsewhere (Michalewicz, 1999; Eiben and Smith, 2003). The rest of this subsection is devoted to the description of solution representation and the
genetic operators used specifically in this study.

The diagram illustrating the progress of the genetic algorithm used in this study is shown in Figure 9. A candidate solution for the problem at hand is a \( k \)-element subset \( S \) of the original \( n_T \)-element set of available cases \( T \). To represent such solution in a chromosome (i.e., a sequence of numbers), all the cases in the original case base \( T \) are numbered. Given such numbering, the chromosome representing a subset of \( T \) is an \( n \)-element sequence containing '1' on the \( i \)-th position of the sequence if the \( i \)-th element belongs to \( S \) and '0' if it does not. Therefore, each chromosome contains exactly \( k \) ones and \( (n_T - k) \) zeros (Cano et al., 2003). To generate an offspring from the best adapted individuals, a one-point crossover recombination technique is utilized. Given two parents, a single point (called locus) is chosen randomly in the chromosome. Then, chromosomes of parents are split into two parts in the locus, generating two offspring individuals. The first offspring inherits the first part of the chromosome of the first parent and the second part of the chromosome of the second parent. The second offspring inherits the first part of the chromosome of the second parent and the second part of the chromosome of the first parent.

Such crossover, even though shown efficient in various applications, has a significant drawback for the application at hand. The problem at hand is a constrained optimization problem because only candidate solutions of a certain form (i.e., sets of given size \( k \)) are proper candidate solutions to the problem. Such constraint on the candidate solution incurs a constraint on the chromosome, namely a chromosome representing a proper solution must contain exactly \( k \) ones. The one-point crossover operator used in this study does not guarantee that when the parents satisfy the constraint, the offspring satisfies it as well. To deal with this issue, a repair function is applied to the offspring chromosomes. The repair function randomly changes ones to zeros if the number of ones is too large or zeros to ones if the number of ones is too small in order to obtain the proper number of zeros and ones and respectively a
subset with the required number of elements.

At every iteration of the GA algorithm, each chromosome is subject to random mutation as well. The mutation is performed by allowing each position in the chromosome to flip (i.e. change one to zero or zero to one) with a certain probability. Typically, the mutation probability is kept very small. Such operator, similar to the presented crossover operator, is "constraints-blind". Therefore, the chromosome repair function described in the previous paragraph is applied. Note that the repair function could be applied only once at the end of the mutation step. However, here the repair function is applied twice, at the end of the crossover and mutation steps separately, so that the two aspects of the GA algorithm are clearly delineated. Crossover and mutation operations are a fundamental aspect of GAs. The repair

Figure 9. Diagram illustrating steps of the genetic algorithm
function is just a modification to ensure that the solutions satisfy the constraints im-
posed by the specific problem. Actually, a repair function is one of the standard ways
of approaching constrained optimization problem and has been previously described
in the GA literature (Michalewicz, 1999; Eiben and Smith, 2003). Finally, to select
the parents and individuals to survive, proportional selection with windowing was
applied and a roulette rule was used to implement it (Eiben and Smith, 2003). As a
fitness function, simply the ROC area index $AUC(S)$ was used.

2. Greedy example selection

The greedy case selection algorithm (GREEDY) is an incremental algorithm which
chooses the best possible available solution at each step. The algorithm starts with
an empty subset $S$. In the first step, the algorithm chooses an example $E_i^*$ such
that $S_1$ containing only $E_i^*$ provides the highest $AUC(S_1)$. Then, in each subsequent
step $i$, the algorithm finds an example $E_i^*$ such that the set $S_i$ containing all the
cases selected in the previous steps and the example $E_i^*$ provides the best $AUC(S_i)$
among all possible selections during that step. The algorithm stops after $k$ steps
providing a subset of the desired size. Note that GREEDY is similar to forward
selection techniques (Blum and Langley, 1997). It is guaranteed to find a globally
optimal subset only for $k = 1$ since at any step previously selected cases cannot be
eliminated.

3. Random mutation hill climbing

Random mutation hill climbing (RMHC) was first applied to case base reduction
by Skalak (Skalak, 1994). The steps of this iterative technique are as follows. First, a
random subset $S$ of a desired size is selected. Then, in each iteration, one randomly
chosen element from $S$ is switched with one randomly chosen element from the re-
remaining cases \((T - S)\). If such change improves the objective (i.e., it strictly increases \(\text{AUC}(S)\)), it is accepted. Otherwise the change is reversed. The algorithm terminates when the maximum number of iterations is reached.

4. Random selection

The random selection algorithm randomly selects a subset of a given size without replacement and equal probability of selection of each example. As mentioned before, the RANDOM selection algorithm was implemented for comparison purposes to establish whether more sophisticated selection strategies are indeed superior.

C. Databases and experimental design

1. Databases

For this study, the Digital Database of Screening Mammography (Heath and et al., 1998) was utilized. The original mammograms were digitized using a Lumisys scanner to 12-bit images at 50\(\mu\)m per pixel. From the mammograms, 512 pixels \(\times\) 512 pixels regions of interest (ROIs) were extracted around mass and normal mammographic locations. Three separate databases of ROIs were created. One database was used for developmental purposes and proof of concept while the other two databases were used for additional validation. There was no overlap between the development database (Database 1) and the validation databases (Databases 2 and 3).

Database 1, used in the main part of the experimental evaluation consisted of 600 ROIs, 300 depicting biopsy-proven malignant and benign masses (positive class) and 300 depicting normal breast parenchyma (negative class). Mass ROIs were centered on the physician annotation provided in the DDSM truth files. The normal ROIs were regions indicated as suspicious by a prescreening CAD system developed at
Duke Advanced Imaging Laboratory (Catarius et al., 2004) to operate at 4 false positives per image on average. Note that these suspicious ROIs were considered normal according to the DDSM truth files and belonged to the negative class. Thus, the IT-CAD system was essentially tested as a second level analysis scheme for reduction of computer-generated false positives. To avoid any bias, efforts were made to ensure that this database did not include different views of the same mass. Therefore, all 300 mass ROIs corresponded to completely different masses. The same was true for all 300 normal ROIs.

Database 2 used for the final validation consisted of 200 ROIs, 101 depicting biopsy-proven masses and 99 depicting normal breast parenchyma. The ROIs were obtained in exactly the same way as the ROIs in Database 1. In other words, they did not include multiple views of the same mammographic locations.

Database 3, also used for additional validation of the conclusions, consisted of 98 ROIs, 58 depicting biopsy-proven masses and 40 depicting normal breast parenchyma. These ROIs were extracted around locations indicated as suspicious by a breast imaging specialist with more than 15 years of experience. The radiologist was blinded to the ground truth. The radiologist was asked to report any suspicious locations using a specially designed Graphical User Interface (GUI) that displayed one mammographic view at a time. 512x512 pixels ROIs were extracted around the reported locations. Database 3 was used to evaluate how the results obtained by the intelligent techniques translate to a slightly different clinical task; reduction of perceptually generated false positives.

2. Experimental design

In the main stage of the study, a ten fold cross-validation scheme was applied using Database 1 to assess effectiveness of each case selection method. The data was randomly split into ten folds. Nine folds (540 examples, 270 masses and 270 normals)
were used as the development set while the tenth fold (60 examples, 30 masses and 30 normals) was reserved as the test set. The same experiments were repeated 10 times so that each fold served for testing once. At each data split, the train and test sets included ROIs from completely different mammograms.

For each case selection method, the dataset folds were used in the following way. Given each data split the development dataset (i.e., the nine folds) was used as a pool of examples available for building the knowledge case base of the CAD system. In the example selection process, the leave-one-out approach was implemented within the development dataset as described in the methodology section. With the case base selection step complete, the IT-CAD system was applied using the resulting case base on the test set (i.e., the 10th fold reserved for testing only). This process was repeated ten times, till each fold served as a test set. The whole experiment was also repeated separately for various desired case base sizes $k$: 10-500 (see Eq. 11).

To assess the statistical significance of the CAD performance differences between the example selection methods and the original CAD (i.e., using the full case base), the ROCKIT software was used allowing for comparison of ROC areas for correlated data using a parametric estimation of the ROC curves (Metz et al., 1998a; Metz et al., 1998b).

For further validation of the conclusions additional experiments were conducted such that the entire Database 1 was used as the development dataset (utilized for construction of the case base as described in methodology section 2.1). Databases 2 and 3 were used only for testing. To account for the variability introduced by the relatively small size of these two databases, AUC performance was estimated using bootstrap sampling (Efron and Tibshirani, 1993).

D. Experimental results

In the experiments, the following implementation parameter values were employed.
For GAS, the number of chromosomes was set to 50, the number of offsprings was set to 50, and the probability of mutation was 0.0005. The maximum number of iterations for GAS was 100. For RMHC, a 2000 iteration limit was used. These parameters were optimized empirically to obtain the best performance while keeping the computation time reasonable. In the analysis of the experimental results two different design paradigms were taken into account, namely one where the CAD system designer imposes a constraint on the case base size (k) either because of a limited storage capacity or restricted response time, and another where the system designer aims for the best possible system performance regardless of the resulting case base size and its implications on computational efficiency. The underlying assumption of the first paradigm is that there may be superfluous cases present in the case base. The second paradigm is based on the assumption that some cases may have detrimental effect in the overall diagnostic performance of the CAD system. Both superfluous and detrimental cases could be eliminated, providing a system optimized in terms of diagnostic performance and computational efficiency. Results of the experiments based on Database 1 are presented separately for each design paradigm in Sections 1. and 2. Section 3. presents additional validation of the conclusions drawn using Database 1 on Databases 2 and 3.

1. Satisfying limited storage requirements

In this design scenario, a CAD system designer can only afford to store a limited number (k) of examples in the case base of the system. This number is essentially dictated by the system’s storage and computational power limitations. Figure 10 compares the IT-CAD system’s performance obtained by each case base selection method for various desired sizes k of the case base. Performance is shown in terms of the average AUC(S) obtained by a particular example selection method across all cross-validation folds. The horizontal axis of the graph shows the desired size k of
the case base subset (as defined in Eq. 11).

Overall, the three intelligent methods outperform the random selection in a wide range of desired database sizes. The difference becomes more dramatic as the allowable case base size is further restricted. This finding suggests that if only a very small number of cases can be stored in the case base, then intelligent selection of cases is critical to ensure optimized performance. At the upper limit (i.e., using all available examples), all techniques provide the same results (0.745±0.020), since there is no reduction of the case base. This performance is shown in Figure 10 (all lines converge to the same point for \( k = 540 \)). For the problem at hand, the RMHC algorithm provides consistently the best results. Average RMHC results were better than random selection for all examined desired database sizes. Statistical comparison of the methods was performed for three representative case base sizes of 20, 100 and 200.
Table 2. Two-tail p-values for the AUC pairwise comparison of techniques for three representative case base sizes in Database 1

<table>
<thead>
<tr>
<th>Case base size (k)</th>
<th>20</th>
<th>100</th>
<th>200</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMHC vs. GREEDY</td>
<td>0.6698</td>
<td>0.0088</td>
<td>0.7065</td>
</tr>
<tr>
<td>RMHC vs. GAS</td>
<td>0.5848</td>
<td>0.025</td>
<td>0.0167</td>
</tr>
<tr>
<td>GREEDY vs. GAS</td>
<td>0.2568</td>
<td>0.7932</td>
<td>0.0249</td>
</tr>
<tr>
<td>RMHC vs. RANDOM</td>
<td>&lt;0.0001</td>
<td>0.0004</td>
<td>0.0003</td>
</tr>
<tr>
<td>GREEDY vs. RANDOM</td>
<td>&lt;0.0001</td>
<td>0.0655</td>
<td>0.0004</td>
</tr>
<tr>
<td>GAS vs. RANDOM</td>
<td>&lt;0.0001</td>
<td>0.1178</td>
<td>0.0478</td>
</tr>
</tbody>
</table>

examples. Table 2 shows the two-tailed p-values for pair-wise comparisons of the obtained AUC by different selection methods. These p-values were obtained using the ROCKIT software (Metz et al., 1998a; Metz et al., 1998b) after merging the IT-CAD prediction responses across all cross-validation folds.

Table 2 shows that for very small case base sizes (i.e., 20 examples) all intelligent techniques statistically significantly outperformed random selection (p<0.0001). For such low \( k \) the difference in performance between the intelligent techniques and random selection is the most apparent. However, no statistically significant differences are observed among the intelligent techniques for such low \( k \) values. For larger case base sizes (i.e., 100 examples), RMHC statistically significantly outperforms all other techniques. At this size, even though the difference between the other two intelligent techniques (GREEDY and GA) and RANDOM were substantial, they did not reach statistical significance. When the allowable case base size increases (i.e., 200 examples), no statistically significant difference is observed between RMHC and GREEDY but both techniques significantly outperform GAS and RANDOM. Although the performance obtained for this case using GAS is lower than for other intelligent methods,
Figure 11. The most frequently selected examples by RMHC in Database 1 when the resulting case base size was 20.

GAS is still statistically significantly better than RANDOM.

An interesting finding of this study is that with as few as 10 or 20 intelligently selected cases, the IT-CAD system achieves comparable performance to that when relying on the full case base. This finding may seem inconsistent with general wisdom that a large case base is essential in clinical applications. However, such dramatic reduction has been previously shown the in machine learning field (Wilson and Martinez, 2000; Skalak, 1994; Pekalska et al., 2006). Furthermore, it must be noted that most CAD studies assessing the impact of the number of training examples were based on the assumption that the examples are drawn from their populations randomly. This study approaches the problem from a different perspective and shows that application of intelligent techniques can identify a set of only few examples that are critical for the system to maintain its diagnostic capability at the same level as when using the entire case base.

In addition, the cases that were more frequently selected in the early stages of the
case selection process \((k = 20)\) were examined. Figure 11 presents the seven most frequently selected examples (ROIs) when the case base was optimized with RMHC and the resulting case base contained 20 examples. Note that for the applied data handling scheme, each example was in the pool of cases nine times (i.e., nine possible folds). The ROIs presented in Figure 11 were selected at least four times. These are ROIs that were consistently selected as the most useful diagnostically. It is interesting that five out of the seven examples represent masses. Furthermore, the overwhelming majority are malignant masses. The shape, size, and margin characteristics are quite diverse and span a wide range. Specifically, Figures 11a and 11c show lobulated masses with ill-defined and microlobulated margins respectively. Figure 11d depicts an irregular mass with spiculated margins and associated architectural distortion. Figures 11e and 11f show round masses with circumscribed and microlobulated margins respectively. Interestingly, the mass shown in Figure 11e turn out to be malignant while the one shown in Figure 11f was the only benign one. Finally, the two most frequently selected false positive examples are also rather different. Overall, Figure 11 shows that given a requirement of very small resulting case base, the intelligent techniques select a diverse subset of examples with wide range of characteristics. It must be also noted that the performance for low \(k\), even though comparable to the performance of the system with full case base, is lower than the best performance that can be achieved by the system. In fact, CB-CAD performance can be further improved by expanding the variety of examples using one of the intelligent selection techniques.

Also note that since no rule for equal prevalence is imposed, it is not necessary that the reduced set will contain an equal number of mass and normal examples. It is likely that not imposing a constraint on class prevalence is more consistent with the clinical problem where the CAD designer does not know a priori the number of examples from each class that may be detrimental or which class will contain more
Table 3. Best performance obtained by the system developed with different case base selection algorithms. For each algorithm, the numbers in brackets indicate the number of cases for which the best performance was obtained.

<table>
<thead>
<tr>
<th></th>
<th>Original CAD</th>
<th>RMHC (200)</th>
<th>GREEDY (300)</th>
<th>GAS (50)</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUC</td>
<td>0.745 ± 0.020</td>
<td>0.789 ± 0.018</td>
<td>0.787 ± 0.018</td>
<td>0.760 ± 0.019</td>
</tr>
<tr>
<td>0.9AUC</td>
<td>0.259 ± 0.033</td>
<td>0.262 ± 0.037</td>
<td>0.274 ± 0.037</td>
<td>0.213 ± 0.034</td>
</tr>
</tbody>
</table>

clinically useful examples. However, all three algorithms can be easily modified to impose an equal number of mass and normal examples in the selected case base (or any other ratio for that matter).

To address further the issue of imbalance in the selected subsets, various subsets of intelligently selected examples were examined more carefully. It was apparent that even though typically for selected case bases of very small sizes the number of positive examples exceeds the number of negative examples, as the number of desired examples $k$ increases, the ratio of mass to normal examples gets closer to one.

2. Maximizing diagnostic performance

In this scenario no limitations are imposed on the resulting case base size. The main goal is to select examples that provide the best possible CAD performance. Table 3 shows the maximal performance in terms of AUC obtained by the system developed with different case base selection algorithm. Figure 12 and 13 shows the corresponding ROC curves (overall and partial for high sensitivities respectively). The best overall performance (AUC = 0.789 ± 0.018) was obtained by RMHC for 200 cases. The same performance level (AUC=0.787 ± 0.018) was reached by GREEDY for 300 examples. A comparison of the ROC performance obtained by these two methods to the performance of the original CAD system indicates statistically significant improvement in both cases (two-tailed p-value<0.005). However, the improvement obtained by
Figure 12. ROC curves for best IT-CAD performance obtained with each case base selection method.

GAS (AUC=0.760 ± 0.019) did not reach statistical significance (p=0.3). Analysis of the performance measured by 0.9AUC shows small but not statistically significant (p=0.9) improvement obtained by applying RMHC or GREEDY. Actually, applying GAS statistically significantly decreased the 0.9AUC index of the system (p<0.05).

3. Additional validation

To provide further validation of the observations on Database 1, an experiment was performed using the entire Database 1 as a development dataset and using Databases 2 and 3 for testing. This additional validation simulates an actual clinical scenario where the case base is built using the preferred intelligent case selection strategy and then the case-based CAD system is put to practice. Since RMHC emerged as
the intelligent selection technique that provided the best and most robust results in Database 1, these additional validation studies were performed using only RMHC.

The baseline performance of IT-CAD using the entire Database 1 as a case base and tested on Database 2 was AUC=0.748±0.034 (estimate based on 5000 bootstrap samples). To account for the variability due to the stochastic nature of the RMHC and RANDOM methods, the selection was repeated 50 times for each \( k \). Applying RMHC resulted in an improvement for all three case base sizes explored in this study (median values for 50 RMHC runs are given): AUC=0.812±0.030 for \( k = 20 \), AUC=0.778±0.033 for \( k = 100 \), and 0.767±0.033 for \( k = 200 \). It is notable that in Database 2, the improvement in AUC performance for \( k = 20 \) was considerably higher than the baseline performance. However, this could be attributed to the spec-
specific database. The variability introduced by the stochastic nature of RMHC, as expressed by the standard deviation of the performance in the 50 runs, was relatively low: 5.5%, 4.9% and 4.2% of the performance value for 20, 100 and 200 desired case base sizes respectively. This variability decreases with increasing $k$. Random selection resulted in median performance of AUC=0.659±0.039 for $k = 20$, AUC=0.718±0.036 for $k = 100$ and AUC=0.739±0.036 for $k = 200$. The variability introduced by RANDOM was much higher than the variability associated with RMHC and was 16.3% for $k = 20$, 7.3% for $k = 100$ and 4.3% for $k = 200$. Overall, the validation experiment on Database 2 confirms the previous conclusions: (i) intelligent selection techniques allow for a significant reduction of the case base while sustaining or improving the diagnostic performance of the CAD system and (ii) the intelligent techniques outperform the random selection especially when small case base size is desired.

Testing on the Database 3 showed a baseline performance of AUC=0.562±0.061. This is a significantly lower performance than what observed in Databases 1 and 2. However, this is not unexpected. Databases 1 and 2 are more homogeneous since both contain computer-generated false positives. Database 3 contains false positives generated by a highly expert radiologist, thus representing a particularly challenging set. The performance of the system after applying RMHC was improved to AUC=0.654±0.059 for $k = 20$, AUC=0.680±0.057 for $k = 100$, and AUC=0.681±0.056 for $k = 200$. The variability of AUC over 50 RMHC runs was 5.0% for $k = 20$, 3.2% for $k = 100$ and 2.0% for $k = 200$. The performance for RANDOM was AUC=0.554±0.062 for $k = 20$, AUC=0.544±0.062 for $k = 100$ and AUC=0.564±0.060 for $k = 200$. The AUC variability with RANDOM over 50 runs (10.5% for $k = 20$, 6.8% for $k = 100$ and 6.5% for $k = 200$) was again larger than the variability with RMHC selection. Note that again, as expected, the variability introduced by the selection method (for both RMHC and RANDOM) decreases with increasing $k$. These results confirm that case base optimization translates very well
to a database that is more loosely related to the one used for developing the case base. The results obtained on Database 3 support further the hypothesis that a substantial improvement in classification performance of the IT-CAD can be obtained by applying RMHC while at the same time reducing the case base.

4. Comparison to previously reported case base reduction techniques in CAD

The presented algorithms were also compared to already proposed ones in the CAD field (Park et al., 2007; Tourassi et al., 2007b). To compare the techniques presented here to the one recently proposed by Park et al. (Park et al., 2007) the latter was implemented with parameter values examined by the authors, i.e. thresholds varying from 0.05 to 0.35 for true positive ROIs and from 0.65 to 0.95 for false positive ROIs. For all examined thresholds the obtained AUC was lower than that of the original system ((AUC=0.745±0.020). However, a small reduction of the database by 15% was observed with non-significant drop of performance (AUC=0.720±0.020).

Comparison of the entropy-based selection technique proposed by Tourassi et al. (Tourassi et al., 2007b) using the Database 1 resulted in small improvement of overall performance (AUC improved from 0.745±0.020 to 0.752±0.019). Applying this technique allowed for case base size reduction by 26% (to 400 examples) without compromising the performance of the system.

E. Discussion

This part of the dissertation research focused on the problem of building a knowledge database for case-based medical decision support systems. The significance of the problem was discussed and three intelligent techniques were evaluated experimentally for solving the problem. Although the study was performed with respect to an information-theoretic CAD system, the proposed techniques are applicable to virtu-
ally all case-based CAD systems, irrespective of the type of case representation (i.e., feature-based or featureless), case (dis)similarity functions (i.e., metric or non-metric) and/or decision algorithms they employ.

The experimental results show high efficiency of the examined techniques for the IT-CAD system with a significant advantage of the intelligent techniques over random selection. The studied techniques can be applied in various ways depending on the ultimate optimization goal. If the main goal is reducing the available case base with no loss of diagnostic performance or with limited loss, intelligent techniques such as the random mutation hill climbing algorithm are very promising. For this study, all intelligent techniques were able to build a concise database of 10-20 cases (less than 4% of its original size) without compromising the overall performance of the system. This finding may seem controversial, however it is consistent with evidence provided in machine learning with a variety of benchmark databases. For example, Wilson et al. (Wilson and Martinez, 2000) showed that the case-base can be reduced to less than 1% of the original database without loss of performance. Skalak (Skalak, 1994) showed on some benchmark problems that for the k-nearest neighbor (k-NN) rule, the number of examples can be reduced to as few as 1% of the original database. Recently Pekalska et al. (Pekalska et al., 2006) demonstrated that the database of examples in case-based systems can be in some cases reduced to about 20 examples when using k-nearest neighbor classification rule without decrease of performance. The study by Pekalska et al. is of particular relevance to this study as it demonstrates a similar finding using a non-metric, dissimilarity based classifier. The same authors also show that when more sophisticated classification rules are applied the database can be reduced to as few as three examples without compromising performance of the system.

This result may appear inconsistent with findings of previous studies which demonstrated that a large and diverse database is needed to develop a successful CAD clas-
sifier. Note, however, that these studies assume that the examples available for the
development of the CAD system are drawn randomly from the available population.
In fact, the results presented here concur with these previous findings by showing
that as more examples are selected randomly from the available pool, the CAD per­
formance increases consistently and that random selection of just a few examples is
insufficient, resulting in a significant decrease of performance (Figure 10). The in­
vestigations of this study extend these previous results by showing that given a large
collection of available examples, sophisticated rather than random case base selection
results in a significantly smaller case base with similar or better predictive power.
Nevertheless, the amount of case base reduction depends on the size and diversity of
the original case base as well as the complexity of the decision problem at hand.

The studied techniques can be also applied when the main goal is to maximize
diagnostic performance regardless of the case base size. In such scenario, applying
the intelligent techniques such as RMHC or GREEDY selection are also well justified.
For this study, both techniques resulted in statistically significant improvement of
the performance while reducing the database size to just 200 or 300 examples (37%
and 56% of the original database respectively). This result suggests that system
designers need to keep in mind that some knowledge examples may have detrimental
effect on the overall performance of the system. This finding was also confirmed by
other investigators (Park et al., 2007). Careful data mining is certainly warranted
to determine when and why some examples may have detrimental effect but such
analysis is beyond the scope of this manuscript. Overall, this comparative study
showed that random mutation hill climbing is the most effective technique for case
selection.

Additionally, a comparison was performed between the presented intelligent algo­
rithms and ones that have been previously reported in the CAD field and it was
shown that the intelligent techniques investigated in this chapter are superior for the
task at hand. The technique proposed by Park et al. was less effective with respect to IT-CAD system. It is likely that this is mainly because the technique by Park et al is tailored to metric-based similarity measures. IT-CAD system does not satisfy this condition due to the non-metric nature of the NMI similarity measure. Similarly, the superior performance of the intelligent techniques implemented in this chapter over the entropy-based selection technique can be attributed to the fact that the CAD system is always part of the case selection process during the intelligent selection which allows for tailoring the selected case base to the system. The entropy-based selection method operates independently of the system. Also it must be noted that the entropy-based selection technique is specifically applicable to IT-CAD system and not necessarily effective with other evidence-based CAD schemes.

Comparing the computational complexity of the intelligent algorithms is difficult as they highly depend on the algorithm parameters (e.g., number of chromosomes, iterations, etc.). In this study, however, the RMHC was roughly 10-folds faster than the GAS. This result reinforces even further that RMHC is truly the superior technique for the task at hand, not only in terms of improving system performance but also in terms of time complexity. Although more careful optimization of the GAS may have led to better performance, this is not a trivial issue from computational point of view. RMHC as well as GREEDY are considerably simpler to implement compared to the GA-based selection technique.

While the time complexity and implementation time of the algorithms are independent of the database, system performance obtained with a particular algorithm varies depending on the clinical application. For example, the experimentation with the GREEDY selection algorithm suggests was that it is particularly sensitive to overfitting when only a small pool of examples is provided to develop a CAD system. Consequently, CAD designer should choose carefully the reduction method considering the specifics of the particular problem. Ideally, designers should compare various
different algorithms before finalizing the case base selection process.

In conclusion, this study presented a comparative analysis of three intelligent techniques for case base optimization in evidence-based CAD systems. Although the analysis was based on a specific CAD system and clinical task, the techniques are applicable to a wide variety of case-based CAD systems regardless of their case representation, similarity measure and/or decision making algorithms. The study demonstrated clearly the advantage of intelligent case base optimization over conventional random selection. Furthermore, random mutation hill climbing emerged as the superior choice. It not only improved markedly the efficiency of IT-CAD system but it also improved significantly its diagnostic performance.
CHAPTER IV
DECISION AND CASE BASE OPTIMIZATION USING ENSEMBLE TECHNIQUES*

A. Introduction

Traditionally, when approaching two class classification problem, a single classifier such as linear discriminant analysis (LDA) (Duda et al., 2000), artificial neural network (ANN) (Zurada, 1992), or k-nearest neighbor classifier (k-NN) (Mitchell, 1997) is used for the task. Such classifiers use all previously acquired examples (clinical cases) to develop the system, typically through training. This approach has been shown to be quite effective and robust. However, it has disadvantages. A single classifier can be susceptible to initialization and training parameters and it is always dependent on the particular training dataset that may or may not properly reflect characteristics of the underlying population. Overtraining can be a significant detrimental factor when the training parameters are not carefully optimized. In the context of these difficulties, ensemble classifiers (also called multi-classifier systems) are becoming a sensible and increasingly popular choice (Kuncheva, 2004). They are known to offer a more efficient use of the available data and, as a result, improved performance. Ensemble techniques have been examined extensively in the machine learning literature (Kittler et al., 1998; Kuncheva, 2004; Ranawana and Palade, 2006) and they have been shown effective in multiple applications (Woods et al., 1997; Kuncheva et al., 2001; Kuncheva, 2002). Applications of ensemble techniques in computer-aided diagnosis

* This chapter, with some modifications, has been included in (Mazurowski et al., 2009). The preliminary results were published in (Mazurowski et al., 2008c).
The underlying idea of ensemble classifiers is to construct multiple sub-classifiers and then develop a combiner that summarizes the predictions provided by all the sub-classifiers into one final decision. Various approaches have been proposed for both, sub-classifier construction and sub-classifier combination (Kuncheva, 2004). In order to construct multiple sub-classifiers one can (i) assign different training/development parameters to each sub-classifier (ii) use different types of sub-classifiers or (iii) use different subsets of the development dataset for each sub-classifier. When the sub-classifiers are created, a combiner must be designed that uses predictions of all the sub-classifiers to make a final decision. Two general ways of combining sub-classifiers are (i) classifier fusion and (ii) classifier selection. In classifier fusion (also called cooperative approach) all the predictions are merged into one final decision. In classifier selection (also called competitive approach), the best/most appropriate sub-classifier is selected and its prediction is used as the final decision.

Even though ensemble classifiers are often more effective than single classifiers, they impose certain challenges such as selecting multiple parameters needed to construct an effective ensemble. While researchers typically concentrate their attention on various ways of constructing and combining classifiers, the issue of ensemble size is mostly left out. Only a few studies tackle the topic (for an overview see (Kuncheva, 2004)). These studies typically propose an "overproduce and select" method where an initial large set of sub-classifiers is constructed and then it is reduced using various methods. Giacinto and Roli (Giacinto and Roli, 2001b; Giacinto and Roli, 2001a), for example, use clustering of the constructed neural network-based sub-classifiers and select the optimal subset based on their diversity and performance. Margineantu and
Dietterich (Margineantu and Dietterich, 1997) use Kappa statistic to prune the Adaptive Boosting ensemble to a prespecified size. A similar approach called "Thinning the ensemble" can be found in the study by Banfield et al (Banfield et al., 2003). All these techniques start with a set of sub-classifiers and select a subset to construct the final ensemble. Note that the size of the initial ensemble needs to be specified by the designer. Furthermore, all of these techniques make use of the concept of diversity to make a selection. In this chapter adaptive incremental techniques are proposed that start with a single classifier and then gradually add sub-classifiers based on the overall performance of the resulting system. The proposed techniques do not require the designer to specify the size of ensemble but determine it automatically.

The focus of this study is on building ensembles for case-based classifiers (Aha et al., 1991). Instead of extracting rules from the available training data, case-based classifiers store the actual examples in the database of the system (called case base). When a new unknown query arrives, its similarity to the case base examples is assessed and based on these similarities a decision regarding the query is made. Therefore, case base size in the system directly impacts the system’s storage requirements and response time. These two issues are of particular importance in the information-theoretic CAD (IT-CAD) (Tourassi et al., 2003b) system employed in this study. IT-CAD stores the previously acquired examples as entire images (large storage requirement) and uses a computationally expensive mutual information index as a measure of similarity between examples (Tourassi et al., 2007a).

In most studies, the only purpose of using ensembles is to improve the classification performance of the system. However, if the sub-classifiers are case-based, the total number of examples used in the system may be a concern since it determines the response time and storage requirement of the system. In this chapter, the possibility is explored of using ensemble techniques to reduce the total number of examples stored in IT-CAD while at the same time improving the system's performance.
Figure 14. A diagram showing the general idea of building an ensemble with a case-based system

B. Methods: constructing the ensemble

As previously described, the process of constructing an ensemble consists of two primary steps: constructing sub-classifiers and constructing a combiner. The process is depicted in Figure 14. This section describes in detail both steps and is concluded by a description of the proposed adaptive incremental way of building ensembles.
1. Constructing sub-classifiers

Since a featureless case-based classifier is used, there are no natural parameters (e.g., number of features) that can be varied and there is no training involved in the process of constructing IT-CAD. Thus, the focus is on resampling the development dataset as a technique for constructing sub-classifiers. Therefore, each sub-classifier in this study is simply an IT-CAD based on a certain subset of the development dataset. In this study two different ways of creating these subsets are compared.

1.1 Random division.

In this approach, the development dataset is divided randomly into $L$ mutually exclusive subsets. The sum of such subsets is equal to the original development dataset. The subsets have the same (or approximately the same if the total number of examples does not allow for an equal division) size. Note that in this approach the total number of examples used in the resulting ensemble is the same as in the original classifier.

1.2 Random selection.

In random selection, for each sub-classifier, a set of $N$ examples is selected randomly from the development dataset. The selected subsets can overlap. The maximum total number of distinct examples stored in the system for this approach is $L \cdot N$ ($L$ is a number of classifiers). Note that this is an upper bound for the total number of examples and the actual number can be lower due to the overlap in the selected subsets. In this approach the total number of distinct examples stored in the ensemble can be significantly lower than in the development dataset available for the original classifier and thus, random selection could offer a case base reduction benefit.
2. Combining sub-classifiers

To combine decisions of sub-classifiers, fusion based on linear discriminant analysis was used in the following way. LDA is chosen since it is simple, effective, and popular particularly in medical decision support research. LDA fusion is implemented as follows. Each example $x^{(i)}$ is presented to each sub-classifier in the ensemble $D = \{D_1, D_2, ..., D_L\}$ resulting in a vector of decision values $(d_1^{(i)}, d_2^{(i)}, ..., d_L^{(i)})$. Such vector, together with the ground truth information for the example $f(x^{(i)})$ is used as a training example for the second level LDA combiner. Therefore the number of examples used for training the LDA is always equal to the number of examples in the entire development dataset. When $x^{(i)}$ is included in the case base of one or more sub-classifiers, this example is temporarily removed from such case base while calculating the decision to avoid a bias.

3. An incremental strategy

In most of the previously proposed ensemble techniques the classifier designer must decide a priori the number of sub-classifiers. Too few sub-classifiers can result in a poor performance because of lack of diversity in the ensemble (Kuncheva, 2004). In contrast, too many sub-classifiers can pose problems as well. When a trainable second-level classifier is used as the combiner, the number of inputs in the combiner is equal to the number of sub-classifiers. It is well known that a high ratio of number of inputs to number of examples can deteriorate the classifier performance due to overtraining (i.e., "curse of dimensionality") (Duda et al., 2000). Furthermore, in the random selection approach, more sub-classifiers mean higher time complexity of the classification algorithm when classifying new, incoming queries.

In this chapter an adaptive incremental strategy of building ensembles is proposed that utilizes clinically relevant performance measures as optimization criteria. The
The two adaptive methods use the same following algorithm of evaluating the candidate ensemble performance. Given the ensemble $D = \{D_1, D_2, ..., D_L\}$ the development dataset is divided into an internal training (90% examples) and internal test (10% examples) sets. Then, an LDA is constructed in the way described in section 2.3 using the internal training set and tested on the internal test set. Various methods of evaluating performance could be used at this stage. In this study the receiver operator characteristic (ROC) analysis and the area under the ROC curve (AUC) (Bradley, 1997; Fawcett, 2006; Obuchowski, 2003) are used as they are clinically relevant. The process is repeated ten times implementing a 10-fold crossvalidation scheme. The average internal test AUC performance is used for performance evaluation of the ensemble $D$. It will be denoted as $AUC(D)$. This performance measure is assumed to approximate the generalizing abilities of the ensemble. Given such performance measure, three incremental ways of constructing the ensemble are evaluated.

3.1 Building ensemble with 'no-control' method.

In the first method (it will be called 'no-control') the process of constructing an ensemble is initiated with a single sub-classifier based on a randomly selected set of examples. Then, another sub-classifier, also based on a randomly selected set of examples, is added. The sub-classifiers do not have to satisfy any criteria to be added. This process is stopped by the system designer by specifying the number of sub-classifiers to be included in the ensemble. Therefore, this method is nonadaptive.

3.2 Building ensemble with 'add-if-better' method

In the second method (it will be called 'add-if-better') the ensemble is initiated as an empty set $D = \emptyset$. Then at each step $k$, one sub-classifier is added to the
Figure 15. A diagram showing different kinds of ensemble techniques used in this study.

ensemble. The new sub-classifier becomes an element of the ensemble if it improves the performance of the ensemble \( AUC(D(k)) \) by a certain margin. Formally, the ensemble is extended by adding the new sub-classifier if it satisfies

\[
AUC(D(k)) > AUC(D(k - 1)) + 0.001. \tag{12}
\]

where \( D(k) \) is the candidate ensemble in step \( k \). The margin of 0.001 minimal improvement in performance is added to avoid unnecessary expansion of the ensemble with only marginal improvement in the performance. If the new sub-classifier does not satisfy this condition, it is not included in the ensemble. In such case another candidate sub-classifier is evaluated. This process is repeated up to 50 times in each step. If none of the 50 evaluated candidate sub-classifiers satisfy the condition, the process of ensemble building is terminated.

3.3 Building ensemble with 'pick-best' method.

In the third method (it will be called 'pick-best') the ensemble is also initiated as an empty set \( D = \emptyset \). Then in each step, 50 candidate sub-classifiers are constructed and the sub-classifier that, when added to the ensemble, provides the best improvement in the ensemble performance is included in the ensemble. The process is terminated when none of the 50 sub-classifiers satisfies condition (12).

Figure 15 presents all the ensemble techniques used in this chapter in a diagram.
Note that the proposed adaptive incremental approach and thus the two methods presented above ('add-if-better' and 'pick-best') are applicable only to the random selection approach. In random division as well as in random selection with 'no-control' the number of sub-classifiers must be determined a priori.

C. Databases and experimental design

1. Databases

In order to evaluate experimentally the performance of the ensemble techniques, the Digital Database of Screening Mammography (Heath and et al., 1998) was used. Lumisys scans of the original mammograms to 12-bit images at 50μm per pixel were used. From these images a set of 512x512 pixels regions of interest (ROI) was extracted. A total of 1820 ROIs were used. Nine hundred and one mass ROIs (489 malignant and 412 benign masses) were extracted based on the physicians annotations. Nine hundred and nineteen normal ROIs were extracted randomly by sampling the breast region of normal mammograms (without overlap). The task of the examined CAD system is to distinguish between ROIs depicting masses and those depicting normal tissue.

Database 1 consisted of 1500 randomly selected ROIs. Of them 738 depicted masses and 762 depicted normal tissue. Database 1 was used in the main part of the experimental evaluation. Database 2 consisted of the remaining 230 ROIs. 163 of them depicted masses and 157 depicted normal tissue. Database 2 was used for additional validation.
2. Experimental design

In the experiments the performance of all proposed techniques was evaluated and compared to the performance of the original IT-CAD system. The focus was on three factors that affect the ensemble's performance: the way of creating sub-classifiers, number of sub-classifiers and (in case of random selection) number of examples in each sub-classifier.

For the random division approach the performance of the system with the number of sub-classifiers varying from two to 500 was evaluated. The number of examples in each sub-classifier strictly depended on the number of sub-classifiers (\(N = N_{\text{dev}}/L\) where \(N_{\text{dev}}\) is the total number of examples in development dataset).

For the random selection approach the ensemble performance for \(N = 2, 10, 20, 50, 100, 200\) was evaluated. As described in the methodology section three methods for incremental construction of ensembles were used: 'no-control' (nonadaptive), 'add-if-better' (adaptive) and 'pick-best' (adaptive). For the 'no-control' method, the performance of the system was evaluated in a wide range of \(L\). For the other two methods, the development process was simply executed and the performance of the system as well as the number of sub-classifiers after the process was terminated was recorded.

To ensure an accurate estimation of the system performance, a hold-out data handling scheme was used in the following way. Database 1 was divided randomly into a development dataset (90\% of Database 1, 1350 examples) and a testing set (10\% of Database 1, 150 examples). The entire process of building an ensemble was executed using the development set. The testing set was used only for the final validation. The process was repeated 200 times. The average test performance and its variability over these 200 splits/runs is reported. To compare the performance of ensemble to the performance of the original system, a paired t-test based on these 200 splits was used.

As an additional validation, the entire Database 1 was used as a development
dataset and Database 2 was used for testing. The development and testing was repeated only once for all the techniques and examined parameters. This emulates a real-life system development and testing scenario. In all experiments, for consistency, the Wilcoxon nonparametric estimation of the area under the ROC curve was used.

D. Experimental results

The main results are presented in two subsections. In the first subsection the focus is on the scenario when the system developer's only goal is to maximize system performance. In the second subsection, the focus is on reducing the total number of examples stored in the system. In the following two subsections (Sections 3 and 4) additional validation of the results is performed and a closer look is given to the process of building ensembles with the two proposed adaptive methods.

1. Improving performance

The average AUC test performance (for 200 crossvalidation splits) for the examined techniques is presented in Table 4 with respect to the number of examples \( N \) in each sub-classifier. In this context \( N \) is an algorithm parameter that has to be decided by the system designer. For the non-adaptive methods (random division and random selection with 'no-control') reported the maximum test average performance obtained when varying ensemble sizes \( L \) is reported. For the adaptive methods (random selection with 'add-if-better' and 'pick-best') simply the average performance obtained for the method is reported. The performance of the original, single IT-CAD classifier system was 0.865±0.029. It was established that all the ensemble techniques examined in this paper provided a similar and statistically significant \((p < 0.001)\) improvement of performance as compared to the original IT-CAD. Furthermore, it was observed that for random selection approach performance improvement was independent of the
Table 4. Maximum AUC test performance for the examined ensemble techniques

<table>
<thead>
<tr>
<th>N</th>
<th>random selection</th>
<th>random division</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>'no-control'</td>
<td>'add-if-better'</td>
</tr>
<tr>
<td>2</td>
<td>0.903±0.024</td>
<td>0.902±0.023</td>
</tr>
<tr>
<td>5</td>
<td>0.903±0.023</td>
<td>0.901±0.024</td>
</tr>
<tr>
<td>10</td>
<td>0.903±0.024</td>
<td>0.901±0.024</td>
</tr>
<tr>
<td>20</td>
<td>0.904±0.024</td>
<td>0.900±0.025</td>
</tr>
<tr>
<td>50</td>
<td>0.904±0.024</td>
<td>0.902±0.024</td>
</tr>
<tr>
<td>100</td>
<td>0.904±0.024</td>
<td>0.902±0.023</td>
</tr>
<tr>
<td>200</td>
<td>0.903±0.023</td>
<td>0.901±0.023</td>
</tr>
</tbody>
</table>

number of examples N included in each sub-classifier.

2. Reducing the total number of examples stored in the system

Figure 16 presents the relationship between the number of sub-classifiers L and the average test ROC performance for Database 1. For the techniques where the number of sub-classifiers L is determined a priori by the designer, this relationship is represented by a curve to show how the performance depends on the choice of L. For the techniques where L is determined automatically by the algorithm, it is represented by a single point (a square or a circle). The solid curves show the performance for the random selection approach with 'no-control' for N = 2, 20, 200. Only these three sub-classifier sizes were selected to keep the graph simpler since the results for N = 5, 10, 50, 100 followed the same trends. The dotted curve shows the performance of random division. The performance of random selection with 'add-if-better' and 'pick-best' is represented by squares and circles respectively.

Several conclusions can be drawn from Figure 16. When the designer selects the
Figure 16. A relation between AUC performance of the system and number of subclassifiers
number of sub-classifiers \( L \) (i.e. random selection with 'no-control' and random division) the performance is clearly dependent on \( L \). Specifically, the AUC performance index initially increases and then, after reaching its maximum, it deteriorates with increasing \( L \). The initial improvement of the performance \((L = 1 \text{ to } L \approx 100)\) can be explained by the fact that the response of each additional sub-classifier to a query can be treated as an additional feature of that query and thus can be useful in the decision process and in turn improve the overall performance. On the other hand, the drop in performance for \( L > 150 \) can be explained by the fact that adding sub-classifiers corresponds to adding inputs to a second level combiner which given a limited number of training examples causes overtraining. The number of examples in each sub-classifier \( N \) had no impact on susceptibility of the ensemble to overtraining. The classifier designer needs to remember that the value of \( L \) for which the performance reaches its maximum may depend on the number of examples available for the development dataset and needs to be determined with a considerable experimental effort. This calls for reliable algorithms to select the size of ensembles automatically such as the two adaptive methods proposed in this chapter (random selection with 'add-if-better' and 'pick-best').

As the next step the described techniques were examined based on if and how their performance depends on the total number of distinct examples used in the ensemble. This issue is important not only to select ensemble parameters providing the best achievable performance but also in a scenario when the total number of examples used in a system is a concern. For random division approach the total number of distinct examples is always equal to the number of examples in the original development dataset. Therefore only random selection approach can potentially offer a reduction of the case base used in the system. Furthermore, note that in random selection the total number of distinct examples used in the system can be lower than the product of number of sub-classifiers and number of examples in each sub-classifier \((L \times N)\).
This is the case because the same example can occur in more than one sub-classifier. Therefore, \( L \times N \) constitutes an upper bound for the total number of distinct examples stored in the system.

The total number of distinct examples used in the ensemble is presented in Table 5. For the methods where the number of sub-classifiers is determined automatically by the method, the total number of distinct examples is averaged over 200 splits. For the method where \( L \) is decided a priori by the system designer ('no-control'), the number of distinct examples is presented for \( L \) that provided maximum average test performance (i.e. an optimal choice of the ensemble size). Since the total number of distinct examples for a given number of sub-classifiers \( L \) may slightly vary, the number of examples averaged over 200 splits is provided in the table.

For the 'no-control' method, the resulting total number of distinct examples highly depends on the number of examples \( N \) in each sub-classifier. Specifically, increasing \( N \) resulted in increasing total number of distinct examples. This method offered a notable reduction of the case base for \( N=2-20 \). The reduction for \( N=50-100 \) was marginal and there was no reduction at all for \( N=200 \).

As for the methods where \( L \) is determined automatically (random selection with 'add-if-better' and 'pick-best'), Table 5 indicates that they provide a large reduction to the total number of distinct examples used in the system. For 'add-if-better' as well as for 'pick-best' a notable reduction was obtained for all \( N \). It is clear that for both techniques it is beneficial to use very low number of examples in each sub-classifier \( N \) if case base reduction is a concern.

3. Final validation

To further validate the obtained results an additional experiment was conducted. The entire Database 1 was used as a development dataset. Database 2 was used as a testing dataset. It simulates a real-life scenario when a limited dataset is available to
Table 5. Total number of distinct examples selected for the random selection approach

<table>
<thead>
<tr>
<th>N</th>
<th>random selection</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>'no-control'</td>
<td>'add-if-better'</td>
<td>'pick-best'</td>
</tr>
<tr>
<td>2</td>
<td>219.8±4.3</td>
<td>45.7±10.9</td>
<td>17.7±5.3</td>
</tr>
<tr>
<td>5</td>
<td>484.1±8.4</td>
<td>106.9±21.9</td>
<td>46.2±12.5</td>
</tr>
<tr>
<td>10</td>
<td>796.5±10.4</td>
<td>176.7±36.9</td>
<td>84.8±21.8</td>
</tr>
<tr>
<td>20</td>
<td>874.0±11.0</td>
<td>293.3±57.0</td>
<td>140.6±34.7</td>
</tr>
<tr>
<td>50</td>
<td>1335.1±3.8</td>
<td>556.5±113.1</td>
<td>257.6±73.6</td>
</tr>
<tr>
<td>100</td>
<td>1349.4±0.8</td>
<td>841.6±142.1</td>
<td>429.7±132.7</td>
</tr>
<tr>
<td>200</td>
<td>1350.0±0.0</td>
<td>1149.7±124.4</td>
<td>730.6±197.1</td>
</tr>
</tbody>
</table>

the system designer and the constructed system is tested on new, unknown examples.

The baseline AUC performance (original IT-CAD) for this scenario was 0.871±0.020 (estimate based on 5000 bootstrap samples). The best test AUC performance provided by random division technique was 0.914±0.016 and was obtained for $L=90$ resulting in $N=16.7$ ($1500/90$) examples in a sub-classifier in average. The obtained performance was statistically significantly better than the baseline performance ($p < 0.001$).

Random selection with 'no-control' resulted in the maximum test AUC performance (estimate based on 5000 bootstrap samples) of 0.909±0.016 ($L=160$) for $N=2$, 0.912±0.016 ($L=60$) for $N=5$, 0.903±0.017 ($L=60$) for $N=10$, 0.908±0.016 ($L=60$) for $N=20$, 0.918±0.015 ($L=30$) for $N=50$, 0.912±0.016 ($L=50$) for $N=100$ and 0.909±0.016 ($L=18$) for $N=200$. The resulting AUC performance for all $N$ was statistically significantly better than the baseline (two-tailed $p < 0.001$). To further validate the conclusions concerning the drop of performance for high $L$ caused by overtraining of the combiner, the performance for $L=500$ was examined. It was con-
siderably lower than the maximum performance for all values of \( N \) and was equal to 0.881±0.019 for \( N=2 \), 0.860±0.021 for \( N=5 \), 0.875±0.020 for \( N=10 \), 0.874±0.019 for \( N=20 \), 0.882±0.019 for \( N=50 \), 0.878±0.020 for \( N=100 \), 0.874±0.019 for \( N=200 \).

Applying random selection with 'add-if-better' provided the following test AUC performance: 0.903±0.017 for \( N=2 \) (\( L=22 \)), 0.916±0.015 for \( N=5 \) (\( L=17 \)), 0.914±0.015 for \( N=10 \) (\( L=24 \)), 0.907±0.016 for \( N=20 \) (\( L=19 \)), 0.906±0.017 for \( N=50 \) (\( L=13 \)), 0.896±0.017 for \( N=100 \) (\( L=12 \)) and 0.901±0.017 for \( N=200 \) (\( L=9 \)). The performance was statistically significantly better (two-tailed \( p < 0.001 \)) for all \( N \). Applying random selection with 'pick-best' also statistically significantly improved the performance of the original IT-CAD for all \( N \). The test AUC performance was equal to 0.901±0.017 for \( N=2 \) (\( L=11 \)), 0.904±0.017 for \( N=5 \) (\( L=14 \)), 0.913±0.016 for \( N=10 \) (\( L=9 \)), 0.906±0.016 for \( N=20 \) (\( L=7 \)), 0.899±0.017 for \( N=50 \) (\( L=4 \)), 0.900±0.017 for \( N=100 \) (\( L=6 \)) and 0.903±0.017 for \( N=200 \) (\( L=11 \)). It is apparent that random selection with 'add-if-better' as well as with 'pick-best' allowed for a significant reduction of the total number of examples. Overall, the final validation results were consistent with the conclusions drawn in the primary experiment.

4. A closer look into adaptive process of building ensembles

In the process of building ensembles using the two proposed adaptive methods, at each step, performance of the ensemble is evaluated. This performance is calculated using internal crossvalidation with the assumption that such evaluation will reflect the test performance of the system. In this subsection, performance is monitored during the ensemble construction to examine if the above assumption is in fact valid. Figures 17 and 18 present the validation and test ensemble performance at each step of extending the ensemble until the stop criterion is satisfied.

It can be seen that even though the validation performance strictly raises in each step (internal algorithm condition), test performance can occasionally decrease. How-
Figure 17. Validation and test ensemble performance at each step of building the ensemble for 'add-if-better' method
Figure 18. Validation and test ensemble performance at each step of building the ensemble for 'pick-best' method
ever, the graphs show that in fact test performance correlates well to validation performance.

E. Conclusions and discussion

In this part of the dissertation research, the effectiveness of ensemble techniques was evaluated with the information-theoretic CAD system for detection of masses in mammograms. Advantages and limitations of these techniques were discussed. Two general approaches to constructing sub-classifiers were compared: random division and random selection and LDA classifier was used as a second-level combiner. In response to one of the limitations of ensemble classifiers, specifically the fact that it is often not clear how many sub-classifiers should be included in the ensemble, two adaptive methods were proposed that determine the number of sub-classifiers automatically.

The study results allow us to draw the following conclusions:

- All examined ensemble methods provide a similar and statistically significant improvement in performance ($AUC = 0.898 \pm 0.026$ to $AUC = 0.905 \pm 0.024$) as compared to the original system ($AUC=0.865 \pm 0.029$).

- For the methods where the number of sub-classifiers $L$ is determined a priori by the system designer (random selection with ‘no-control’ and random division), the obtained performance is highly dependent on this choice.

- When random selection is used to construct sub-classifiers, the achieved performance is not affected by the number of examples $N$ in each sub-classifier.

- Ensemble techniques can be used to notably reduce the total number of examples used in the case-based system.
The two adaptive methods (random selection with 'add-if-better' and 'pick-best') proposed in this chapter turn out to be the most effective if a case base reduction is a concern.

- The total number of distinct examples can be reduced to as few as $45.7 \pm 10.9$ examples (3.4% of the original case base size) for random selection with 'add-if-better' method.

- The total number of distinct examples can be reduced even further to $17.7 \pm 5.3$ examples (1.3% of the original case base size) if random selection 'pick-best' method is used.

To conclude, ensemble techniques turned out to be very effective in the examined case-based system. The two proposed incremental adaptive algorithms of constructing ensembles are preferable since in addition to a significant improvement in performance they allow for substantial reduction of case base size and they automatically adapt the number of sub-classifiers to the classification problem at hand.
CHAPTER V
ADDITIONAL CONSIDERATIONS AND FUTURE RESEARCH DIRECTIONS

A. Introduction

In the previous part of the dissertation, machine learning and computational intelligence techniques were proposed within three paradigms: decision optimization, case base reduction and building ensembles. This chapter provides additional considerations on the following issues:

• How do the proposed techniques compare to each other? (Section B)

• Can the proposed techniques be combined to provide further improvement in performance? (Section F)

• What are possible further improvements (Sections C and D) and other applications of the proposed algorithms (Section E)?

B. Comparison of the methods

1. Concept and methods

In the three previous chapters, three general approaches to improving CB-CAD were presented. It was shown that each approach can statistically significantly improve classification performance of the system at hand. Although based on the experimental results aforementioned, some conclusions can be made on the comparison of
the methods, here a rigorous comparison is presented. From each chapter one leading method was selected. This resulted in three methods being compared.

2. Experimental design and results

In the experiments, a database of 1820 regions of interest was used as in Chapters II and IV. 10-fold crossvalidation scheme was applied.

The results of the comparison are presented in Figure 19. The average AUC performance of the original IT-CAD was equal to 0.862±0.029. The best performance was obtained for random mutation hill climbing selection. The best (across when varying desired case base size) average AUC performance for this method was equal to 0.915±0.018 and was obtained for 400 selected examples. The performance for the system with GA-generated importance weights was slightly lower (AUC=0.9068±0.0180). The difference was statistically significant (p < 0.001). Even though RMHC provided a slightly better performance, it has to be noted that obtaining such high performance assumes finding the number of examples providing the optimal performance which in practice requires some additional crossvalidation and may be difficult. Introducing importance weights does not require such additional calculations. The average AUC performance of the 'pick-best' ensemble was equal to 0.894 ± 0.021. It was statistically significantly lower than the performance of RMHC (p < 0.001) and the performance of the system with importance weights (p < 0.001). Note, that even though the 'pick-best' ensemble provided slightly worse performance than the two other methods, it still significantly outperformed the original system (p < 0.002) and provided a dramatic reduction of the case base to only 15.4±4.6 examples (less than 1% of the original case base).
Figure 19. Comparison of the best techniques evaluated in the dissertation
3. Conclusions

The comparison of the most effective methods evaluated in this dissertation show that they all provide similar improvement to the case-based system. Different methods can be chosen based on the ultimate goal of the optimization (maximum performance, case base reduction) or the available database (number of examples for additional crossvalidation).

C. Using importance weights to reduce case base

1. Concept and methods

This short section is to evaluate the effectiveness of the importance weights found by the genetic algorithm (as proposed in Chapter II) for the selection task. The hypothesis is that the least important examples (those with lowest importance weights) can be removed from the system without compromising performance.

2. Experimental design and results

As in the previous section the dataset of 1820 ROIs depicting masses (901) and normal tissue (919) were used for the experiments. 10-fold crossvalidation data handling scheme was applied.

The curve showing the average AUC performance of the IT-CAD system based on the $k$ examples with the highest importance weights is shown in Figure 20. It is compared to the performance of RMHC. The maximum average AUC obtained for the weights-based reduction was equal to $0.888\pm0.026$ (for 500 selected examples) which means an improvement as compared to the original system. The results, however are notably worse than the results for RMHC.
Figure 20. ROC performance when GA-generated weights are used to reduce the case base compared with the performance of random mutation hill climbing.
3. Conclusions

The experiments of this section show that GA-generated importance weights can be applied to the task of selecting examples in the case-based system. However, the obtained performance for a given case base size is worse than for RMHC. This result is not surprising, since the objective function in RMHC optimization is the same as the final evaluation function (performance of the system with a reduced case base). GA-based weighting is guided by another objective function (performance of the system with all the examples with importance weights). Therefore the latter method is not well tailored to the task of selecting examples.

D. Using feature-based k-means clustering in building ensembles*

1. Concept and methods

In this section a possible extension to the ensemble methods evaluated in Chapter IV is presented. Specifically, the random division approach is modified such that k-means clustering is applied to cluster the case base instead dividing it randomly.

To apply k-means clustering, features have to be extracted from the regions of interest. Popular texture features were used for this purpose. Each image was treated as a function assigning a gray level to each position (resolution cell) (Haralick and Dinstein, 1973) and formally

\[
X : L_x \times L_y \rightarrow G
\]

where \(L_x\) is a horizontal spatial domain, \(L_y\) is a vertical spatial domain and \(G\) is a set of gray levels (\(|G| = N_G\)). For each image \(X\) a gray-level co-occurrence matrix (GLCM, also called a gray-tone spatial-dependence matrix) was extracted (Haralick

* This section, with modifications, has been published in (Mazurowski et al., 2008c).
and Dinstein, 1973). This matrix will be referred to as $P(i,j)$. For each $i$ and $j$, $P(i,j)$ is equal to the number of pairs of pixels with a gray level value of $i$ and $j$ that are distanced by $d$ pixels in the direction indicated by the angle $a$. In this study, the simplest scenario is used where $a = 1$ and $d = 0^\circ$. In this scenario GLCM is formally defined as (Haralick and Dinstein, 1973)

$$P(i,j) = \|(k,l), (m,n) \in (L_y \times L_x) \times (L_y \times L_x) : k - m = 0, |l - n| = 1, X(k,l) = i, X(k,l) = j\|$$

where $|S|$ is a number of elements in the set $S$. Further, the GLCM was normalized such that

$$p(i,j) = \frac{P(i,j)}{R}$$

where

$$R = \sum_i \sum_j P(i,j)$$

is a normalizing factor. Also, two marginal probability matrices were calculated:

$$p_x(i) = \sum_j p(i,j)$$

$$p_y(j) = \sum_i p(i,j)$$

Based on the image $X(k,l)$ and the gray-level co-occurrence matrix, the following features were calculated (Haralick and Dinstein, 1973; Sahiner et al., 1996a):

1. mean intensity

$$f_1 = \frac{\sum_k \sum_l X(k,l)}{|L_x||L_y|}$$
2. standard deviation of intensity

\[ f_2 = \sqrt{\frac{1}{|L_x||L_y| - 1} \sum_k \sum_l (X(k,l) - f_1)^2} \]

3. entropy of intensity

\[ f_3 = -\sum_m (p_{hist}(m) \log(p_{hist}(m))) \]

where \( p_{hist} \) is a histogram of the image \( X \)

4. contrast

\[ f_4 = \sum_{n=1}^{N_2-1} n^2 \left( \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} p(i,j) \right) \]

5. correlation

\[ f_5 = \frac{\sum_i \sum_j (i,j)p(i,j) - \mu_x \mu_y}{\sigma_x \sigma_y} \]

where \( \mu_x, \mu_y, \sigma_x, \sigma_y \) are means and standard deviations of intensity

6. energy

\[ f_6 = \sum_i \sum_j (p(i,j))^2 \]

7. homogeneity

\[ f_7 = \sum_i \sum_j \frac{p(i,j)}{1 + |i-j|} \]
Two different approaches were investigated to decompose the original case base. The first approach was random division of the case base into a prespecified number of separate groups. The second approach relied on k-means clustering of the examples according to the above texture features extracted from the ROIs. K-means is a simple clustering algorithm that divides the dataset into groups of similar (close) examples based on the provided features. More details about the algorithm can be found in (Kantardzic, 2003). After the decomposition, each of the resulting groups of examples was used to construct a separate IT-CAD sub-classifier.

In the final step of the proposed method, the predictions of the individual IT-CAD systems were combined into one final decision. For that purpose, the outputs of IT-CAD classifiers were treated as input features to a second-level combining LDA classifier.

To train the combining classifier, leave-one-out data handling scheme was used within the training dataset in the following way. Each of the examples was once excluded from the knowledge database of the ensemble system (i.e. from the database of one of the IT-CAD classifiers) and used as a query to the system. The responses of the classifiers in the form \((DI_1, DI_2, ..., DI_n)\) together with the ground truth of the query were used as training examples to train the second-level LDA combiner. The same method of combining sub-classifiers was used in Chapter IV.

2. Experimental design and results

The underlying hypothesis of this study was evaluated using a database of 1820 ROIs extracted from the Digital Database of Screening Mammography (DDSM). Of those, 901 ROIs contained biopsy-proven masses and 919 ROIs were normal.

The ensemble IT-CAD system was evaluated using leave-one-out (LOO) cross-validation as follows. The original dataset was split 1820 times into a training dataset of 1819 ROIs, with one ROI excluded for testing each time. For each split, the entire
Figure 21. ROC performance of the original IT-CAD system, the featured-based LDA, and the ensemble IT-CAD system for the two grouping techniques (i.e., random division and feature-based, k-means clustering)
Figure 22. ROC performance of the original IT-CAD system, the feature-based LDA, and the ensemble IT-CAD system for the two grouping techniques (i.e., random division and feature-based, k-means clustering) for high classifier sensitivities.
algorithm of grouping and creating the combiner was executed using only the 1819 training ROIs and evaluated on the testing ROI. The number of groups varied between 2 and 20 to assess the importance of this parameter.

Figures 21 and 22 show the ROC performance of the original IT-CAD system, the feature-based LDA, and the ensemble IT-CAD system for the two grouping techniques (i.e., random division and feature-based, k-means clustering). The results indicate that the ensemble IT-CAD system shows significant improvement as compared to the original system as well as the feature-based LDA. Using random division into 20 groups results in performance of AUC=0.90±0.01 which is statistically significantly better (two-tailed p value < 0.05) than the original IT-CAD (AUC=0.86±0.01) and the feature-based LDA (AUC=0.77±0.01). A small but statistically improvement was further obtained using the k-means clustering scheme. Specifically, the performance for 20 clusters was AUC=0.91±0.01 and it was statistically significantly better than that for the random division approach.

Furthermore, the number of groups played a significant role in the effectiveness of the ensemble technique. The general trend was that more groups offered better performance for both random and feature-based grouping. This result is consistent with the results in Chapter IV. The relation between the ensemble performance and number of groups is presented in Figure 23. It is notable that for smaller number of groups, the feature-based clustering scheme is dramatically more effective than the random division grouping scheme.

3. Conclusions

This expended the previously evaluated ensemble techniques by applying a feature-based clustering to decompose the set of available examples. The proposed technique of feature-based decomposition was effective in improving the performance of information-theoretic CAD system. This result supports the idea of fusion of feature-
Figure 23. ROC performance of the two ensemble approaches (random division and feature-based, k-means clustering) for various numbers of groups.

based and featureless approaches in CAD. Further improvement may be possible by incorporating more sophisticated features. This can be a part of future research.

E. Using ensembles to assess reliability*

1. Concept and methods

In previous chapters it is shown that ensemble techniques can be applied to significantly improve performance of information-theoretic CAD system for detection of masses in mammograms (Mazurowski et al., 2008c).

In this work this approach is extended by proposing an ensemble-based technique for reliability assessment in CAD systems. The concept of reliability was previously...

*This section with minor modifications has been included in (Mazurowski et al., 2008d).
presented in the context of feature-based CAD systems (Habas et al., 2007) and IT-CAD (Habas et al., 2006). Reliability is an index assigned to each CAD decision specifying to what degree the response of the CAD system can be trusted. In this section a method of reliability assessment for ensemble-based CAD systems is presented that is based on the variability of responses of the ensemble elements. The proposed method is computationally inexpensive and does not require additional training examples.

The ensemble system is constructed in two steps using the random division approach (Mazurowski et al., 2008c). In the first step the database of examples is divided randomly into \( N \) mutually exclusive, approximately equal subsets. Based on each of these subsets, an IT-CAD system is constructed. In the second step, the decisions are combined into one final prediction. Linear discriminant analysis (LDA) is used for that purpose. The LDA combiner is trained using leave-one-out approach within the training dataset.

The task is to assign a reliability index to a particular CAD decision. The reliability index is a measure of how accurate the particular CAD decision is expected to be. The hypothesis of this study is that the reliability of the decision is dependent on the variability of the responses of the ensemble components. Specifically, if there is an agreement between the ensemble component, the decision can be considered reliable. On the other hand, when the ensemble components do not sufficiently agree, the CAD decision should not be trusted. To assess variability of the predictions among the ensemble components, the standard deviation of their decision indices is used. It must be noted that a similar approach has been previously applied to show variability of responses to a query in the context of training artificial neural networks (Jiang, 2003).
2. Experimental design and results

For this study a database of 1820 regions of interests (ROIs) was used. The ROIs were extracted from the Digital Database of Screening Mammography (DDSM) (Heath and et al., 1998). Of those, 901 depicted biopsy-proven masses and 919 depicted normal tissue.

To evaluate the technique proposed here, a 10-fold crossvalidation data handling scheme was used. The database was divided 10 times into two mutually exclusive sets: a training set (1638 examples) and a testing set (182 examples). The entire process of constructing the ensemble components was performed using the training examples and the system was tested using the test set.

To evaluate effectiveness of the constructed reliability measures, test examples were stratified into five equal-sized groups based on the amount of variability among the decisions made by the ensemble components. The test examples with the lowest variability were included in the "high reliability" group (H). The ones with the highest variability were included in the "low reliability" (L) group, etc. Figure 24 shows the average ROC performance obtained by the ensemble for each stratification group. To
Figure 25. Relation between the final output of the ensemble and standard deviation of the ensemble elements' outputs for different numbers of ensemble elements (a) 10, b) 20, c) 50). Red circles represent negative and blue crosses represent positive test samples.

obtain the average performance, the stratification was performed in the same way for all 10-fold splits. The graph clearly shows that the proposed reliability index can successfully identify the test patterns with least reliable responses. Thus, the reliability index correlates strongly with the ROC performance achieved by the system at each stratum. Note that the baseline ROC area of the ensemble system was $0.902 \pm 0.017$ for 10 elements, $0.894 \pm 0.018$ for 20 elements and $0.898 \pm 0.018$ for 50 elements (the performance of the original system was $0.864 \pm 0.027$). Therefore, the low reliability cases are less accurate than what indicated by the baseline performance. Similarly, the high reliability examples are more accurate than indicated by the baseline performance.

Figure 25 shows the relation between the standard deviation among the responses of the ensemble elements and the final response of the system. It can be seen that even though some correlation between these two variables is apparent, no simple dependency is present. This fact suggests that combining the CAD output and the reliability index into one final decision may lead to an overall more accurate system.
3. Conclusions

It was shown previously that introducing the ensemble approach to classifiers can improve the classification performance. In this preliminary study a variability-based technique for reliability assessment of ensemble classifiers was evaluated. The proposed technique is computationally inexpensive (does not need any additional training or development) and does not require additional training examples. The results show that the technique successfully identifies test examples with low reliability. Low correlation of the reliability index with the classifier output encourage further studies for effective combination of those two components for an improved final decision.

F. Combining methods

1. Concept and methods

In this dissertation, it is shown that the proposed methods provide similar notable improvement in the system performance as compared to the original IT-CAD system. This section tackles a question of whether the proposed methods can be combined to provide even better improvement. There are multiple ways of combining the methods. Here, the focus is on combining importance weights with example selection (Chapters II and III respectively).

The fusion is implemented as follows. First, an example selection is performed using random mutation hill climbing. The desired number of examples is set to 400 as the maximum performance was observed for this case base size (compare Section B of this Chapter). Then, after the case base is reduced, importance weights are found for the 400 selected examples using genetic algorithms. In the importance weights optimization, the entire development dataset was used (not only the 400 selected examples).
2. Experimental design and results

In this experiment the dataset of 1820 ROIs was used similarly as in other experiments of this chapter. 10-fold crossvalidation data handling scheme was applied to obtain a reliable estimation of the system performance.

First, to initially evaluate the possibility of combining the two methods a relation between decision values of the system optimized using RMHC and the system optimized by introducing importance weights was examined for one 10-fold crossvalidation split. Decision values for the test set are provided. This relation is presented in Figure 26. The correlation between decision values for the two systems was equal to 0.99. For comparison, relation between decision values for the importance weights-improved system and decision values for the original system as well relation between decision values for RMHC-improved system and decision values for the original system are presented in Figure 27 and Figure 28 respectively. The corresponding correlation values were 0.88 and 0.85 respectively. These results indicate a very poor possibility of further improvement of performance by combining the two methods.

The combination of the methods resulted in the average AUC performance 0.915±0.017 as compared to AUC=0.915±0.018 for the system improved by RMHC only. The additional improvement provided by introducing weights was marginal and wasn't statistically significant ($p > 0.4$).

3. Conclusions

The results obtained in this section suggest that combining the importance weights methods with example selection is very unlikely to provide a further improvement in performance. It is possible that the system improved by either of the methods approaches the ideal observer performance for the problem which means that the further improvement is simply impossible. Combining the proposed ensemble methods
Figure 26. Relation between decision values for the RMHC-improved and the importance weights-improved system for one 10-fold crossvalidation split
Figure 27. Relation between decision values for the importance weights-improved system and the original system for one 10-fold crossvalidation split.
Figure 28. Relation between decision values for the RMHC-improved system and the original system for one 10-fold crossvalidation split.
with the importance weights methods and example selection can be a topic of further research.
CHAPTER VI
CONCLUSIONS

In this dissertation a comprehensive framework for improving case-based classifiers using computational intelligence and machine learning methods was proposed. The framework was applied to a computer-aided medical decision system for detection of masses in mammograms and was tested using clinically relevant evaluation schemes.

The original contributions of this dissertation include:

- introduction of a novel algorithm for assigning case-independent importance weights to examples in case-based system (Mazurowski et al., 2008b); the proposed method provided a statistically significant improvement in classification performance

- adaptation of computational intelligence and machine learning-based case base optimization algorithms to clinically relevant optimization criteria and comparative analysis of some available algorithms (Mazurowski et al., 2007; Mazurowski et al., 2008e); statistically significant improvement in classification performance as well as notable reduction of the case base was obtained

- introduction of a novel incremental adaptive scheme for building ensemble classifiers (Mazurowski et al., 2009); the proposed algorithms automatically adapt size of ensemble to the problem at hand; the proposed techniques provide a statistically significant improvement in classification performance as well a notable reduction of the case base

- introduction of a novel technique of fusing feature-based and featureless case-
based systems using an ensemble approach (Mazurowski et al., 2008c); the new technique uses k-mean clustering to decompose the set of available examples; the proposed technique offered additional improvement in classification performance.

- introduction of a novel technique for evaluating reliability of individual classifier decisions (Mazurowski et al., 2008d); the technique is based on ensemble classifiers and for identifying unreliable decisions of the CAD system.
REFERENCES


CURRICULUM VITAE

Maciej A. Mazurowski

Contact information
Computational Intelligence Laboratory, Lutz Hall, Room 407
University of Louisville, Louisville, KY 40292, USA
Phone: 1 502 852 3165
E-mail: m.mazurowski@ieee.org
Web: www.mazurowski.com

Education
Currently: Doctoral candidate in Electrical Engineering, University of Louisville,
Louisville, KY, USA
July 2005: M.S. in Electrical Engineering, Wroclaw University of Technology, Wro­
claw, Poland
July 2005: M.A. in Philosophy, University of Wroclaw, Wroclaw, Poland

Main research interests
Applications of computational intelligence algorithms to medical decision systems.
Computational intelligence in control in single-agent and multi-agent systems. Evolu­
tion of communication in multi-agent systems.

Appointments
August 2008 – December 2008: Research Assistant, Department of Electrical and
Computer Engineering, University of Louisville, Louisville, KY, USA
August 2006 – July 2008: University Fellow, Department of Electrical and Computer
Engineering, University of Louisville, Louisville, KY, USA
August 2005 - July 2006: Teaching Assistant, Department of Electrical and Computer
Engineering, University of Louisville, Louisville, KY, USA

Professional Memberships
IEEE  student member since October 2006
IEEE Computational Intelligence Society  since October 2006
SPIE - student member since December 2007
Service to the profession

- Program Committee member, International Joint Conference on Neural Networks (IJCNN 2009), Atlanta, GA, USA, June 14–19
- Program Committee member, IEEE Congress on Evolutionary Computation (CEC 2009), Trondheim, Norway, May 18–21, 2009
- Publications Chair, Workshop on Building Computational Intelligence and Machine Learning Virtual Organizations, Fairfax, VA, USA, October 24, 2008
- Technical Committee Member, International Joint Conference on Neural Networks (IJCNN 2008), Hong Kong, China, June 1–6, 2008
- International Program Committee Member, IEEE Congress on Evolutionary Computation (CEC 2007), Singapore, September 25-28, 2007

Publications

Journals

Conference proceedings


- J. M. Zurada, J. Wojtusiak, F. Chowdhury, J. E. Gentle, C. J. Jeannot, and M. A. Mazurowski, *Computational intelligence virtual community: framework and implementation Issues*, International Joint Conference on Neural Networks (IJCNN 2008), June 1-6, 2008, Hong Kong, pp. 3152-3156


• P. M. Szecowka, A. Szczurek, M. A. Mazurowski, B. W. Licznierski and F. Pichler, *Neural network sensitivity analysis applied for the reduction of the sensor matrix*, Lecture Notes in Computer Science 3643, pp. 27-32, 2005


**Edited volumes**

• M. A. Mazurowski, J. M. Zurada, J. Wojtusiak, R. Ragade, J. Gentle, A. Abdullin (Eds.), *Proceedings of the 2008 Workshop on Building Computational Intelligence and Machine Learning Virtual Organizations* (Fairfax, VA, USA, October 2008)