MACHINE LEARNING FOR LARGE IMAGE DATASETS

A thesis submitted to the faculty of
San Francisco State University
In partial fulfillment of
The Requirements for
The Degree

Master of Science
In
Computer Science: Computing for Life Sciences

by

John Collins

San Francisco, California

May 2013

CERTIFICATION OF APPROVAL

I certify that I have read MACHINE LEARNING FOR LARGE IMAGE DATASETS by John Collins and that in my opinion this work meets the criteria for approving a thesis submitted in partial fulfillment of the requirements for the degree: Master of Science in Computer Science: Computation for Life Science.

Kazunori Okada
Associate Professor, Computer Science

William Hsu
Associate Professor, Computer Science

Dragutin Petkovic
Professor, Computer Science

MACHINE LEARNING FOR LARGE IMAGE DATASETS

John Collins
San Francisco, California
2013

This thesis looks at machine learning via two case studies. In the first, we apply CBIR to medical image analysis. While previous studies focused on feature design, our study focuses on metric design. Our technique learns a metric using information theoretic metric learning. We compare our learned metric on a SIFT bag-of-words system against *a priori* measures from literature and evaluate it using the *ImageCLEF*-2011 benchmark. Our results show the advantage of this metric learning approach and of L^1-distance based measures. The second case study is motivated by the need of X-ray crystallographers to elucidate molecular structure from quality crystallized proteins, and by the scarcity of successful crystals. The process is human-curated, time-consuming and error-prone and we aim to automate search and classification using machine learning. Our results prefer elastic net and cascade classifiers composing random forest with linear discriminant or naïve Bayesian techniques.

I certify that the Abstract is a correct representation of the content of this thesis.

_____ _____

Chair, Thesis Committee Date

ACKNOWLEDGEMENTS

Thanks especially to Professor Kazunori Okada, Oliver Newland and Steve Guerrero for their advice, technical contributions and collaboration.

Thank you to the College of Science and Engineering and Computer Science departments at SFSU for their support, specifically to Monique Jorgensen and Professor Dragutin Petkovic.

Thanks to *ImageCLEF* for their organization, curation and publication of the materials for the first study presented here. Thanks to Genentech who provided the data and supervision for the second case study in this thesis.

Thanks lastly to my great friends Margaret McCarthy and Kristin Pollock whose help editing various documents including this one was invaluable.

TABLE OF CONTENTS

LIST OF TABLES

Table Page

LIST OF FIGURES

Chapter 1

Introduction

Ours is a time of unprecedented plenty. Huge datasets of all kinds abound and so it is with images. Search and retrieval techniques have been brought to the masses via the internet and on comes the day when image search rivals textual. However, many challenges still remain in such fields where a gap exists between description and understanding. Even in more specific settings, automation is still catching up with human domain expertise in terms of image classification accuracy, especially when misclassification is costly. Machine learning attempts to learn from data, and, as there is a proliferation of image data, much can be and has been gained by applying machine learning to images. Many fields including industrial automation, medical robotics and security have been overhauled by the kind of automation

that combines image data with effective analysis. However, many potential benefits remain to be realized. For instance, automatic diagnosis is still a long way off, automated surgery is still guided by human hands and even the best face recognition systems are subject to confusion.

Image datasets pose unique challenges, as do all datasets, when data size is very large. Big data is unweildy. Even in the days of cloud computing, many researchers tend to store their data locally because the time spent in data transfer is unreasonably long, or the ownership restrictions too tight. Large capacity machines are required to hold the data in memory while processing and parallelization must often be employed if computations are to complete in a reasonable time. Fortunately, gallant strides continue to be made in all of the areas mentioned via the popularization of cloud computing and storage resources, GPU processing and the incorporation of parallelization techniques into the modern researcher's toolkit, as well as into the tools themselves.

However, image datasets present additional challenges not shared by datasets in general. Image processing suffers from the so-called *semantic gap*, whereby judging the similarity of two images as seemingly diverse as say, a human skin tumor and a microscopic tumor cell, may be reasonable for a human being but a very difficult problem for a computer. Even when the context is known and all images are of the same ilk, wherein we use human-applied labels to distinguish the image types, the labeling is subject to human error and uncertainty, and

so the extent to which an algorithm *can* learn is hindered. That said, modern embracement of statistics has helped to alleviate the burden of uncertainty in image processing by giving such uncertainty a name and a number.

1.1 Study Outline

In this thesis, we aim to address issues from two fields whose goals are to learn from large image datasets. Thus, the thesis will describe two case studies. In the first case study, in chapter 2, we will describe a machine learning approach to the field of Medical Content-Based Image Retrieval (M-CBIR) through the forum of the competition *ImageCLEF* [30]. In this problem setting, contestants are asked to construct a system which returns ranked sets of relevant images to given query images and are measured by the relevancy scores of their results. Because the domain is medical images and these images are drawn from a wide span of medical journal articles, the construction of a good model has obvious relevance for the medical community at large, from doctors in assisted automatic diagnosis to possible health care cost savings from other automated retrieval. The second case study, in chapter 3, presents an image classification problem from the domain of crystallography. In a crystal growth laboratory, researchers spend large chunks of their time manually searching for images which exhibit favorable characteristics for X-ray crystallography. Since such a

process is very time consuming, tedious, prone to error and, moreover, because crystals are very rare and thus success rates low, models which can either automate this process or greatly reduce the search space are worthy of investigation. Here, the problem is to take a large set of images from a crystal growth process, each labeled with one of three types, and build a model which will generalize well to new data and thus classify new images as they are generated.

1.2 Challenges

In case study 1, the problem specific challenges are two-fold. Firstly, the sheer data size is very cumbersome. This problem size involved more than 200,000 images. Secondly, the image set was composed of images from vastly different sources (X-ray, electron microscope, PET, photo) with no necessary contextual connection except that they come from the medical field. Given the *semantic gap* problem, therefore, feature types need to be very general and data size very large. In case study 2, the biggest challenge is that good crystals are difficult to grow and so there are naturally far more non-crystal than crystal images. Hence, the data is very imbalanced and so any statistical technique used will be challenged by negative bias. Both case studies have in common their problem domain, that is, large datasets of images. Many of the challenges encountered were of a similar type. For example,

both datasets resulted in large amounts of extracted data which proved difficult to process and rendered necessary the need for exploring solutions via the distribution of memory, and computational parallelization.

1.3 Approach

The M-CBIR task *ImageCLEF* is by nature an unsupervised one, and thus, no image labels are provided. We proposed a novel approach incorporating a metric learned on results of the 2011 competition and applied it to the 2012 competition [1]. This augmented our comparative study of the effect of varying the metric used to judge similarity on data vectors generated using standard techniques. This application of metric learning and the comparative study compose the novelty of our approach. For the crystal image problem, case study 2, the novelty of our approach is also two-fold. Firstly, the automatic generation of, and selection from, a large number of features has not been attempted in previous crystal classification studies. Secondly, the use of more sophisticated classifiers and the comparison and contrast between these classifiers is new in our study.

1.4 Contributions

In case study 1 we make two contributions. First, we propose a M-CBIR system adapting a similarity measure learned from data using information theoretic metric learning [2]. Second, we detail a comprehensive comparative study using various similarity measures on a large dataset and evaluated using the public benchmarking dataset available from *ImageCLEF*-2011. Our results show a preference for our metric learning approach and for other L^1-based measures that we tested. Case study 2 also produced two contributions. We demonstrate the benefit of automating the feature selection process for crystal image classification versus previous studies by allowing the data to speak for itself. We also show that more sophisticated classification techniques than those used in previous studies can be effective. Moreover, our study does not shy away from the data imbalance issue which is central to the real world problem.

Chapter 2

Case Study 1: Learning Metrics for Content-based Medical Image Retrieval

Following a brief introduction, this case study's exposition continues with an in-depth literature review in section 2.2. Section 2.3 presents our metric learning method and other technical components of M-CBIR methods evaluated in this study. Section 2.4 outlines our experimental study including data, experimental design, and results. Finally, Section 2.5

discusses our study's result and potential future work.

2.1 Introduction

In recent years, the application of content-based image retrieval (CBIR) [3, 4] to medical image analysis has become an active research field. Such medical CBIR (M-CBIR) focuses on retrieving medical images similar to a single, or a set of, query images without using semantic annotations, and can be applied to various decision support problems in pathology and radiology [5]. Typically, an M-CBIR is a two-step pipeline composed of feature extraction followed by similarity comparison, both of which are equally important for successful applications. Previous work on CBIR has led to the development of various feature designs, such as SIFT [6], SURF [7] and Gabor wavelets [8]. Despite the relative maturity of these feature designs, similarity measures in CBIR have not been investigated thoroughly. Previous studies on metric design in CBIR [9–11] are still few and the lack is especially evident for M-CBIR.

Addressing this shortcoming, we present our investigation on metric design for an M-CBIR application. Our contributions are two-fold. First, we propose an M-CBIR system with information theoretic metric learning that adapts its similarity measure according to known

relevance side-information [2]. Second, we report a comparative study with a comprehensive list of similarity measures of many types using a large dataset. Our experimental evaluation employs a public benchmarking dataset available from *ImageCLEF-2011*, which includes various 2D digital image sources (e.g., tomographic images, compositions, plots, etc) derived from figures in radiology journal documents. Our results demonstrate the advantage of our metric learning approach and of L^1-based measures that we tested.

2.2 Literature Review

Because this project touches on a number of different topics, namely the *ImageCLEF* competition and medical CBIR [4, 5, 12–15], feature extraction [6, 7, 16–22], and distance functions [9, 10, 23–28] , the literature review will contain a few disparate subsets.

2.2.1 Medical CBIR and *ImageCLEF*

As previously mentioned, CBIR stands for Content Based Image Retrieval [29]. The subfield of Medical CBIR (M-CBIR) refers to CBIR where the domain of study has to do with biological and medical research or clinical medicine. Development in M-CBIR has, of late,

been intertwined somewhat with *ImageCLEF*, which is the Image division of the wider-scope CLEF (Cross-Language Evaluation Forum), so these topics will be discussed together. Several good papers [4, 5, 12] give an overview of the subject and the problems existing therein. Other publications describe *ImageCLEF* in terms of its history and motivation [13], and more specifically the image retrieval subtopic [14] and its means of evaluation [15].

Müller [5] , and Lehmann [12] give good introductions to the field and its problems. In essence, and this is reflected in *ImageCLEF*, the problems abound because of the so-called *semantic gap*. As previously mentioned, this is a phenomenon whereby algorithms for comprehending scenes do a considerably worse job than a human might at comprehending substantively similar but contextually diverse images. In well controlled sub-domains of any field, say when images are all taken from the same sensor, are of the same object, and are sensed under controlled conditions, computers can do quite a good job. Ascribing similarity between two images of, say, a tumor is naturally more difficult when one is a digital image of an underarm bump caused by swollen Lymph nodes and the other is a microscopic tumor cell image. When faced with such contextual diversity, few untrained humans could identify such similarity and so it is with machines.

However, the fact remains that improvement in Medical Image Retrieval could be a game changer in terms of speed and accuracy of diagnosis, reduction in medical costs and automations which would free up experts for other tasks. In [5], Müller talks about Medical

Image Retrieval Systems being put to use mainly as picture archiving and communications systems (PACS). Archiving is a need which has intensified due to increased availablility of large amounts of data. Diagnosis and decision making, he argues, can be improved by finding "other images of the same modality, the same anatomic region or of the same disease". As it stands, most image systems use text annotations alone to operate but the *ImageCLEF* [14] IR competition has shown that incorporating image data with text can perform much better.

The *ImageCLEF* competition has come about to address the gaps mentioned above. The *ad hoc* IR competition is one of the sub-competitions, and it too is further subdivided into three tasks: IR using text alone, IR using text plus image data, and IR using image data alone. Most of the information for the discussion about *ImageCLEF* was taken from [30]. Clough et.al [13] give a background to Image Retrieval as it relates to *ImageCLEF* including its main aim of "investigating cross-language image retrieval in multiple domains". In later years this mission statement developed to "combining textual and visual features for cross-language image retrieval" .

In the *Ad hoc* Image Retrieval competition specifically, there have been a wide range of system types deployed. In the first few years of the competition, the dataset was much smaller and teams performed well with simple features like downscaled versions of the images and simple texture features like Gabor Wavelets [14]. However, as the database grew in size

and diversity, such features became less useful and visual patch-based features became the norm. Others [19, 20] have employed feature extraction and image representation methods similar to the one (SIFT) used in our study. The contribution of this paper is to add to the work on the *ImageCLEF* competition a comprehensive survey of *a priori* similarity measures applied to the search mechanism, and an application of the notion of metric learning [31] in a novel way using the results from a previous year's competition.

2.2.2 Feature Extraction and Image Representation

This section on feature extraction will talk about our choice of SIFT [6, 16] as a tool for extraction and image representation and its appropriateness for an image set such as the one provided by *ImageCLEF*. Other feature extractors were considered, but most extraction techniques do well only in controlled domains. For example, the PCA-based eigenface [32] recognition works well when all images are roughly similar but differ in some predictable characteristics. Such characteristics will be reflected in the variance and thus emphasized in the directions of the first few principal components. For example, if the database is composed exclusively of equal-sized digital camera images of faces all looking in the same direction this type of image representation may suffice. Similarly, generalized edge-detectors can be used to isolate the crucial part of an image [33] but such techniques work best when images contain things of the same general type, and are from the same kind of sensor. The

diversity in content and image source which is fundamental to a dataset such as the one used in *ImageCLEF* requires a robust feature extractor which can co-identify objects regardless of scale, rotation and image size, and can perform well even with a diversity of different image content.

SURF [7, 17, 18] and SIFT are the two most prominently used such extractors. A decision was made that SIFT was the most appropriate due to its wide use [14] and established performance [19, 20] on diverse datasets like this one. SIFT, a gradient based descriptor, is rare in its robustness due to its multi-scale and rotationally invariant representation of features.

2.2.3 Distance Metrics

This final section discusses some of the roots of the *a priori* distance measures employed in this study as well as a discussion of previous work done on learning metrics and its usefulness in CBIR. This section also sets the stage for the main contribution of this thesis.

A priori Metrics

Our work on *a priori* distance metrics will discuss a list of well and lesser known metrics [11] of varied type including the cross-bin variety which assume no order in feature representation. The set of such measure include EMD [10, 23, 24] and diffusion [25], and information theoretic methods as variants of Shannon entropy [34]. A full list of the *a priori* distance functions which will be compared is given below.

- Euclidean (L_2)
- Taxicab (L_1)
- Max Norm (L_∞)
- Cosine
- Correlation
- χ^2
- Kullback-Leibler Divergence
- Jeffrey Divergence (Jensen-Shannon)
- Kolmogorov-Smirnov Distance
- Cramer Von Mises
- Earth Mover's Distance
- Diffusion Distance

Other authors have completed comparisons of *a priori* similarity measures in the past [11] on different data sets. They found that certain measures performed better on smaller data

sizes and others performed better when data sizes were bigger. It is one of the proposed contributions of this study to perform a comparison of all available measures of dissimilarity to the *ImageCLEF* dataset for the purposes of Image Retrieval.

Metric Learning

Much previous work has been done on metric learning [2, 31, 35] but the novelty in the approach suggested here is to construct a relevance feedback loop from relevance judgements from a previous year (2011) of the *ImageCLEF* competition to construct a learned distance measure.

2.3 Our M-CBIR System

We designed a system from beginning to end, incorporating existing implementations where possible. Abstractly a generic system consists of the parts and flow shown in Figure 2.1. Our system will be detailed in this section.

This sets up the main problem. That is given a database and some input queries, score the

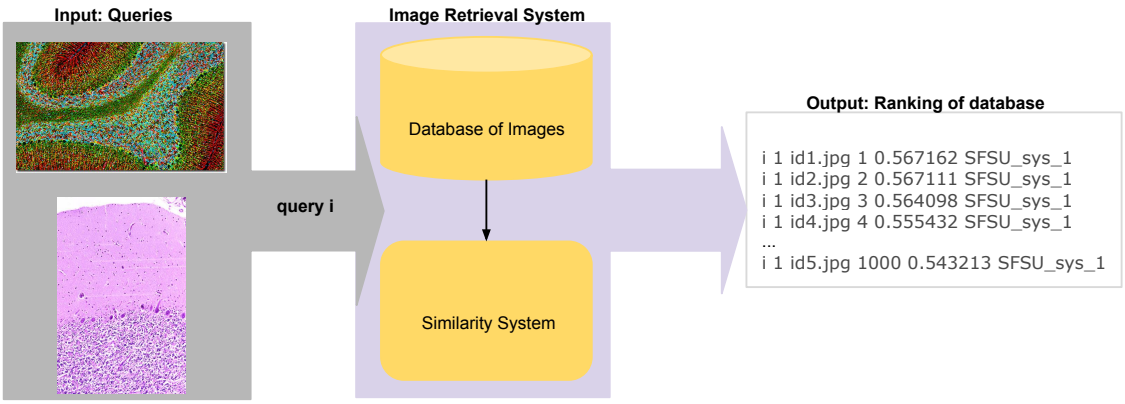

Figure 2.1: Generic *ImageCLEF* System

database images by similarity to the queries in some sense. Figure 2.2 lays out the system in more detail for a typical multi-image query. First we create a feature vector for each image, then we apply various transformations to each of these feature vectors. Next we judge the similarity of each database image to each query image by calculating the similarity between the feature vectors. Finally we score the rank of each database image as the mean of the individual similarities. How we describe an image using a feature vector, and how we discern similarity is the focus of this section.

2.3.1 Feature Set Design

We extracted a featurefrom each image in the well-known bag-of-words (BoW) scheme [36] with SIFT features [6] as described below.

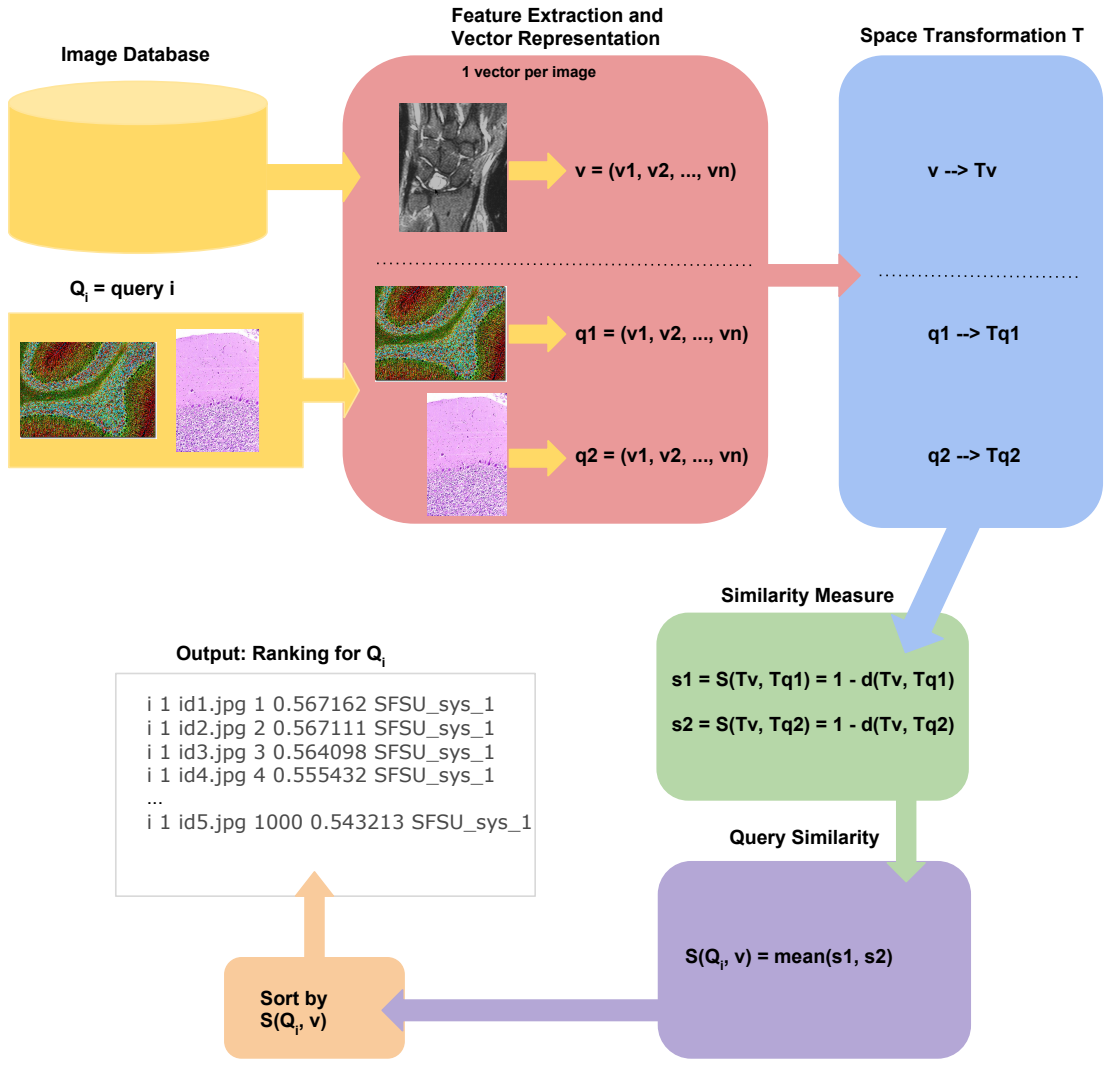

Figure 2.2: Detailed view of our System

Given a training dataset, we first construct a K-word *codebook* by 1) identifying and extracting SIFT features from all images in the dataset and 2) creating K representative features via K-means clustering over the extracted SIFT features. We use the standard algorithm for SIFT feature extraction, yielding a 128-element histogram of local gradient directions. We include the 4 extra parameters composed of the 2 spatial coordinates of the

SIFT key-point, the scale parameter, and the dominant-orientation parameter, making the total number of dimensions of our feature vector to be 132. Each SIFT feature vector is centered and scaled using Z-Score normalization, and we randomly initialize centers for the K-means clustering.

Given this codebook, the BoW method extracts a feature vector of length K for each new image by 1) performing the same SIFT feature extraction and 2) constructing a normalized histogram representing the frequency distribution of the extracted SIFT features with respect to the codebook. For each feature, we find the nearest-neighbor among the K representative codebook vectors that is closest - in the Euclidean sense - to the input feature. Finally, we perform a number of standard feature transformation such as PCA and TF-IDF, for better retrieval performance.

We consider two types of post-extraction feature transformation: principal component analysis (PCA) [37] and term frequency-inverse document frequency (TF-IDF) [38]. PCA is a standard dimension reduction method which computes an eigen subspace of (BoW) feature vectors derived from the training dataset. Each new feature vector is then projected onto this subspace before similarity comparison.

TF-IDF originally comes from textual data mining. Its goal is to penalize common words (i.e., codebook feature vectors) across the training dataset. The BoW feature vector de-

scribed above corresponds to TF. IDF is computed for each codebook vector as an inverse frequency of training images that include the codebook vector as a match. Each new feature vector is then multiplied with the resulting IDF filter.

We experiment with four types of feature transformation including combinations of PCA and TF-IDF: 1) PCA($\cdot$), 2) TF-IDF($\cdot$), 3) PCA(TF-IDF($\cdot$)), and 4) TF-IDF(PCA($\cdot$)) using operator notaton.

2.3.2 Database Ranking by Similarity Comparison

Given a query image, the goal is to rank database images according to their distance or similarity to the query. In some cases a query may consist of multiple images. In this case, we calculate the average similarity of the query set to each database image and use this average for the ranking. Many standard similarity measures exist, but most take the form of a distance/dissimilarity measure in their natural expression except for some rare cases (e.g., cosine similarity). When considering a dissimilarity measure $d(x, y)$, we calculate similarity with its additive inverse by $1 - d(x, y)$ where x and y are appropriately scaled so that $d(x, y) \in [0, 1]$. We also abuse the term *metric* to indicate both similarity and dissimilarity measures in this study. Strictly speaking, a metric is a distance function that satisfies three conditions of positive definiteness ($d(x, y) \geq 0$; $d(x, y) = 0$ *iff* $x = y$), symmetry

($d(x,y) = d(y,x)$), and the triangle inequality ($d(x,z) \leq d(x,y) + d(y,z)$). However, some standard dissimilarity measures we considered violate these condition (e.g., Kullback-Liebler divergence is not symmetric).

2.3.3 Metric Design by Learning

Metric Learning [35] is the process of adapting a metric of a set S according to side-information about the similarity or dissimilarity of some known data points in S.

Let $\mathbf{x} = (x_1, .., x_n)$ represent the n dimensional query image and $\mathbf{y} = (y_1, .., y_n)$ represent another image against which to be compared. Let $\boldsymbol{\lambda}$ denote an n-dimensional vector in which λ_i determines the weight given to the i-th feature $x_i \in \mathbf{x}$. A weighted L^2 metric on S can then be defined as $d_\lambda(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^{N} \lambda_i (x_i - y_i)^2}, \forall \mathbf{x}, \mathbf{y} \in S$.

A more general form is given by the Mahalanobis distance,

$$d_A(\mathbf{x}, \mathbf{y}) = ||\mathbf{x} - \mathbf{y}||_A = \sqrt{(\mathbf{x} - \mathbf{y})^T A (\mathbf{x} - \mathbf{y})} \tag{2.1}$$

where A is a symmetric, positive semi-definite matrix and $\boldsymbol{\lambda} = diag(A)$.

One idea of metric learning is to learn the appropriate weights $\boldsymbol{\lambda}$ or A from training data [35]. Different approaches have been reported in the literature for such metric learning [31].

We adopt information theoretic metric learning (ITML) [2] in our M-CBIR system. ITML is a popular metric learning algorithm that iteratively enforces pair-wise similarity/dissimilarity constraints, yielding the learned matrix A of the Mahalanobis distance as an output.

The Mahalanobis distance is a bijection to a Gaussian distribution with its covariance set as an inverse of A. Exploiting this bijective property, ITML poses the metric learning problem as a convex optimization of the relative entropy between a pair of Gaussian distributions with the unknown A and the identity matrix I under the similarity/dissimilarity constraints,

$$\begin{aligned} \min_{A \succ 0} \quad & KL(p(\mathbf{x}; \mathbf{m}, A) || p(\mathbf{x}; \mathbf{m}, I)) \\ \text{Subject to:} \quad & d_A(\mathbf{x}_i, \mathbf{x}_j) \leq u \;\; (i,j) \in S \\ & d_A(\mathbf{x}_i, \mathbf{x}_j) \geq l \;\; (i,j) \in D \end{aligned} \tag{2.2}$$

where S and D are the sets of similar and dissimilar points, respectively. This formulation regularizes the optimization problem so as to seek a metric that satisfies the given constraints

and is closest to the Euclidean distance.

Davis et al. [2] demonstrated the equivalence of this metric learning formulation and low-rank kernel learning problem [39], yielding an efficient solution to the problem in (2.2) based on Bregman's method [40]. This dual ascent optimization method iteratively projects onto one constraint at a time with a closed-form projection update without the need for costly numerical eigenvalue decomposition and is thus efficient.

Note that a pair-wise distance computation by Equation 2.1 can also be realized by first performing a linear transformation $S \mapsto T = A^{1/2}S$ and by computing the L^2 distance for the pair in T. This linear transformation maps similar data points in S closer together and dissimilar data points farther apart in T and yields a more computationally efficient pair-wise distance computation. Adopting this property, we treat the ITML's result A as a post feature transformation and evaluate it with different similarity measures in our experiment.

2.3.4 Standard Similarity Measures

The subjectivity inherent to the idea of similarity is reflected in the varying types of similarity measures which can be defined.

Let $\bar{x}$ represent the mean value of $\mathbf{x}$ and $\bar{y}$ that of $\mathbf{y}$, while $\boldsymbol{\mu}$ denotes an average of $\mathbf{x}$ and $\mathbf{y}$: $\boldsymbol{\mu} = \frac{\mathbf{x}+\mathbf{y}}{2}$. Further, let $\mathbf{X}$ and $\mathbf{Y}$ represent the cumulative distributions of $\mathbf{x}$ and $\mathbf{y}$ when they are considered as probability distributions ($\sum_{i=1}^{n} x_i = \sum_{i=1}^{n} y_i = 1$), respectively. That is $\mathbf{X} = (X_1, ..., X_n)$ where $X_j = \sum_{i=1}^{j} x_i$ and similarly for $\mathbf{Y}$ and $\mathbf{y}$. We use $\mathbf{z} = (z_1, z_2, \cdots, z_n)$ and $\mathbf{z}^{(l)}$ to denote the l-times iteratively Gaussian-smoothened, then 2-downsampled vector representation of $|\mathbf{X} - \mathbf{Y}|$. The following list of various similarity or dissimilarity measures were considered in our study.

$$L^2(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x}-\mathbf{y})^t(\mathbf{x}-\mathbf{y})} \tag{2.3}$$

$$L^1(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} |x_i - y_i| \tag{2.4}$$

$$L^\infty(\mathbf{x}, \mathbf{y}) = \max_{i=1}^{n} |x_i - y_i| \tag{2.5}$$

$$CO(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{x} \cdot \mathbf{y}}{\|\mathbf{x}\|\|\mathbf{y}\|} \tag{2.6}$$

$$CC(\mathbf{x}, \mathbf{y}) = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n} (x_i - \bar{x})^2 (y_i - \bar{y})^2}} \tag{2.7}$$

$$CS(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} \frac{(x_i - \mu_i)^2}{\mu_i} \tag{2.8}$$

$$KL(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} x_i \log \frac{x_i}{y_i} \tag{2.9}$$

$$JF(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} x_i \log \frac{x_i}{\mu_i} + y_i \log \frac{y_i}{\mu_i} \tag{2.10}$$

$$KS(\mathbf{x}, \mathbf{y}) = \max_{i=1}^{n} |X_i - Y_i| \tag{2.11}$$

$$CvM(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} (X_i - Y_i)^2 \tag{2.12}$$

$$EMD\text{-}L^1(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{n} |X_i - Y_i| \tag{2.13}$$

$$DD(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{\log_2 n} \sum_{j=1}^{n/2^j} \mathbf{z}_i^{(j)} \tag{2.14}$$

where L^2: Euclidean distance, L^1: cityblock distance, L^∞: infinity distance, CO: cosine similarity, CC: Pearson correlation coefficient, CS: Chi-square dissimilarity [11], KL: Kullback-Liebler divergence [11], JF: Jeffrey divergence [11], KS: Kolmogorov-Smirnov divergence [11], CvM: Cramer-von Mises divergence [11], $EMD\text{-}L^1$: earth movers distance with L^1 ground distance [24] (EMD in 1D feature space is equivalent to the Mallows Distance [41]), DD: diffusion distance [25].

2.4 Experiments

In this section we detail the experiments we conducted.

2.4.1 Data

We use datasets made available by *ImageCLEF* [30]. *ImageCLEF* has offered standardized benchmark tests for a variety of language-neutral CBIR tasks since 2003. The data used in our experiments are from the medical image retrieval task of the *ImageCLEF* competition administered in 2011 [42]. Three types of datasets were available for this study: *training*, *query*, and *relevance judgment*.

Training data consist of $230,088$ images taken from Pubmed Central database (`www.ncbi.nlm.nih.gov/pmc/`) that contains more than 1 million figures from published medical journals. Images are therefore of diverse types including those that have little relevance to our retrieval task, as shown in Figure 2.3.

Query data consist of 30 distinct queries, each of which consists of 1-3 query images. These query images are of standard medical image types of different modalities and fields of view. Some examples are shown in Figure 2.4.

Relevance judgment data provides our ground-truth information used in both metric learning and performance evaluation. For each query, a subset *pool* of the entire data set was first collected from the top N matches of an existing M-CBIR system by the *ImageCLEF*

organizers. These pooled images are then manually judged by physicians and medical students at Oregon Health and Science University using a web-based GUI tool to be either *relevant* or *irrelevant*. All images not in the pool are judged to be irrelevant. For all queries, these relevance scores are computed for all training images, without annotating the images, and stored in a file.

2.4.2 Performance Evaluation Measure

Mean average precision (MAP) is used as a measure to quantify performance of our M-CBIR systems. MAP is a popular performance measure in the information retrieval field, and is defined as the average per-query precision,

$$MAP = \frac{1}{Q} \sum_{q=1}^{Q} p_\mu(q) \tag{2.15}$$

where Q is the number of queries and

$$p_\mu(q) = \frac{1}{R_q} \sum_{k=1}^{n} P(k) \cdot rel(k) \tag{2.16}$$

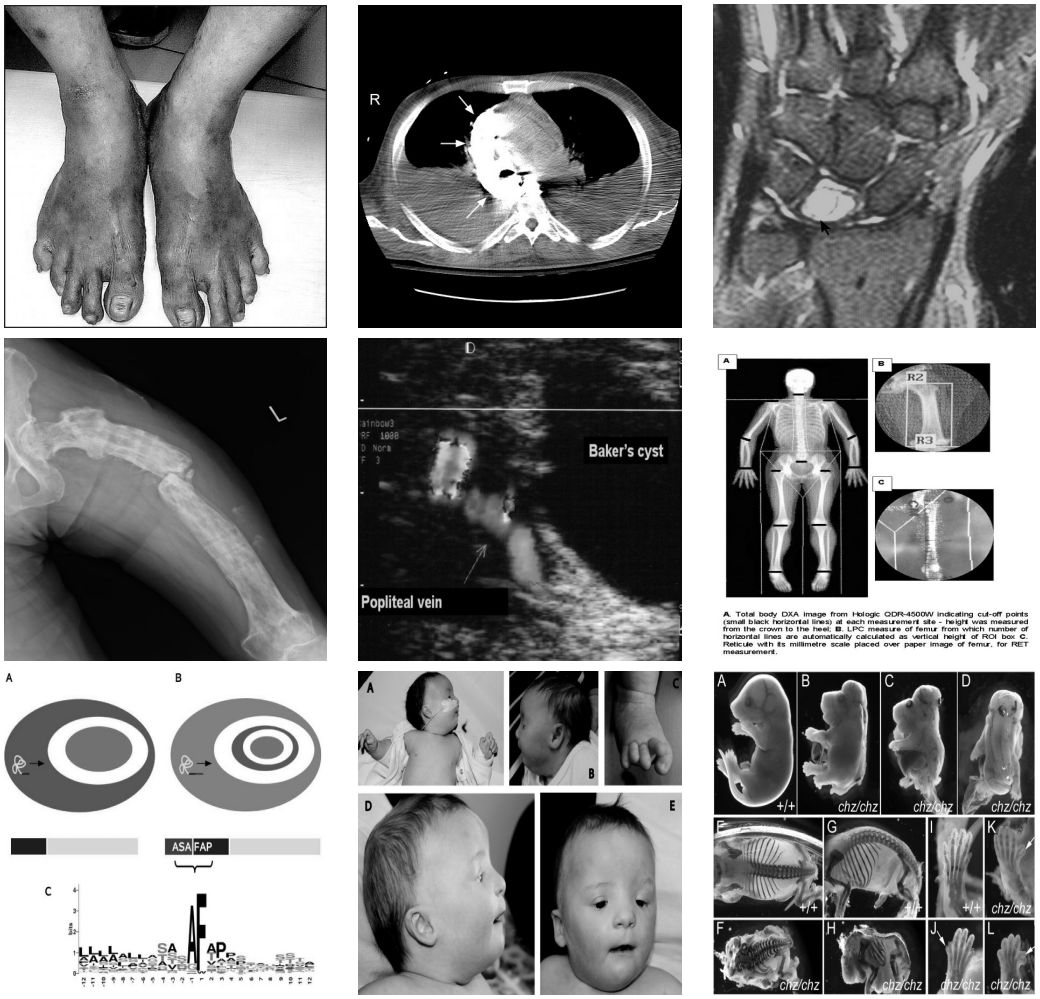

Figure 2.3: Various training images for *ImageCLEF-2011* displaying their diversity. Modalities of these images are, respectively: optical, CT, MRI, X-ray, ultrasound, DXA, graphical, optical and mixed

where $P(k)$ is the precision (ratio of relevant images in the top k) at the k-th image, R_q is the number of retrieved images which are relevant to the q-th query, $rel(k)$ is a binary indicator function for relevance or lack thereof, and n is the total number of images retrieved for the q-th query.

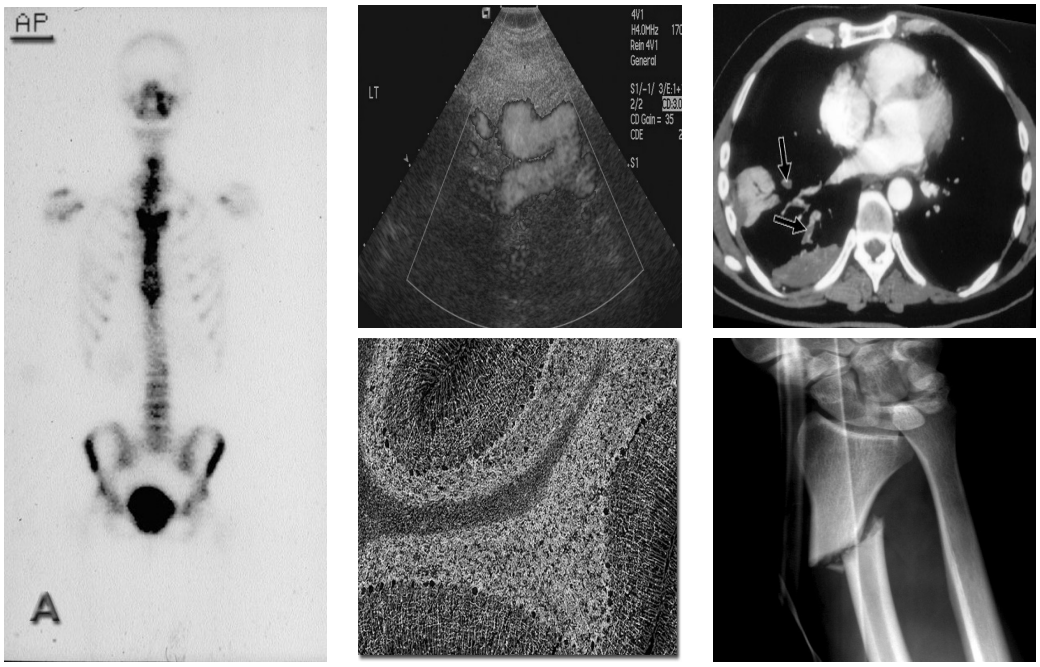

Figure 2.4: Examples of query images for *ImageCLEF-2011*

2.4.3 Results

We evaluate the MAP score of our ITML systems in comparison with the twelve standard similarity measures in section 2.3.4 applied on the four types of post-extraction feature transformations. We set the codebook size K to be 1000, following the previous M-CBIR report using similar feature design [20]. We did not observe benefits in learning a fully parameterized A, see section 2.3.3, in our pilot study and so, for computational simplicity, we utilize a metric learning formulation with a diagonal A.

Table 2.1 summarizes the resulting MAP scores. The highest scores of 0.0227 were achieved

by our proposed ITML transformation with the L^1 and diffusion distances. The next highest scores of 0.0214 were achieved with L^1 and diffusion distances without using any post-extraction feature transformation. Among the cases with post feature transformations, TF-IDF(PCA) performed best at 0.0209 with the correlation coefficients used as similarity measure.

Table 2.1: Result of similarity measure comparison using the MAP score with *ImageCLEF-2011* data. None: without feature transformation. PCA_M: codebook constructed using the first M principal components. PCA: all principal components. TF-IDF(PCA) is the TF-IDF transformation of the PCA transformed data. PCA(TF-IDF) is the PCA transformation of the TF-IDF transformed data. ITML is $A^{1/2}S$ where A is a covariance of Mahalanobis metric learned for S. Bold-typed numbers indicate the best performing combinations.

Measure	**Data Transformation**								
	None	PCA_{75}	PCA_{200}	PCA_{500}	PCA	TF-IDF(PCA)	TF-IDF	PCA(TF-IDF)	**ITML**
L^2	0.0169	0.0207	0.0168	0.0194	0.0203	**0.0208**	0.0157	0.0172	0.0126
L^1	**0.0214**	0.0183	0.0091	0.0196	0.0182	0.0180	0.0207	0.0180	**0.0227**
L^∞	0.0029	0.0032	0.0011	0.0012	0.0029	0.0016	0.0034	0.0097	0.0023
CO	0.0169	0.0207	0.0168	0.0194	0.0203	**0.0208**	0.0157	0.0173	0.0126
CC	0.0184	0.0207	0.0168	0.0194	0.0203	**0.0209**	0.0201	0.0172	0.0185
CS	0.0133	0	0	0	0	0	0.0163	0	0
KL	0.0004	0	0	0	0	0	0.0004	0	0
JF	0	0	0	0	0	0	0.0008	0	0
KS	0.0010	0.0176	0.0003	0.0020	0.0107	0.0176	0.0008	0.0008	0.0005
CvM	0.0011	0.0047	0.0017	0.0014	0.0091	0.0104	0.0009	0.0008	0.0006
EMD-L^1	0.0011	0.0031	0.0016	0.0014	0.0089	0.0098	0.0009	0.0006	0.0006
DD	**0.0214**	0.0183	0.0091	0.0196	0.0140	0.0137	0.0207	0.0177	**0.0227**

2.5 Discussion

We proposed a metric learning-based medical CBIR method and presented a systematic experimental comparison of various similarity measures by using a large public database. Our experimental results demonstrated an advantage of the proposed ITML approach which outperformed other CBIR metrics we tested.

In *ImageCLEF*'s medical image retrieval task in 2011, the best MAP score achieved by using only visual information was 0.0338 [42]. Our ITML-based score would have placed 10th place among 26 submissions in the competition. Our scores were relatively much lower than retrieval performance with text annotative information exploited, indicating the difficulty of the visual M-CBIR task we tackled.

Note also that since the diffusion distance is also based on L^1, the best performing standard metrics in our experiment were both based on the L^1 metric. This seems sensible because L^1 distance tends to outperform L^2 distance in a high-dimensional space. Since over-fitting in our metric learning was a concern, we chose to alleviate this by using a simpler form, forcing A to be diagonal. A small difference in the MAP score between the ITML results and the others supports this choice.

As future work, we plan to improve the overall retrieval performance by tuning our feature design. We also plan to compare different metric learning algorithms to better understand the role of metric design in this M-CBIR application.

Chapter 3

Case Study 2: Crystal Image Classification

This chapter introduces the difficult problem of crystal image classification and presents a series of experiments comparing different approaches to feature extraction, feature selection, model selection and classification.

3.1 Introduction

Protein crystallization [43] is a difficult process, generally garnering a low success rate for the growth of uncontaminated, well-ordered crystals which are large enough to exhibit a diffraction pattern when exposed to X-rays. Crystallographers at Genentech `www.gene.com`, in a high-throughput setting experiment with different growth processes in order to optimize for successful crystal rate and quality. Such researchers manually label images of these growth processes from circular wells and often have a great many images to examine manually. Thus, they would benefit from a process which could identify images of crystals reliably [44], and/or reduce the search space by identifying obvious non-crystals automatically. Our study aimed to investigate and implement such a process in order to improve the efficiency of such the workflow described with the larger goal of increasing the efficacy of protein structure characterization in biochemical research and pharmaceutical development at large.

This thesis describes a machine learning approach with a large annotated protein crystallization trial dataset and model selection via 10-fold cross validation. We yield several accurate 3 category classifiers based on the elastic net algorithm [45], and cascade approaches using Linear Discriminants [37] and naïve Bayesian [37] methods composed with random forest [46]. All of these techniques are powerful and popular techniques for building predictive models which are robust and efficient. We design our image feature set to be

both large enough in number and general enough in description to account for the variety of appearance and geometry of crystal formations. The features we consider include Gabor wavelets [47], gray-level co-occurrence matrix (GLCM) [48], and shape context [49]. Using our dataset, we performed feature selection using Random Forest (RF), Fisher's Discriminant Ratio (FDR) and Elastic Net (EN) eliminating FDR as a technique thereafter based on poor performance. We then compared model selection using classification results from among four classification methods of naïve Bayes (NB), linear discriminant analysis (LD), RF, and EN. The system diagram in Figure3.1 gives an outline of the system. For the remainder of this study we will refer to the crystal image categories as clear (for category 1), precipitate (for category 2) and crystal (for category 3).

Our Feature Selection experiments showed that about 2,000 of our features, properly selected, were enough to achieve the same accuracy as the entire data set in the case of RF compared with 1,597 or 2,042 non-zero coefficients chosen in the EN scheme for 2 and 3 category models respectively. In both cases this represented a significant reduction in complexity from 83,835 features.

Comparing models was a non-trivial task given the combined goals of maximal precision, recall and prediction accuracy, and minimal false negatives. Further, since a 2 category predictor which differentiated clear from the others was deemed to be useful, we also built models for the 2 category case. We decided that EN produced the best combination of

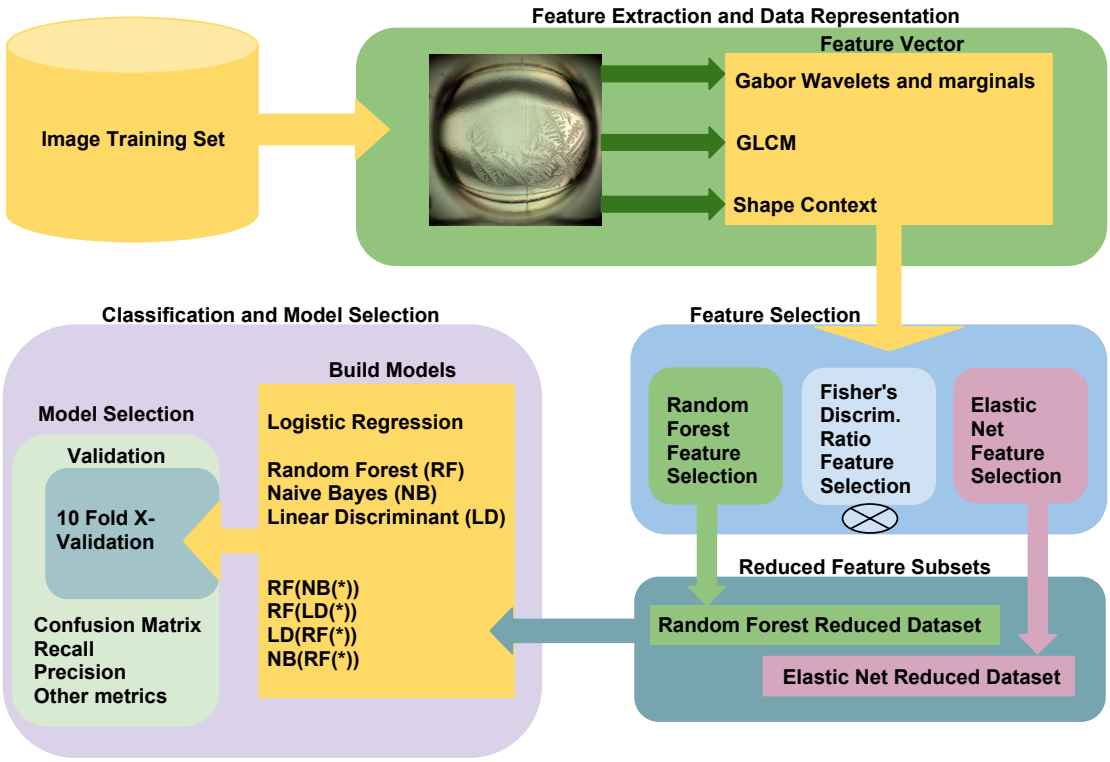

Figure 3.1: The Structure of our Study

these qualities. RF and EN both achieved similar prediction accuracies at around 85% and 82% for 2 and 3 category classification respectively. The EN model achieved 56.76% precision but only a meager 5.24% crystal recall. RF performed more poorly here but both NB and an RF- NB cascade classifiers achieved good recall at over 65%. Both had poor precision at around 17%. Recognition that RF performed best at categorizing clears from the rest motivated the implementation of a number of cascade classifiers to augment our comparative investigation by providing improvements in both recall and precision. Analyses of these results also motivated a problem-specific score which we will present later. Overall

the EN, and the LD- RF and NB-RF cascades showed both good accuracy and decent recall and precision while keeping the number of false negatives sufficiently low.

3.2 Literature Review

There have been several attempts [50–55] to automate classification of the photographic results of protein crystallization experiments. We examine some of these previous studies here.

An early approach to crystal classification was proposed by Bern et al. [51]. In order to detect crystals, the team segmented the drop from the well of the crystal images and incorporated boundary and edge detection to extract both curvature and gradient-based features. They sub-divided the crystal category into "high-confidence" and "micro-crystal" [51]. Out of a total of 1,834 images, 1,145 were clear, 497 contained precipitate, 189 were microcrystal hits, and 192 contained high-confidence crystals. The authors evaluated a manually-constructed decision tree and a C5.0 tree with boosting. The results showed a preference for the manually constructed tree, as can be seen in Table 3.1 [51].

In Cumbaa and Jurisica's work [52], the authors experimented with a much larger data

set and distinguished the images into two categories only, 5,600 crystal and 189,472 non-crystal, ignoring the precipitate class. The authors incorporated chemical data from the growth process into their feature set which was otherwise composed of correlations with micro-crystals, measures of straight lines, Euler-number measures, quadtree decomposition statistics, and multiple energy-based features. They used two standard classifiers: a Fisher LD and a kNN classifier. High overall accuracy was achieved in both cases at 83% and 85% respectively, and while kNN performed slightly better, they both achieved lower accuracy, 68.0% and 76.0% respectively, for crystal recall [52].

In Yang et al. [53], a promising classification system was proposed in which a much smaller data set ($n = 110$) was used. The authors achieved a crystal image detection rate of 84.8% with just 16 features. Their feature set was composed of GLCM and connected components measured in a small number of arrangements [53]. Their classifier was designed in a cascade manner (as seen here and elsewhere [54]). Its composition was a manually-constructed decision tree to separate "clear" from "non- clear", followed by a linear discriminant classifier to distinguish "precipitate" and "crystal" [53].

The best accuracy achieved from these papers was by Kawabata et al. [54]. Here, the authors used 5 classes including "crystal", "clear", and 3 distinct "precipitate" classes. Their study included a small data set of 874 images which did not have the data imbalance skewed against crystals which is a feature of our study and a feature of the real problem space.

They also used a cascade classifier composed of a linear discriminant to distinguish between crystal and non-crystal classes, followed by another linear discriminant to distinguish between crystal sub-classes. The performance achieved was excellent, with 90.1% recall for precipitate images as "precipitate", and 93.5% recall for "crystals".

In [55], Cumbaa and Jurisica took a broad approach, attempting to distinguish 10 classes. They were equipped with an enormous training set, drawing around 90% from 147, 456 labelled images. The authors extracted an eclectic array of features as detailed below, 14, 908 in all, and achieved a good crystal identification accuracy rate of around 80%.

- Energy (intensity change over area)
- Euler Numbers (number of objects minus number of holes)
- Radon-Laplacian
- GLCM features
- Edge Features (Laplacian and Sobel filter based features)

Table 3.1: Summary of Prior Work

					Recall	
Study (Classifier)	n	$n_{crystal}$	p	Acc.	Precip.	Crystal
Bern et al. '04 [51] (hand-built DT)	1,057	196	7	77.2	51.2	81.6
Bern et al. '04 [51] (C5.0 + AdaBoost)	766	286	7	75.7	68.5	78.7
Cumbaa and Jurisica '05 [52] (LD)	190,572	**5,600**	59	**83.0**	N/A	**68.0**
Cumbaa and Jurisica '05 [52] (kNN)	190,572	**5,600**	59	**85.0**	N/A	**76.0**
Yang et al. '06 [53] (Fisher-Based)	110	39	16	80.9	58.3	84.8
Kawabata et al. '08 [54] (LD-Based DT)	439	229	14	92.7	90.1	93.5
Cumbaa and Jurisica '10 [55] (RF, Crystal vs. Clear)	13,830	**1,879**	14,908	**61.6**	N/A	**80.2**
Cumbaa and Jurisica '10 [55] (RF, Precipitate vs. Clear)	9,656	2,897	14,908	**86.4**	**88.7**	N/A

3.3 Protein Crystal Data

Our data set consists of $11,648$ grayscale images of a fixed size of 1024×1024 pixels courtesy of Genentech. All images were manually inspected and categorized by multiple human labelers into three categories: category; no crystal (hereafter clear), category-2; potential (hereafter precipitate), and category-3; hit (hereafter crystal). The consensus of the categorizations by these readers are used as the ground-truth labelings in our dataset. Figure 3.2 shows some examples of these images. Note that some crystal and precipitate images may appear similar to those in other classes, and so manual labeling can have some errors; our manual categorizations had approximately 80% agreement across labelers. Also, there exists large within-class variance in image appearance for crystal and precipitate classes,

suggesting that these classes may include sub-classes. However, we chose three categories in our labeling for its ease of manual collection in our high-throughput setting. The distribution of categories in our data is very skewed, which is representative of the true trial data at large. Table 3.2 shows the distribution.

Table 3.2: Distribution of categories in our dataset

Type	Count	%
1	3740	32.11
2	7507	64.45
3	401	3.44

Incidences of crystals are both very rare and highly valuable, and so a classifier which minimizes type II errors (i.e., false negatives) is extremely desirable. To summarize, the existence of the following in the dataset present the main challenges: i) very few crystal cases, ii) much variation in crystals' shape, appearance, and location, iii) illumination variation, iv) artifacts, and v) labeling uncertainty.

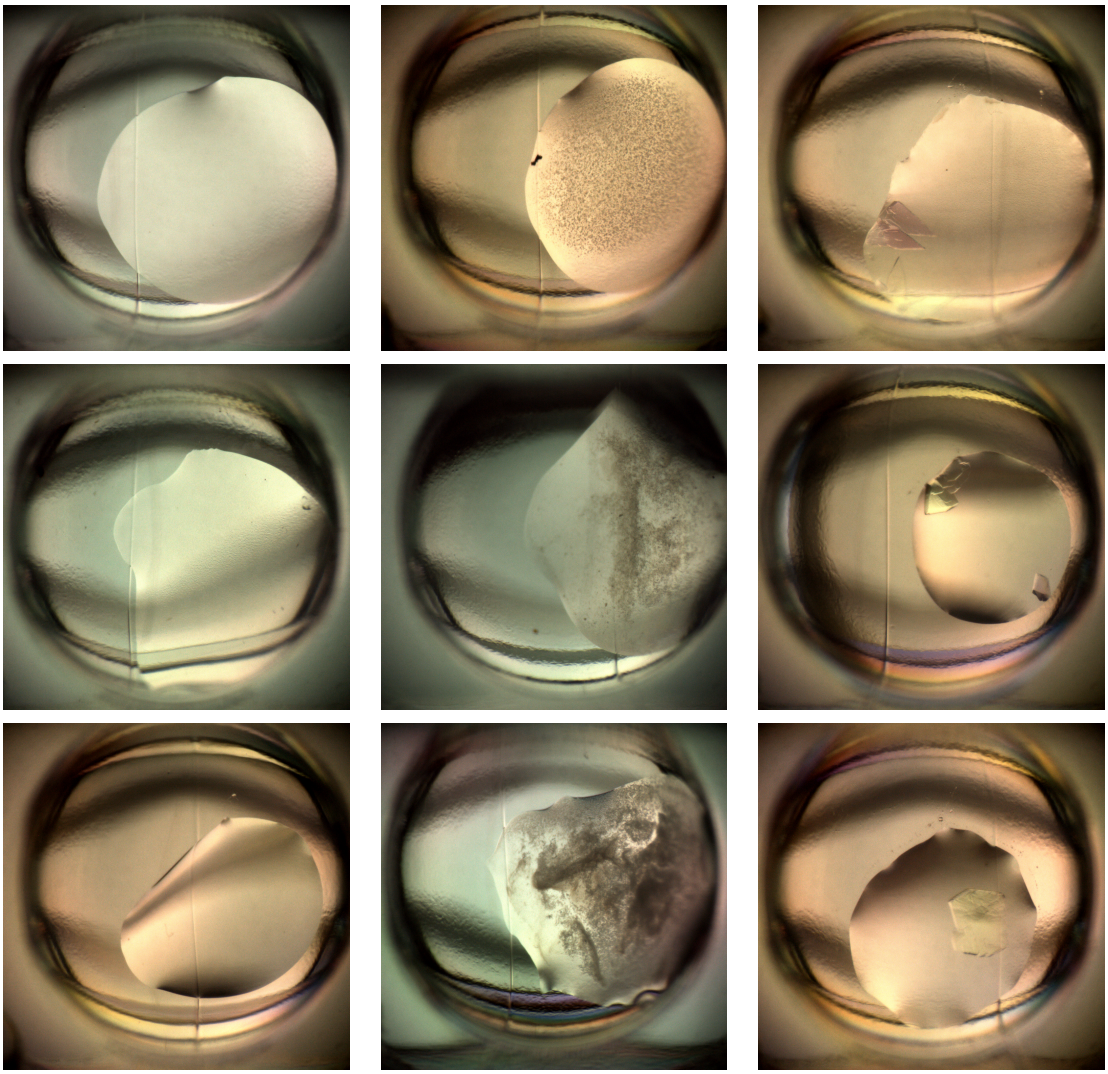

Figure 3.2: Some example images. Left: no crystal, middle: precipitate, right: crystal

3.4 Feature Set Design

Our feature set is designed to cover an appropriate generality of image patterns as well as types. Since the images' circular wells displayed a uniformity of size and position, a region of interest (ROI) was defined with a center $(\frac{W}{2}, \frac{W}{2})$ and a diameter of $\frac{7}{10} \times W$ where W is the image width, in order to ignore pixels outside the well. Within the ROI for each image, we extract the following feature types: 1) Gabor wavelet features, 2) marginal Gabor features, 3) GLCM texture features, 4) shape context features. There are $K = 83,835$ feature values in our design, distributed among the types as shown in Table 3.3. In the rest of this section we'll discuss the feature types.

Table 3.3: The type and number of features

Type	**Count**
Gabor wavelet	63,240
Gabor marginals: scale	12,648
Gabor marginals: orientation	7,905
GLCM	22
Shape context	20
Total	83,835

3.4.1 Gabor Wavelet Features

A 2D Gabor filter takes the form of an oriented complex plane wave enveloped by a Gaussian. The Gabor wavelet transform is a spatially localized Fourier transform, realized by convolving an image with a bank of such filters sampled over various orientations and fre-

quencies. At each pixel, this results in a set of complex-valued wavelet coefficients. The vector of their magnitudes encodes local image appearance for varying edge orientations and frequencies and is used as our local image feature. This feature has successfully been applied in many computer vision tasks, such as face recognition [47]. For efficient implementation, the convolution theorem is exploited so that the time- consuming convolution is avoided by 1) transforming an image using the Fast Fourier transform (FFT), 2) multiplying the transformed image with the Fourier image of Gabor filters in frequency domain, 3) transforming the result by applying the inverse FFT. In our study we use a total of 40 filters comprised of 8 orientations and 5 frequency scales. We extract a 40 dimensional feature vector at locations sampled every 16 pixels inside the ROI, resulting in 1,581 pixel locations, thus the total of 63,240 values per image. Figure 3.3 shows the set of Gabor wavelet responses for an example image.

3.4.2 Marginal Gabor Features

Since crystals appear in different shapes, sizes, and orientations, it is prudent to look at marginal distributions of the Gabor wavelet features over scales and orientations, summing over respective dimensions in the array of responses shown in Figure 3.3. As a result, the marginals-over-scale features consist of 8 values for the 8 orientations (summing images vertically in Figure 3.3) and the marginals-over- orientation features include 5 values for

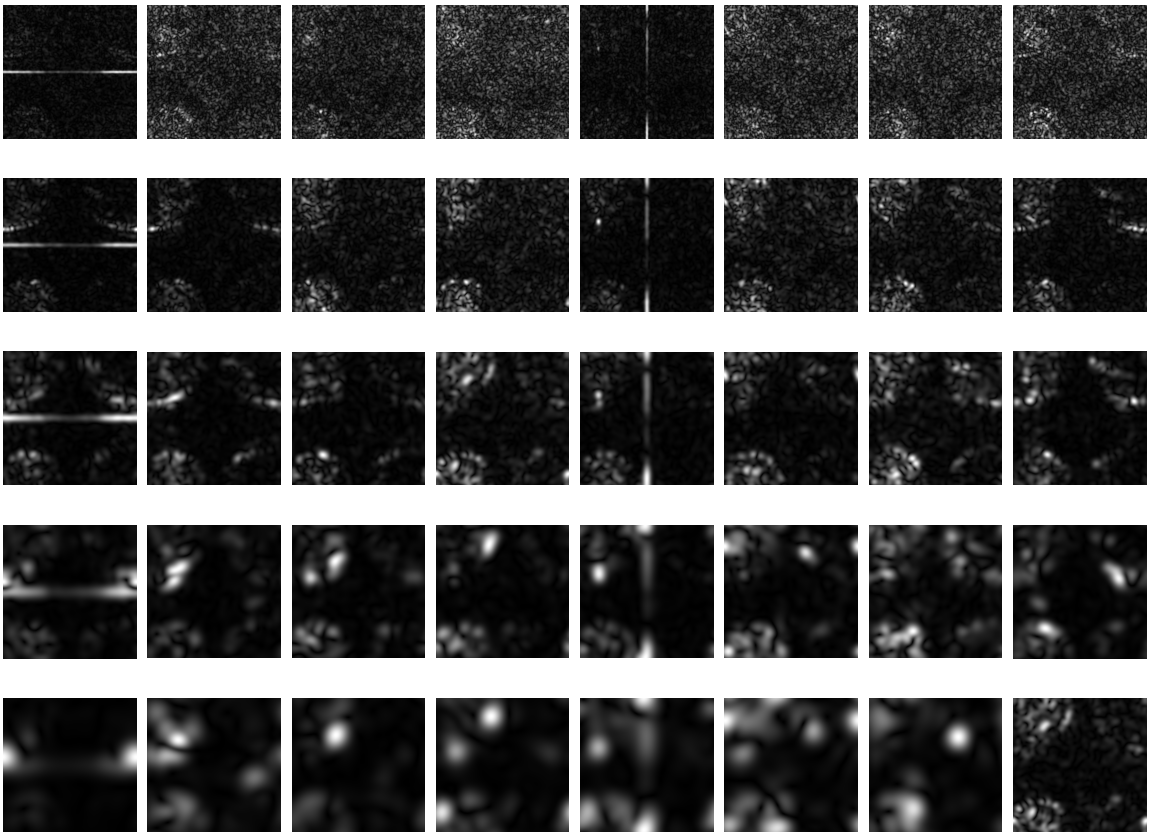

Figure 3.3: Gabor wavelet responses of an example image for 5 scales (vertical axis) and 8 orientations (horizontal axis)

the 5 scales (summing the same images horizontally). This is performed for each of 1,581 Gabor wavelet features, resulting in 12,648 marginals-over-scale and 7,905 marginals-over-orientation features.

3.4.3 GLCM Texture Features

GLCM [48] is a popular texture descriptor for grayscale images based on intensity co-occurrences of pixels placed at certain distances. Given a $N_x \times N_y$ image I with Q gray levels, a GLCM is defined as a set of $Q \times Q$ matrices M_d parameterized by a spatial offset (dx, dy)

$$M_d(i,j) = \sum_{x=1}^{N_x} \sum_{y=1}^{N_y} f(x,y,i,j,I)$$

where $f(x,y,i,j,I) = 1$, if $I(x,y) = i$ such that $I(x+dx, y+dy) = j$ and $d = \sqrt{dx^2 + dy^2}$, else 0. We compute a list of 22 texture features from GLCM computed for the ROI of each image: autocorrelation [56], contrast [56], correlation$_1$ [57], correlation$_2$ [48], cluster prominence [56], cluster shade [56], dissimilarity [56], energy [56], entropy [48], homogeneity$_1$ [58], homogeneity$_2$ [56], maximum probability [56], variance [48], sum average [48], sum variance [48], sum entropy [48], difference variance [48], difference entropy [48], information$_1$ of correlation [48], information$_2$ of correlation [48], inverse difference [57], and inverse difference moment [57].

3.4.4 Shape Context Features

Shape context [49, 59] is a popular scale invariant shape descriptor for various object recognition tasks. Given a point set $\{\mathbf{p}_i\}$ sampled along a target object's boundary in a 2D image, a shape context descriptor at each boundary point $\mathbf{p}_i$ is derived as a 2D histogram of the other boundary points $\{\mathbf{p}_j | \mathbf{p}_j \neq \mathbf{p}_i\}$ with the log-polar bins centered at the point $\mathbf{p}_i$ and size-normalized by the mean distance of the boundary points. In our protein crystal detection application, boundary segmentation of target crystals is not readily possible due to their radically varying appearances. We adopt results of the Canny edge detector as our boundary points; our images have up to 100 boundary points. We then create a composite shape context feature by averaging 2D histograms computed at these edge points with fixed-size 5 orientations $\times$ 4 scale log polar bins, resulting in 20 scale-invariant geometric features. The implementation used here was adapted by a colleague, Oliver Newland [60].

3.5 Classifier Design

Here we will describe the different classifiers we used.

3.5.1 Protein Crystal Image Classifiers

Four main supervised classification models are considered in our application context: naïve Bayes (NB) [37], linear discriminant analysis (LD) [37], random forest (RF) [46], and elastic net (EN) [45]. Added to this set were a number of cascade classifiers, which are compositions of those above: LD- RF, RF-LD, NB-RF, and RF-NB. All classifiers are trained from our dataset for our three-class problem and subjected to 10-fold cross validation for model selection.

The NB classifier is a standard probabilistic classifier with the naïve assumption that features are conditionally independent. We learn the feature-wise likelihood and class prior distributions from our annotated dataset. A NB classifier then outputs the class label that maximizes its posterior probability given a test datum.

The LD classifier is another standard classifier based on a feature space transformation that maximizes the between-class scatter, while concurrently minimizing the within-class scatter. For our 3-class problem, LD yields a 2D subspace onto which test data can be projected for easier classification. An LD classifier then compares distances between the test to the mean for each class in the subspace and outputs the label of the mean closest to the test.

The RF classifier is a popular ensemble classifier based on decision trees (DTs). Bootstrapped samples are first created by randomly sampling a training set with replacement. An RF is built by training a decision tree with each bootstrapped sample, using only a randomly sampled subset (with replacement) of features at each node. Given a test set, each DT generates a classification; a plurality voting of these decisions yields the RF's output. For our dataset, 100 trees and 2000 features are used for constructing our RF models. Increasing the number of trees beyond this default was found to not greatly improve performance and precision maxed out at about the top 2,000 most highly ranked features using RF's innate variable importance; see Figure 3.4. RF's other parameter, the number of features to split on at each node is commonly pegged to the square root or a third of the total number of features and we used the former.

The EN classifier [45] is a generalized linear regression technique, used here in the logistic sense for classification, which efficiently chooses an optimal model which is robust in combining the L^2 regularization of the ridge regression and L^1 regularization of the lasso regression in a weighted combination: $(1-\alpha) \times L^2$-penalty $+ \alpha \times L^1$-penalty with $\alpha \in [0, 1]$. The resulting model with an appropriate normalization inherits the natural built-in feature selection of the lasso method to produce a sparse model while avoiding its shortcomings (e.g., at most n variables can be selected when $p >> n$ where p and n denote the numbers of features and data cases, respectively). To choose the optimal parameters for EN, we ran a grid search on α and λ where α varied between 0.1 and 1.0 in increments of 0.1 and a full 100 point regularization path was built for λ according to the elastic net. In choosing the

best (α, λ) combination we optimized for classification accuracy.

Cascade classifiers have been used in previous studies in this area [54]. In the form used here, these are either of the form X-RF or RF-X. A classifier of the form X-RF is one in which classifier X is used to detect positives and only afterwards is RF used on those data deemed negative by X. A classifier of the form RF-X is one in which RF is first used to detect negatives, and only afterwards is X used to detect positives from the remainder. For our experiments, X takes the form of either NB or LD.

3.5.2 Feature Selection

The number of features we utilize, K, is quite large. It is often beneficial to perform feature selection (FS) to obtain a subset of the K available features to achieve maximal classification accuracy, while easing the joint burdens of time and space complexity. Three data-driven FS methods using RF, LD and EN classifiers are considered in this study. RF comes with a built-in variable importance measure based on ranking either average Gini or information gain increases of decision trees. This ranking can then be used to select the top K_{RF} most important features without rerunning RF training. Fisher's discriminant ratio (FDR) takes each feature on its own and asks how good that feature is at classifying the training set using a straightforward ranking based on the class-restricted data means and variances of

that feature via FDR($feature_I$) = $\frac{(m_i - m_j)^2}{s_i^2 + s_j^2}$ where m_i, m_j and the means and s_i^2, s_j^2 are the variances of the restrictions respectively. EN involves very natural FS via regularization penalties which tends to set the values of non-predictive variables to zero and reduce highly correlated groups of features to a single feature using a combination of ridge and lasso penalties.

3.6 Experiments

This section will describe the experiments conducted in terms of model parameters, computing environments, computational and space constraints and will finish with a exposition of the results of the feature selection and model selection experiments themselves.

3.6.1 Model parameters

For the feature selection of RF, we used out-of-bag estimates to determine the importance of each variable. We built an RF model on the entire data set with 100 trees and 290 (= $\sqrt{p}$ where p is the number of parameters) parameters sampled at each node. For classification models we used 500 trees on the reduced-size data set. The parameters for EN were opti-

mized using a grid search on $\lambda \times \alpha$ as we will describe in section3.6.2. The other classifiers are more primitive and do not require parameter tuning.

3.6.2 Computing environments, software and constraints

Many computing environments were utilized in this study.

In particular, dimensionality reduction was laborious and required an environment with massive amounts of memory and processing power. In memory, the dataset was about 8GB but, depending on the algorithm and implementation, the space required could grow to more than four times that. All feature extraction was done using Matlab [61].

In terms of RF, a failing of Matlab is its computationally inefficient implementation of CART decision trees and hence of RF via its TreeBagger ensemble classifier. An independent Matlab interface (`https://code.google.com/p/randomforest-matlab/`) to the original Fortran code written by by Leo Breiman and Adele Cutler exists which is equivalent to the R implementation [62].

While at Genentech, access to their enormous cluster allowed large scale parallelizations

using R and we used this environment to train the very efficient elastic net models. Here we used R's glmnet package [63], again written in Fortran. However, in the end, most of our classification routines were constructed using python and its deft machine learning tools in scikit-learn [64] running on the excellent infrastructure-as-a- service platform picloud [65].

Matlab took between one and two days to extract the full feature set running serially on a machine with 16GB of memory. A full grid search for elastic net of 9 α's times 100 λ's took less than a day since the glmnet implementation is very efficient at batch calculating the entire regularization path for λ using tuned optimization techniques and a "warm start" [63]. This routine was run in parallel on 9 cores using R's parallelization package SNOW.

In the python / picloud setting for RF, feature selection took a number of hours running serially on a large memory machine. When training classifiers however, RF's trivial parallelization (since trees are independent we could technically train a 100 tree RF on 100 cores in parallel) allowed for very fast classification. In actuality we trained 500 tree forests on 32 compute nodes.

All other techniques, FDR, NB and LD, were trivial from a computational complexity standpoint, as were the cascade classifiers since these results can be computed from their composite prediction label sets.

3.6.3 Feature selection experiments

The three techniques examined for feature selection were FDR, RF and EN. Both FDR and RF rank features and require manual selection to decide on eventual feature set size. They are therefore easily comparable. EN does model selection and feature selection at the same time and returns the non-zero (*model coefficients*, *index*) pairs. To compare the feature selection performance of RF and FDR, we ran 10 fold cross validation for RF on the entire data set using 100 trees and using $\sqrt{p}$ features for splitting, producing a 2 category classification accuracy of 86.27%. The 2 category version was used because of FDR's limitation to classification in two categories and it was assumed that the comparison would generalize. Next, following RF feature ranking, the top K_{RF} features were selected and 10 fold cross validation using RF classification was carried out on the data subset. Concurrently, the same 10 fold cross validation experiments were performed on the top K_{FDR} feature subset with features ranked using FDR. Figure 3.4 shows the results. Clearly 2,000 features is ample to guarantee close to maximal accuracy using the RF method and RF performs much better in this respect than FDR. For this reason we ignore FDR as a viable technique from here on.

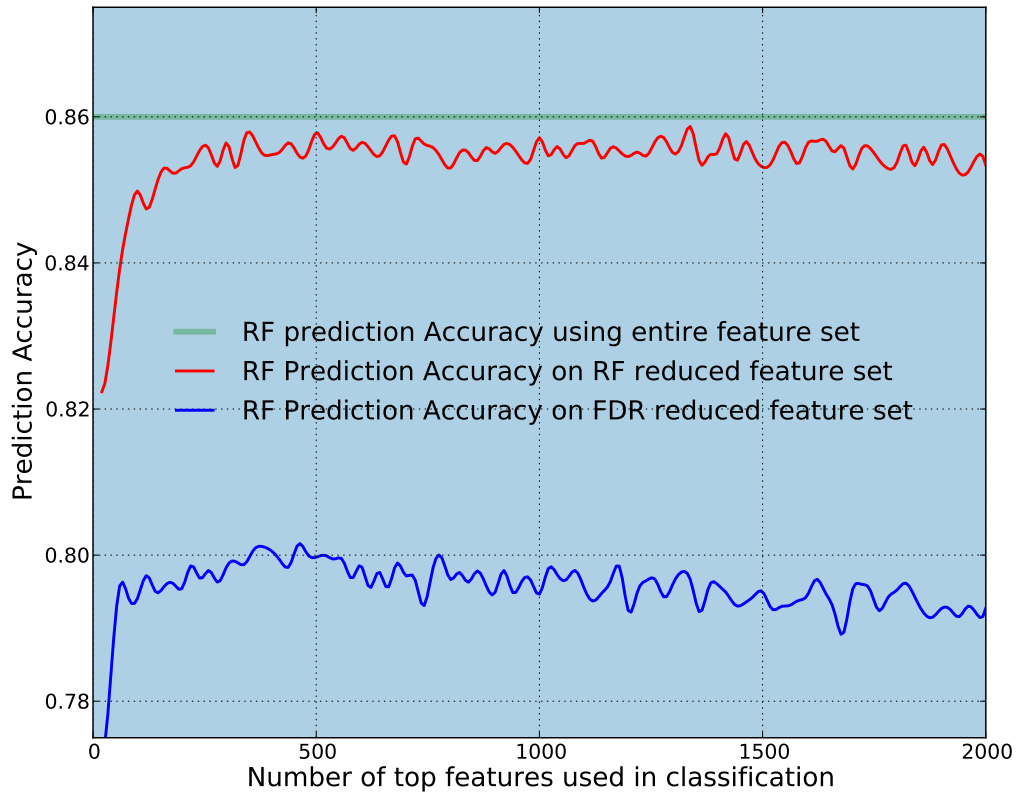

Figure 3.4: RF Prediction Accuracy for top K features selected using RF and FDR

3.6.4 Feature Sources

Selected feature types for each experiment was examined and the results are shown in Figures 3.5 and 3.6. For 2 category feature selection experiments, we can see that of the top features, EN tends to prefer shape context features and this carries for EN to the 3 category case. The features favored by FDR have a similar distribution. In contrast,

RF seems to favor a more even spread of Gabor-type feature. In all cases, shape context features are well represented. There is only one case where a feature from the GLCM set was selected, for EN in the 3 category case.

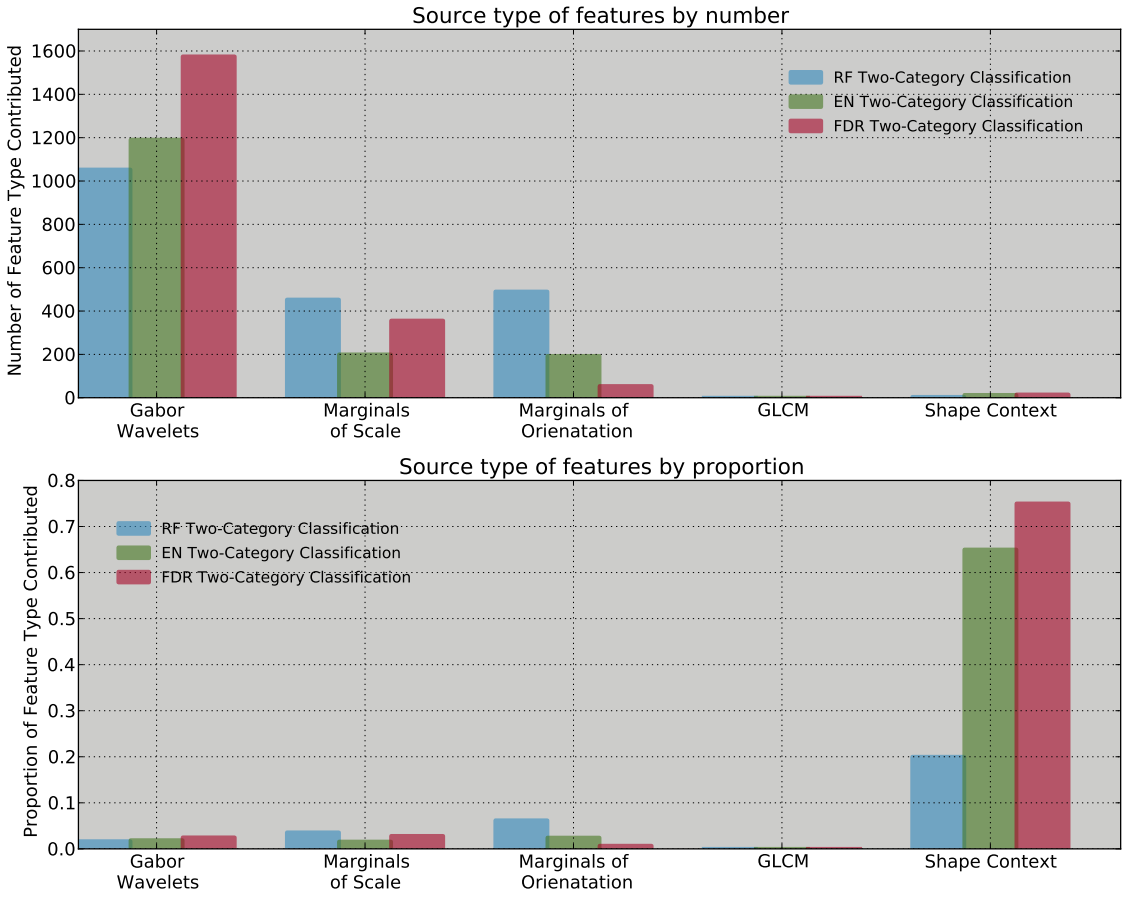

Figure 3.5: Feature Source Distributions for 2 category selection

The breakdown by number presented in Tables 3.4.

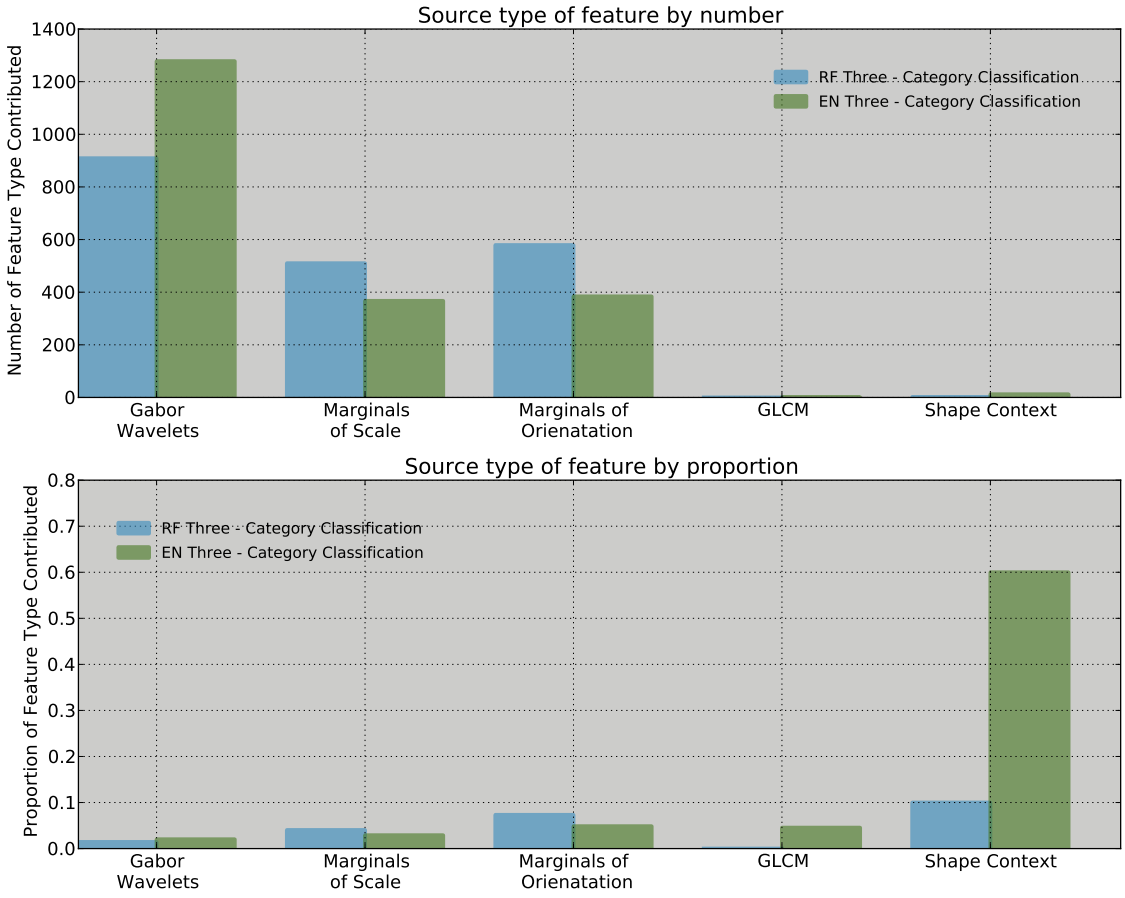

Figure 3.6: Feature Source Distributions for 3 category selection

3.6.5 Model selection experiments

Model selection experiments were carried out under the following conditions. We tried each of the classifier techniques NB, LD and RF on the data subsets reduced by RF and EN feature selection techniques using 2 category models. For the 3 category models, we classified using these techniques and added the cascades NB-RF, RF-NB, LD-RF and RF-LD on the 3 category reduced data subsets. We further considered the EN classifier only

Table 3.4: Distribution of features selected by method

Technique	Gabor Wavelets	Marginals of scale	Marginals of orientation	GLCM	Shape context	**Total count**
RF 2 category	1,053	453	490	0	4	2,000
FDR 2 category	1,575	356	54	0	15	2,000
EN 2 category	1,191	200	193	0	13	1,597
RF 3 category	909	356	54	0	2	2,000
EN 3 category	1,278	367	384	1	12	2,042

on the EN reduced data subsets since this classifier is a logistic model with coefficients trained for such a function. For each model type we carry out 10 fold cross validation for classification and compare the results below using several numeric and tabular metrics, namely prediction accuracy, recall, precision and confusion matrices. Of particular concern from a practical standpoint was the number of crystals falsely identified as clear. We will argue for a more problem-specific metric and discuss its relevance.

3.6.6 Classification and Model Selection Results

We discuss all classification results here. Since we are comparing techniques which have different natural validation methodologies, for example RF's out-of-bag error, we impose standardization by insisting on using 10-fold cross validation for all cases. Since a good classifier in this case is one which is weighted against predicting crystals to be negatives, we look closely at crystal recall, that is the number of correctly identified crystal images divided by the total number of crystal images. We will also examine the crystal precision,

classification accuracy and compare and contrast the confusion matrices themselves.

Table 3.5 shows the classification accuracy for 2 category classification in our studies. Briefly, these results are less interesting in general because they do not involve the identification of crystals. However, it is informative to look at the classifiers conjunctively. RF performs well at classifying the data subset selected using RF and EN performs well at classifying the data subset selected using EN, neither of which should be surprising since those feature subsets are individually optimized for those classification techniques.

Table 3.5: Prediction Accuracy for 2 category Classifiers on both RF and EN reduced datasets

	NB	**LD**	**RF**	**EN**
RF red.	73.33%	74.11%	85.16%	**N/A**
EN red.	77.48%	79.88%	68.3%	84.99%

For classification in the 3 category setting, both RF and EN outperform the others, though some of the cascade classifiers also perform well. As we can see from tables 3.7 and 3.8, this does not tell the entire tale. RF is a good predictor for 2 category classification but fails to predict very many crystals at all, rendering it somewhat useless on its own.

Recognizing this failing, a simple adaptation is to introduce the cascade classifier to try to improve performance by using RF to first classify into 2 categories, clear versus the rest, and then using a more robust classifier like NB or LD to distinguish between precipitate and crystal. We have also implemented cascade classifiers in which we first perform NB or

LD, and then perform RF. In terms of prediction accuracy LD-RF outperforms the other cascade classifiers and rivals the RF and EN classifiers also. As we'll see later in this section, this classifier benefits from the accuracy of the RF classifier, while maintaining the ability to handle the data imbalance.

Table 3.6: Prediction Accuracy for 3 category Classifiers on both RF and EN reduced datasets

	NB	**LD**	**RF**	**EN**	**RF-NB**	**NB-RF**	**RF-LD**	**LD-RF**
RF red.	57.13%	68.54%	82.02%	**N/A**	58.25%	68.58%	73.45%	79.52%
EN red.	66.53%	73.09%	81.67%	82.0%	66.95%	73.89%	75.94%	80.09%

Precision is the number of true positives divided by the number of predicted positives. This is the bottom right item in the confusion matrix divided by the sum of the rightmost column. Prediction Accuracy is simply the number of correct classifications made as a percentage of the total number of predictions . Obviously Percentage Error = 100 - Prediction Accuracy. In terms of a confusion matrix, correct classifications are on the main diagonal and therefore an ideal result is a diagonal confusion matrix.

Precision is the number of true positives divided by the number of positives. This is the bottom right item of the confusion matrix divided by the sum of the rightmost column.

$$\text{Precision} = \frac{\text{Number of True Positives}}{\text{Number of Predicted Positives}}$$

Recall is the number of true positives divided by the number of actual positives. This is

the bottom right item of the confusion matrix divided by the sum of the bottom row.

$$\text{Recall} = \frac{\text{Number of True Positives}}{\text{Number of Actual Positives}}$$

As we can see from Tables 3.7 and 3.8, classifiers incorporating NB achieve good scores for recall. These classifiers are more apt than the others to classify positives, leading to a large number of false positives. Moreover, such classifiers have comparatively poor classification accuracy rates. RF's big failing as a classifier is exposed by its almost null recall but we achieved much improvement by augmenting it with a another classifier. In particular, the LD-RF cascade classifiers achieve good accuracy and a decent trade-off between precision and recall. EN also shows poor recall but performs admirably otherwise.

Minimization of false negatives is a particular tenet of this study and so we examine these numbers in Table 3.9. In this case, EN performs better than the other classifiers, with NB-RF and LD-RF doing well also.

Table 3.7: Confusion Matrix Comparison on RF Reduced Dataset

	2 category	**3 category**	**precision**	**recall**
NB	$\begin{bmatrix} 3,287 & 453 \\ 2,653 & 5,255 \end{bmatrix}$	$\begin{bmatrix} 3,309 & 307 & 124 \\ 2,636 & 3,066 & 1,805 \\ 40 & 81 & 280 \end{bmatrix}$	12.68%	69.83%
LD	$\begin{bmatrix} 2,350 & 1,390 \\ 1,626 & 6,282 \end{bmatrix}$	$\begin{bmatrix} 2,391 & 1,131 & 18 \\ 1,626 & 5,507 & 374 \\ 35 & 280 & 86 \end{bmatrix}$	17.99%	21.45%
RF	$\begin{bmatrix} 2,962 & 778 \\ 950 & 6,958 \end{bmatrix}$	$\begin{bmatrix} 3,010 & 730 & 0 \\ 961 & 6,542 & 4 \\ 13 & 386 & 2 \end{bmatrix}$	33.33%	0.5%
RF-NB Cascade	**N/A**	$\begin{bmatrix} 3,462 & 210 & 68 \\ 2,668 & 3,049 & 1,790 \\ 46 & 81 & 274 \end{bmatrix}$	12.85%	66.33%
NB-RF Cascade	**N/A**	$\begin{bmatrix} 2,954 & 662 & 124 \\ 946 & 4,754 & 1,807 \\ 7 & 114 & 280 \end{bmatrix}$	12.66%	69.83%
RF-LD Cascade	**N/A**	$\begin{bmatrix} 3,499 & 427 & 14 \\ 1,963 & 5,171 & 373 \\ 41 & 274 & 86 \end{bmatrix}$	18.18%	21.45%
LD-RF Cascade	**N/A**	$\begin{bmatrix} 3,006 & 716 & 18 \\ 960 & 6,171 & 376 \\ 13 & 302 & 86 \end{bmatrix}$	17.92%	21.45%

Table 3.8: Confusion Matrix Comparison on EN Reduced Dataset

	2 category	**3 category**	**precision**	**recall**
NB	$\begin{bmatrix} 3,405 & 335 \\ 2,288 & 5,620 \end{bmatrix}$	$\begin{bmatrix} 3,389 & 252 & 99 \\ 2,283 & 4,061 & 1,163 \\ 19 & 104 & 278 \end{bmatrix}$	18.05%	69.33%
LD	$\begin{bmatrix} 2,651 & 1,089 \\ 1,254 & 6,654 \end{bmatrix}$	$\begin{bmatrix} 2,617 & 1,114 & 9 \\ 1,427 & 5,806 & 274 \\ 38 & 272 & 91 \end{bmatrix}$	24.33%	22.69%
RF	$\begin{bmatrix} 606 & 3,154 \\ 558 & 7,350 \end{bmatrix}$	$\begin{bmatrix} 2,991 & 749 & 0 \\ 985 & 6,522 & 0 \\ 11 & 390 & 0 \end{bmatrix}$	0.0%	0.0%
EN	$\begin{bmatrix} 3,051 & 689 \\ 1,051 & 6,849 \end{bmatrix}$	$\begin{bmatrix} 3,122 & 617 & 1 \\ 1,088 & 6,404 & 15 \\ 5 & 375 & 21 \end{bmatrix}$	56.76%	5.24%
RF-NB Cascade	**N/A**	$\begin{bmatrix} 3,502 & 185 & 53 \\ 2,347 & 4,022 & 1,138 \\ 27 & 100 & 274 \end{bmatrix}$	18.7%	68.33%
NB-RF Cascade	**N/A**	$\begin{bmatrix} 2,945 & 696 & 99 \\ 960 & 5,384 & 1,163 \\ 7 & 116 & 278 \end{bmatrix}$	18.05%	69.33%
RF-LD Cascade	**N/A**	$\begin{bmatrix} 3,340 & 394 & 6 \\ 1,818 & 5,417 & 272 \\ 46 & 265 & 90 \end{bmatrix}$	24.46%	22.44%
LD-RF Cascade	**N/A**	$\begin{bmatrix} 2,988 & 743 & 9 \\ 983 & 6,250 & 274 \\ 10 & 300 & 91 \end{bmatrix}$	24.43%	22.69%

Table 3.9: Number of crystals incorrectly classified as clear in 3 category experiments for both RF reduced and EN reduced datasets

	NB	**LD**	**RF**	**EN**	**RF-NB**	**NB-RF**	**RF-LD**	**LD-NB**
RF red.	40	35	13	N/A	46	7	41	13
EN red.	19	38	11	5	27	7	46	10

3.7 Discussion

This study constructed a very large set of features of various types. Feature selection was used to identify a quality subset of features and classification was carried out on the reduced dataset in an attempt to compare and contrast different models. No one of the metrics used in this comparison on its own tells the whole story and many classifiers have a claim to be the best performing one.

For the 2 category problem, the situation is not so ambiguous. RF and EN outperform the other classifiers. One could argue that since EN has a higher false negative rate, in this domain RF is a better classifier.

In the 3 category case, there is more ambiguity. Recall that a major motivator for this work is a suite of techniques for reducing the manual search workload of a group of crystallographers for whom a crystal is considered both rare and valuable. Hence, any automatic process which rules out crystals is a poor system. However, this involves a trade- off between recall and precision since the original goal is to reduce the search space. We would like high accuracy combined with a low false negative rate. One potential use of the models might involve the rejection of anything classified as clear. This amounts to discarding a number of actual crystals in each case, though the number is smallest in the case of EN.

3.7.1 Towards a more relevant metric

Other common metrics used in such tasks are the F-score family of metrics [37] , which are weighted averages of recall [37], precision [37]. However, often a specific problem motivates a specific metric and this is an attempt to detail one suitable for this problem. Ignoring classification accuracy, we have two optimization goals:

1. **Maximize the number of negatives:** Reduce the search space by as much as possible by identifying as many things as possible as clear, to be discarded
2. **Minimize the number of false negatives:** Discard as few crystals as possible

Maximizing the following formula for score S_m achieves this where $\hat{y}$ are the predictions made by model m for the true labels y, $fn_{m,y}$ is the number of false negatives in the predictions made by model m for y, and $n_{\hat{y}}$ is the number of negatives produced by the model m.

$$S_m(\hat{y}) = 1 - \frac{1 + fn_{m,y}}{1 + n_{\hat{y}}} \tag{3.1}$$

As evidence that this score is well defined, consider that the denominator can never be zero since we have added a 1 to account for the case where a classifier is so poor that it classifies nothing as negative. The 1 in the numerator is to make sure that we can achieve a score of 1 if our classification is perfect. Similarly, since the set of false negatives is a subset of the set of negatives, this ratio is always between 0 and 1 inclusive. If this ratio is close to zero then S_m is close to 1. This will occur when we have a very small number of false negatives or when the total number of negatives is very large. When S_m is close to 0, the ratio must be close to 1, meaning that we are predicting almost all negatives falsely; something we'd like to avoid. Either that or we are not ruling very much out, which is again something we wish to avoid. Figure 3.7 shows that the better performing metrics are what we might expect from our previous analyses. Specifically, these include EN and the cascade classifiers which were composed of first a robust classifier to identify positives, followed by RF to distinguish the negatives.

An examination of the pros and cons of this work are given below.

3.7.2 Contributions

Many feature extraction techniques were compared and contrasted. We showed that shape context was a highly effective feature while others, namely Gabor filter types and marginals

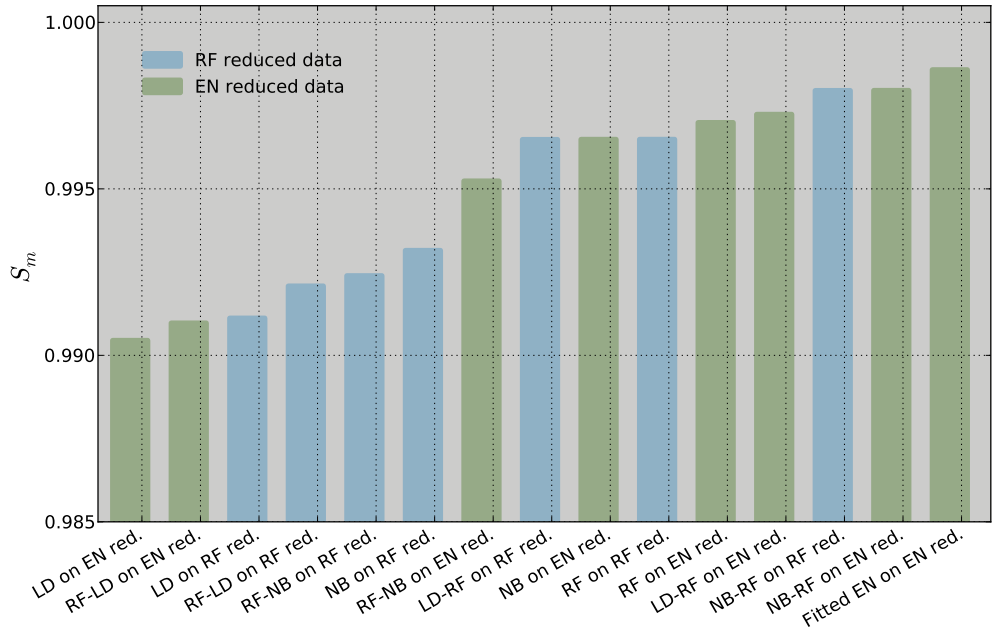

Figure 3.7: Score S_m for each classifier of the 3 category variety. The height of the bars is the score according to the metric S_m detailed above.

of scale and orientation of these were somewhat effective but that the GLCM based texture features were rarely selected. This study deals with a dataset which reflects the data imbalance seen in the real world and so it's assumed the methods detailed here should generalize well enough to be used by other researchers.

Compared with previous studies, we have used sophisticated techniques for choosing features by considering a large number of features and allowing the data to speak for itself using automatic feature selection. Other studies have variously used linear discriminants, decision trees and other primitive classifiers; we introduce some more contemporary and

sophisticated techniques via RF and EN.

Our feature selection experiments combined with our classification results showed that RF and EN feature selection methods were much more successful than the FDR method. We showed that both the RF and EN feature selection methods can easily reduce a huge data set like ours down to about 2,000 relevant features. This greatly enhances manageability by providing memory and performance efficiencies during the research process.

Different model selection metrics were explored and various classification algorithms incorporated. Our results show the complexity of model selection in such a domain given that no model was optimal, though we might make a case that our experiments show EN to be the best option for domain-specific and practicality reasons. Decent clear-versus-the- rest classification results were produced for a robust EN model which can be used to rule out many images and greatly reduce the researchers' search space (by 36%).

Lastly, a metric, see equation 3.1, was identified, the maximization of which was shown to be an effective way to choose between models.

3.7.3 Limitations

- More features should be considered, in order to counteract the tradeoff between recall and precision.
- The imbalance in the dataset was not overcome. This was especially apparent in the case of RF.
- We failed to identify a "best classifier" which excels in all metrics.
- From a practical standpoint, precipitate images are potentially interesting and so a loss function which better reduces precipitate images which are discarded as clear is desirable and was not implemented.

3.7.4 Future Work

- Identifying new features to add to this data might lead to new and interesting results. Color was not used, for example, in any of our feature types.
- Some hope exists for RF if it can be adjusted to deal better with the data imbalance. Potential strategies include weighting of the samples but challenges remain with respect to over-fitting since a model constructed thus is not likely to generalize well.

- Weighting against false negatives should be attempted. Perhaps the maximization of the metric defined above should be incorporated into model and/or feature selection, or into a classifier itself.

- A loss function which incorporates a penalty for precipitate images being classified as clear is desirable from the viewpoint of the researchers motivating this study and has since been attempted [60] with some success.

- Since some of the images are time series there exists the possibility, not explored here, of controlling (by subtraction) for those objects which stay fixed across time and hence reducing the noise level. Again this has been pursued in [60].

Chapter 4

Discussion

In summarizing the outcomes of these studies, it is worthwhile to reflect on the challenges they shared and those characteristics which set them at odds. Following this, we will give a brief synopsis of the different results and finally, we will sum up our contributions.

4.1 Comparative Characteristics and Challenges

These studies have common attributes and each has unique challenges which we will discuss here. In essence these studies are beasts of a different nature, case study 1 being a straight supervised learning project; case study 2 a traditionally unsupervised project to which we have successfully applied, among other things, a supervised technique. More specifically, case study 1 addressed the problem of retrieving images similar to a set of query images using image content only. Case study 2, on the other hand, aimed to model a function capable of automatically classifying crystal images into three categories with a minimal false negative rate.

4.1.1 Level of Difficulty

In terms of difficulty, the case study 2, being supervised in nature and involving images under some very controlled conditions is the more approachable of the two. A wealth of common techniques from the computer vision world could be applied to identify and extract features. Similarly, a wealth of statistical learning techniques are apt to be used on this data set. The real challenge, as we have seen, resides with the data imbalance and with the problem setting. That is, statistical tools perform better with more data and so few positives exist in the training set that challenges abound for any learning algorithm. And

since a motivating goal of this project was to reduce the scientists' search space, the goal of minimizing the number of false negatives is to be especially espoused. Future work should customize algorithms to especially penalize false negatives rather than simply striving for maximal prediction accuracy or precision.

All other things being equal, the problem posed in case study 1 is more difficult. Unsupervised learning techniques are in a more primitive state than their supervised counterparts and generally require *much* more data if they are to achieve similar accuracy. This is especially true in the case of the challenging dataset that is the core of ImageCLEF. That said, the bar of success is problem-specific so that results representing success are much lower than those from similar competitions in the supervised world.

4.1.2 Memory and Computational Challenges

Both of these datasets presented memory and computational challenges which we will compare here.

Memory Constraints

The ImageCLEF-2011 dataset comprised 230,088. We constructed feature vectors with 1,000 dimensions for use in our experiments. This led to an overall data size of over 300 million double precision floating point numbers, for a total in-memory processing size of 1.8GB. So, despite the large number of images, this final data matrix itself was a manageable size. However, after feature extraction and during feature design and development of the 1,000 element codebook, we performed K-means clustering with K = 1,000 on a subset of all SIFT vectors extracted from all images. We clustered 64 million of the total set of 490,453,655 SIFT features, (about 13%), for a total in-memory size of more than 65GB. This step proved technically challenging, requiring a custom piece of software written in C, and parallelized using MPI, in order to satisfy both time and space complexity constraints. In fact, since no machine we had access to could hold all 65GB, and taking advantage of the fact that K- means can be made to parallelize reasonably easily using updates at each iteration, we passed chunks of the data to nodes for processing, as they were read in, so that the memory was divided equally.

Even though case study 2 used about 5% as many images as the *ImageCLEF* dataset, we extracted 83,835 features from each image. We therefore constructed a pre-feature selection data matrix of at most 8GB. A dataset of this size was often unwieldy to manage and much effort was expended in whittling this data down to a manageable size.

Computational Constraints

For case study 1, the most difficult task from a computational standpoint was the large data K-means run. However, our custom implementation severely shortened the run time to something manageable, though a reasonable run time did rely on parallelization via MPI on a large compute cluster. For case study 2, we employed, at different times, personal computers, large memory machines, very large node compute clusters and cloud-based services. Of particular interest was the speedup due to parallelization of RF, since each decision tree can be trained independently. Using R, the very efficient glmnet package combined with the suite of distributed processing tools provided by the SNOW package allowed a very large and computationally intensive set of parameter tuning and model building tasks to run in a reasonable time.

4.1.3 Validation

In terms of validation, the difference between these studies is vast. In the case of case study 1, a public benchmarking dataset is available via ImageCLEF-2011. This competition-based validation was a major motivator. The benchmark publishes a suite of commonly used metrics from the field of Information Retrieval for each competition, of which MAP is given the most weight and was thus used as our metric here. Our novel approach performed

admirably as compared with the other competitors under this benchmark. For the case study 2, no such validation metric exists. We relied on commonly used metrics which are more suited to datasets less heavily imbalanced against positives. Since we presented many techniques and compared them using several of these common metrics, it is not as straightforward to compare our result sets. In another sense, we can be quite confident that the results we *did* show will generalize well to new data since we performed ten fold cross validation in all classification experiments.

4.2 Summary of Study Results

Before discussing the overall contribution of these studies, it is a good idea to refresh the reader to the main results.

4.2.1 Case Study 1

We showed that a metric learning approach to M-CBIR using ITML was advantageous for the unsupervised techniques we tried. That is, a Mahalanobis-type metric learned from the results of the competition performed better in certain cases than the other *a priori* metrics

we tried, and some of the L^1-based *a priori* metrics performed adequately also. Our ITML approach would have placed us 10th of 26 submissions in the 2011 competition.

4.2.2 Case Study 2

We found EN to be the best performing classifier in our studies though our reason for choosing this is somewhat subjective. When classifying more narrowly into category-1 versus categories 2 and 3, we found RF to be the equally good in terms of prediction but RF was a clear winner in terms of predicting fewer false negatives. We examined and were pleased with the performance of several of our cascade classifiers. Here, we recognized RF's good performance at classifying out negatives and augmented that quality with a classifier more apt to identify true positives. Using this technique we found success with RF-NB, RF-LD, NB- RF and LD-RF. Of particular note are the NB-RF and LD-RF classifiers. NB-RF achieved robust results on each of the feature-selected subsets and LD-RF achieved very good classification accuracy. In terms of validation, the fitted logistic EN model, NB-RF in two guises, and LD-RF composed the top four results using the metric S_m introduced in Equation 3.1.

4.3 Contributions and Conclusion

The contribution of case study 1 was to construct a metric learning system, show its validity as an approach to M-CBIR and to run a comparative metric design study using a comprehensive list of similarity measures. For case study 2, we compared and contrasted both feature selection and model selection techniques, showing that some were more suited to operating on an imbalanced dataset than others. Moreover, we introduced a metric that implements the specific goals of the project from a practical standpoint, namely maximization of search space reduction and minimization of false negatives.

It is not easy to weigh the contributions of these studies against each other. Since M-CBIR is a more difficult problem to solve in general, even a relatively low score can be considered reasonable. However, the same competition based evaluation did not exist for case study 2. Compared with previous work, both sets of results stand up well. The novelties of adding and evaluating a metric learning approach in case study 1, of examining so many cascade classifiers and introducing a problem-specific metric in case study 2 are all worthy contributions.

In case study 1, our M-CBIR method helps to progress the field and future applications may see improved ability to better search for images using image content alone. This would

be of tremendous merit to the medical community, to the scientific community and thus to the global community at large. Crystal growth processes are still tedious, and success rates are frustrating and expensive. Our study elucidates the need for better general algorithms to deal with data imbalance and shows that an automatic crystal classification scheme is feasible.

This thesis opened with the statement, "Ours is a time of unprecedented plenty". In terms of data availability, this is assuredly true, and so it goes with image data. It is hoped that the reader is convinced that such data prosperity is exciting and will lead to new and useful applications, and that this thesis has gone some way to furthering two separate fields of current research, applied some well-known, robust and validated techniques in some original ways and yielded some novel and interesting results.

REFERENCES

[1] J. Collins and K. Okada, "A Comparative Study of Similarity Measures for Content-Based Medical Image Retrieval," *Accepted to IEEE EMBC 2013*, 2013.

[2] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon, "Information-theoretic metric learning," in *Proc. Int. Conf. Machine learning*, 2007, pp. 209–216.

[3] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Trans. Pattern Anal. and Machine Intell.*, vol. 22, pp. 1349–1380, 2000.

[4] T. Deserno, S. Antani, and R. Long, "Ontology of Gaps in Content-Based Image Retrieval," *Journal of Digital Imaging*, vol. 22, pp. 202–215, 2009.

[5] H. Müller, N. Michoux, D. Bandon, and A. Geissbuhler, "A review of content-based image retrieval systems in medical applications—clinical benefits and future directions," *Intl. J. Medical Informatics*, vol. 73, pp. 1–23, 2004.

[6] D. G. Lowe, "Distinctive Image Features from Scale-invariant Keypoints," *Int. J. Computer Vision*, vol. 60, pp. 91–110, 2004.

[7] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "Speeded-up robust features (SURF)," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[8] T. S. Lee, "Image representation using 2D Gabor wavelets," *IEEE Trans. Pattern Anal. and Machine Intell.*, vol. 18, pp. 959–971, 1996.

[9] O. Pele and M. Werman, "The Quadratic-Chi Histogram Distance Family," in *Proc. European Conf. Computer Vision*, 2010, vol. 2, pp. 749–762.

[10] Y. Rubner, C. Tomasi, and L. J. Guibas, "A metric for distributions with applications to image databases," in *Proc. Int. Conf. Computer Vision*, 1998, pp. 59–66.

[11] J. Puzicha, J. M. Buhmann, Y. Rubner, and C. Tomasi, "Empirical evaluation of dissimilarity measures for color and texture," in *Proc. Int. Conf. Computer Vision*, 1999, vol. 2, pp. 1165–1172.

[12] T. M. Lehmann, B. B. Wein, J. Dahmen, J. Bredno, F. Vogelsang, and M. Kohnen, "Content-based image retrieval in medical applications: a novel multistep approach," in *SPIE*, M. M. Yeung, B. Yeo, and C. A. Bouman, Eds., 1999, vol. 3972.

[13] P. Clough, H. Müller, and M. Sanderson, "Seven Years of Image Retrieval Evaluation," in *ImageCLEF*, H. Müller, P. Clough, T. Deselaers, B. Caputo, and W. B. Croft, Eds., vol. 32 of *The Information Retrieval Series*. Springer Berlin Heidelberg, 2010.

[14] H. Müller and J. Kalpathy-Cramer, "The Medical Image Retrieval Task," in *ImageCLEF*, vol. 32, pp. 239–257. 2010.

[15] J. Kalpathy-Cramer, S. Bedrick, and W. Hersh, "Relevance Judgments for Image Retrieval Evaluation," in *ImageCLEF*, vol. 32, pp. 63–80. 2010.

[16] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proc. IEEE Int. Conf. Computer Vision*, 1999, vol. 2.

[17] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded Up Robust Features," in *Proc. European Conf. Computer Vision*, A. Leonardis, H. Bischof, and A. Pinz, Eds., vol. 3951 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, 2006.

[18] L. Juan and O. Gwon, "A Comparison of SIFT, PCA-SIFT and SURF," *International Journal of Image Processing (IJIP)*, vol. 3, no. 4, pp. 143–152, 2009.

[19] K. E. A. Van De Sande and T. Gevers, "University of Amsterdam at the Visual Concept Detection and Annotation Tasks," in *ImageCLEF*, H. Müller, P. Clough, T. Deselaers, B. Caputo, and W. B. Croft, Eds., vol. 32 of *The Information Retrieval Series*. Springer Berlin Heidelberg, 2010.

[20] U. Avni, J. Goldberger, and H. Greenspan, "Medical image classification at Tel Aviv and Bar Ilan Universities," in *ImageCLEF*, vol. 32, pp. 435–451. 2010.

[21] H. P. Moravec, "Obstacle Avoidance and Navigation in the Real World by a Seeing Robot Rover," *tech report CMURITR8003 Robotics Institute Carnegie Mellon University doctoral dissertation Stanford University*, vol. PhD, 1980.

[22] C. Harris and M. Stephens, "A combined corner and edge detector," *Proc. Alvey Vision Conf., 1988*, 1988.

[23] G. Monge, *Mémoire sur la théorie des déblais et des remblais*, royale des sciences France, Académie and Royale, I., 1781.

[24] H. Ling and K. Okada, "An Efficient Earth Mover's Distance Algorithm for Robust Histogram Comparison," *IEEE Trans. Pattern Anal. and Machine Intell.*, vol. 29, pp. 840–853, 2007.

[25] H. Ling and K. Okada, "Diffusion Distance for Histogram Comparison," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2006, vol. 1, pp. 246–253.

[26] J. Puzicha, T. Hofmann, and J. M. Buhmann, "Non-parametric similarity measures for unsupervised texture segmentation and image retrieval," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 1997.

[27] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern Recognition*, vol. 29, no. 1, 1996.

[28] D. Geman, S. Geman, C. Graffigne, and P. Dong, "Boundary detection by constrained optimization," *IEEE Trans. Pattern Anal. and Machine Intell.*, vol. 12, no. 7, 1990.

[29] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Qian Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker, "Query by image and video content: the qbic system," *Computer*, vol. 28, no. 9, pp. 23–32, 1995.

[30] H. Müller, P. Clough, T. Deselaeres, and B. Caputo, Eds., *ImageCLEF: Experimental Evaluation in Visual Information Retrieval (The Information Retrieval Series)*, vol. 32, Springer, 2010.

[31] L. Yang and R. Jin, "Distance Metric Learning: A Comprehensive Survey," Tech. Rep., Department of Computer Science and Engineering, Michigan State University, 2006.

[32] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cognitive Neuroscience*, vol. 3, no. 1, 1991.

[33] W. Geng, P. Cosman, J.-H. Baek, C. C. Berry, and W. R. Schafer, "Quantitative Classification and Natural Clustering of Caenorhabditis elegans Behavioral Phenotypes," *Genetics*, vol. 165, no. 3, 2003.

[34] C. E. Shannon, "A mathematical theory of communication," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 5, 2001.

[35] E. P. Xing, A. Y. Ng, M. I. Jordan, and S. Russell, "Distance metric learning with application to clustering with side-information," in *Proc. Advances in Neural Information Processing Systems*, 2003, vol. 15, pp. 505–512.

[36] R. T. Dattola, "FIRST: Flexible Information Retrieval System for Text," *J. Am. Soc. Info. Sci.*, vol. 30, pp. 9–14, 1979.

[37] R. O. Duda, D. G. Stork, and P. E. Hart, *Pattern classification*, Wiley, 2 edition, 2000.

[38] G. Salton, E. A. Fox, and H. Wu, "Extended Boolean Information Retrieval," *Comm. of the ACM*, vol. 26, pp. 1022–1036, 1983.

[39] B. Kulis, M. Sustik, and I. S. Dhillon, "Learning Low-rank Kernel Matrices," in *Int. Conf. on Machine Learning*, 2006, pp. 505–512.

[40] Y. Censor and S. A. Zenios, *Parallel Optimization: Theory, Algorithms, and Applications*, Oxford University Press, 1997.

[41] E. Levina and P. Bickel, "The Earth Mover's distance is the Mallows distance: some insights from statistics," in *Proc. Int. Conf. Computer Vision*, 2001, vol. 2, pp. 251–256.

[42] J. Kalpathy-Cramer, H. Müller, S. Bedrick, I. Eggel, A. G. S. de Herrera, and T. Tsikrika, "Overview of the CLEF 2011 Medical Image Classification and Retrieval Tasks," in *CLEF (Notebook Papers/Labs/Workshop)*, 2011.

[43] S. D. Durbin and G. Feher, "Protein crystallization," *Annu. Rev. Phys. Chem.*, vol. 47, pp. 171–204, 1996.

[44] R. Hui and A. Edwards, "High-throughput protein crystallization," *Journal of Structural Biology*, vol. 142, pp. 154–161, 2003.

[45] H. Zhou and T. Hastie, "Regularization and variable selection via the elastic net," *J. R. Statist. Soc. B*, vol. 67, no. 2, pp. 301–320, 2005.

[46] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[47] K. Okada, J. Steffens, T. Maurer, H. Hong, E. Elagin, H. Neven, and C. von der Malsburg, "The Bochum/USC Face Recognition System and How it Fared in the FERET Phase III Test," in *Face Recognition: From Theory to Applications*, pp. 186–205. 1998.

[48] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," *IEEE trans. Systems, Man and Cybernetics*, vol. SMC-3, no. 6, pp. 610 –621, 1973.

[49] S. Belongie, J. Malik, and J. Puzicha, "Shape context: A new descriptor for shape matching and object recognition," in *Proc. Advances in Neural Information Processing Systems*, 2000, pp. 831–837.

[50] J. Wilson, "Towards the automated evaluation of crystallization trials," *Acta Crystallographica Section D: Biological Crystallography*, vol. 58, no. 11, pp. 1907–1914, 2002.

[51] M. Bern, D. Goldberg, R. C. Stevens, and P. Kuhn, "Automatic classification of protein crystallization images using a curve-tracking algorithm," *Journal of Applied Crystallography*, vol. 37, pp. 279–287, 2004.

[52] I. Cumbaa C., Jurisica, "Automatic classification and pattern discovery in high-throughput protein crystallization trials," *Journal of Structural and Functional Genomics*, vol. 6, pp. 195–202, 2005.

[53] X. Yang, W. Chen, Y. Zheng, and T. Jiang, "Image-based classification for automating protein crystal identification," in *Intelligent Computing in Signal Processing and Pattern Recognition*, D. Huang, K. Li, and G. W. Irwin, Eds., vol. 345 of *Lecture Notes in Control and Information Sciences*, pp. 932–937. Springer Berlin Heidelberg, 2006.

[54] K. Kawabata, K. Saitoh, M. Takahashi, H. Asama, T. Mishima, M. Sugahara, and M. Miyano, "Evaluation of protein crystallization state by sequential image classification," *Sensor Review*, vol. 28, pp. 242–247, 2008.

[55] C. Cumbaa and I. Jurisica, "Protein crystallization analysis on the world community grid," *Journal of Structural and Functional Genomics*, vol. 11, pp. 61–69, 2010.

[56] L. Soh and C. Tsatsoulis, "Texture Analysis of SAR Sea Ice Imagery using Gray Level Co-occurrence Matrices," *IEEE trans. Geoscience and Remote Sensing*, vol. 37, pp. 780–795, 1999.

[57] D. A. Clausi, "An analysis of co-occurrence texture statistics as a function of grey level quantization," *Canadian Journal of Remote Sensing*, vol. 28, no. 1, pp. 45–62, 2002.

[58] Y. Ben Salem and S. Nasri, "Rotation invariant texture classification using support vector machines," in *Proc. Int. Conf. Communications, Computing and Control Applications*, 2011, pp. 1 –6.

[59] S. Belongie and J. Malik, "Matching with shape contexts," in *Proc. IEEE Workshop Content-based Access of Image and Videl Libraries*, 2000, pp. 20–26.

[60] O. Newland, *Highly Loss-Sensitive Classification of Protein Crystallization Images*, San Francisco State University, 2013.

[61] MATLAB, *version 7.10.0 (R2010a)*, The MathWorks Inc., Natick, Massachusetts, 2010.

[62] A. Liaw and M. Wiener, "Classification and regression by random forest," *R News*, vol. 2, no. 3, pp. 18–22, 2002.

[63] J. Friedman, J. Hastie, and R. Tibshirani, "Regularization paths for generalized linear models via coordinate descent," *Journal of Statistical Software*, vol. 33, no. 1, pp. 1–22, 2010.

[64] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine Learning in Python ," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[65] K. Elkabany, A. Staley, and K. Park, "Picloud - cloud computing for science. simplified," in *In SciPy 2010: Python for Scientific Computing Conference*, 2010.