LOCAL FEATURE BASED PATTERN CLASSIFICATION

- FROM PRINCIPLE TO APPLICATION

by

Sharmin Nilufar

MSc, University of Rajshahi, Bangladesh, 1997

THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

in

MATHEMATICAL, COMPUTER, AND PHYSICAL SCIENCES

(COMPUTER SCIENCE)

THE UNIVERSITY OF NORTHERN BRITISH COLUMBIA

December 2005

Abstract

This thesis demonstrates that local feature based approaches are always more stable than global feature based approaches for pattern classification problems. Guided by the original theory that a regional matching approach is more robust than a national matching approach for two-dimensional pattern classification, this thesis examines the applications of the theory in one-dimensional and two-dimensional pattern classifications. We propose two local feature based approaches for two significant applications of pattern classification, namely start codon prediction and content based image classification. For start codon prediction which is considered as a typical one-dimensional pattern classification problem, we have developed a districted neural network that can be taken as a regional voting version of the conventional neural network. Experiments have been performed on the well known translation initiation sites (TIS) data sets and results have shown significant improvement of prediction accuracy. For two-dimensional pattern classification, we propose differential latent semantic index (DLSI) approach for content based image classification. The feasibility of using local features in the DLSI method is also investigated and an extensive experimental study on a real image database has proved its effectiveness.

TABLE OF CONTENTS

List of Tables

List of Figures

Acknowledgement

This thesis is the result of two years of work in which I have been supported and encouraged by many people. I now have the opportunity to express my gratitude for all of them. The first person I would like to thank is my supervisor Dr. Liang Chen for kindly suggesting the subject and providing me guidance throughout the development of this study. During the last two years I have known Dr. Liang Chen as a supportive and principle-centered person. His interest and integral view on research has made a deep impression on me. I owe him lots of gratitude for his support sympathy, encouragement and systematic guidance throughout the course of my work.

I would also like to express my sincere thanks to the other members of thesis supervisory committee namely, Dr. Charles Brown and Dr. Jianbing Li who took effort to help me with valuable comments and suggestions. This research has been financially supported by the University of Northern British Columbia and Natural Sciences and Engineering Research Council of Canada (NSERC). I thank them all for their confidence in me. I am also grateful to the Chairman of the department of Computer Science at the University of Northern British Columbia for providing me an excellent work environment during the past years. Thanks are also due to the authorities of University of Rajshahi, Bangladesh for the grant of study leave. I sincerely thank my mother and father who helped me to widen my vision and taught me many important things that really matter in life. University of Northern British Columbia gave me the opportunity to meet many good friends; Jia Zeng, Baljeet Malhotra, Ran Zheng, Jeyaprakash, Pruthvi Polam, Tyler Neilson and Kevin James Brammer. I would like to thank them all for

their co-operation, help and encouragement during the research work. Finally I am very grateful to my husband Faruk Golam, for his love, support and patience during my thesis work.

This thesis includes the contents of our previous publications in this research area. (Please refer to the appendix for a list of the publications.) I would like to express my thanks to all the co-authors of those publications.

Chapter I

Introduction

This chapter introduces the motivation and the major contributions of this thesis. The organization of the rest of the thesis is also presented.

1 Motivation

The classification of pattern plays an important role in our lives. In most instances we can say that humans are the best pattern classifiers, yet we do not understand how humans recognize and classify patterns. It is a basic capability of all human beings; when we see an object, we first gather all information about the object and compare its properties and behaviors with the existing knowledge stored in our brain. If we find a proper match, we recognize and classify it.

With developments in technology, computerization has taken a significant place in our daily activities. Pattern classification is an active research area that studies the operation and design of systems that classify patterns in data. Important application areas of pattern classification include image analysis, character recognition, speech analysis, gene recognition, man and machine diagnostics, person identification and industrial inspection. After many years of research, the design of a general purpose machine pattern recognizer/classifier still remains an elusive goal.

In this thesis we discuss typical supervised machine-learning problems for pattern classification applications where the pattern to be classified or recognized is either *one-*

dimensional (1D) or *two-dimensional* (2D). For these problems, it has been shown [CT03a] that a local feature based simple approach, named regional voting, is always more stable and robust than the corresponding global feature based approach, named national voting [CT03a] [CT03b]. Chen & Tokuda [CT05] conjecture that the general local feature based approach is always better than the general global feature based approach; although it has not been proved mathematically. The regional matching approach is similar to the Electoral College where the nation is divided into some regions and the winner is first decided for each pre-divided region. Finally the winner of the whole nation is determined by simple majority of the winning regions using the "winner-take-all" principle within the pre-divided regions. It is contrusted to the simple voting system where the majority voting across the whole nation selects the winner of the nation.

This thesis applies the conjecture, that the general local feature based approach is always more robust than the general global feature based approach, in applications and shows that, at least in some important applications, the conjecture is true. We have considered two important pattern classification problems, namely start codon prediction and content based image classification, and developed local feature based approaches for these problems.

2 Major Contributions

In this thesis we have proposed two pattern classification schemes based on local features and also demonstrated their applications in two different areas of pattern recognition, namely gene recognition and image classification. We have identified these two

classification problems as 1D and 2D pattern classification respectively. The list of the major contributions of this thesis is as follows:

- We have theoretically proved that, for 1D pattern classification, the local feature based districted matching approach is more robust and stable than the global feature based undistricted matching approach.

- We have developed a districted neural network on the idea of regional voting for start codon prediction. Experiments have been performed on the well known translation initiation sites (TIS) data sets. Experimental results have shown significant improvement in the prediction performances.

- We have applied a differential latent semantic index (DLSI) approach based on feature extraction for content based image classification. The effectiveness has been proved by experiments. The novel idea of using local features in addition to global features in DLSI approach has also been proposed to improve the classification accuracy.

- These approaches and experiments also reconfirm the belief that a local feature based approach is always more stable than a global feature based approach, as conjectured by previous reports on theory of voting [CT03a] [CT03b].

3 Overview

The following chapters outline the methods and applications of local-feature based approaches for one-dimensional and two-dimensional pattern classification. Chapter 2

gives an overview of basic concepts of pattern classification and provides literature reviews on local and global feature based pattern classification methods. Chapter 3 describes the methods and applications of the two pattern classification techniques (i.e. districted matching approach and differential latent semantic index). The detailed experimental study and results of both of these approaches are demonstrated in Chapter 4. A summary of this thesis and a discussion of future work are provided in Chapter 5.

Chapter II

Background

This chapter gives a brief overview on some basic concepts of pattern classification scheme. It accentuates the difference between global feature based and local feature based pattern classification schemes. A review of a number of global and local feature based classification methods is also provided. Some general performance measures for the classification techniques are described in the last section.

1 Introduction to Pattern Classification

Pattern recognition/classification methods are used to automatically recognize and classify different kinds of physical objects or abstract multidimensional patterns. Several types of commercial pattern recognition systems exist which can automatically classify fingerprint images, handwritten cursive words, human faces, speech signals, printed text, blood cells, human genes, etc. Most machine vision systems utilize pattern recognition/classification approaches to identify objects for sorting, inspection, and assembly. The design of a pattern classification system requires the development of following modules: (i) sensing, (ii) feature extraction and selection, (iii) decision making, and (iv) system performance evaluation. The availability of powerful personal computers and inexpensive and high resolution sensors has influenced the development of pattern recognition algorithms in new application domains (e.g. bioinformatics, text, image and video retrieval).

Classification of patterns can be performed in either supervised or unsupervised ways. In *supervised classification* the input pattern is classified to be an element of a predefined class defined by the system designer. *Unsupervised classification* classifies and assigns the input pattern to a previously unknown class.

Nowadays, interest in the area of pattern recognition has grown due to many promising applications which are not only challenging but also computationally more demanding. These new applications include bioinformatics, data mining, document and image classification, multimedia retrieval, voice and speech recognition, remote sensing etc.

Pattern classification techniques can be divided into four groups: 1) template matching, 2) statistical classification, 3) syntactic or structural matching, and 4) neural networks. These groups are based on the representation, design style and recognition functions. For example, in template matching, patterns are represented by samples, pixels or curves. In syntactic approaches, pattern primitives are used for representing the pattern. Statistical approaches and neural network methods usually used features to represent pattern. Several classification models can also be employed together to design hybrid systems. An overview of these classification methods is given in [JDM99].

2 Features of Pattern

Features play a significant role in pattern classification approaches. A feature simply represents a measurement on a pattern, or a combination of measurements on a pattern. In a pattern classification method, features are produced by functions of the raw or pre-processed measurement data. These functions are generally called *feature extractors.*

Features can be global or local depending on how they are extracted from the patterns. The pattern classification approaches can be classified into two groups based on the type of features used in classification method: global feature based pattern classification and local feature based pattern classification.

2.1 Global Feature Based Pattern Classification

Most pattern classification systems are likely to use global features. *Global features* describe an entire pattern at a time. Global features have the capability to specify a whole object with a single vector. Consequently, their use in typical classification procedures is easy and straightforward. However, global features are very sensitive to noise. In global feature based methods, patterns are usually represented in high dimensional space and many learning techniques cannot be used efficiently.

Global features, have been used in applications in various fields of pattern classification such as bioinformatics, face recognition [TP91], hand writing recognition and image compression. Global feature based pattern classification is a common technique for finding patterns in data of high dimensionality. For example, in bioinformatics Hua *et al* [HS01] introduced a support vector machine to predict the subcellular localization of proteins. They used global features called amino acid composition which is the fraction of each amino acid in a protein sequence. Pedersen *et al* [PN97] used trained neural networks that use a combination of local start codon context and global sequence information for prediction of translation initiation sites. Features used in this method are the nucleotides (A, C, G and T).

For face recognition, a single feature vector that represents the whole face image is normally used as input to a classifier. Several global classifiers have been proposed in the field of face recognition, (e.g. minimum distance classification in the eigenspace [TP91], Fisher's discriminant analysis, and neural networks [FC90]). It has been observed that these global techniques work well for classifying frontal views of faces. However, these global feature based methods are not robust against simple pose changes since global features are extremely sensitive to translation and rotation of the face.

2.2 Local Feature Based Pattern Classification

Local features represent the part of any pattern. Local features are processed at multiple points in a pattern and are therefore more stable and robust to noise, occlusion and clutter. Local feature based method usually represent pattern by a variable number of feature vectors. As a result, it may require specialized classification algorithm to handle it.

Although several applications based on local feature for pattern classification have been proposed, only a few works have provided detailed theoretical explanations about why local feature based methods are stronger than the global feature based methods. Chen and Tokuda [CT03a] examined the robustness of the local feature based regional matching scheme over global feature based national matching technique. The effect of concentrated noise on a typical decision-making process of a simplified two-candidate voting model was discussed. They proved that the regional matching process using local features is more robust and stable than a national approach using global features.

They established that the local matching scheme is capable of accepting a higher level of noise than the global matching scheme before the result of the decision changed. A shifting strategy which considers all the possible different partitioning of the nation was used for further improvement. In addition to the theoretical analysis, two realistic experimental results are presented. The first experiment described the problem of mixed white-black flag, where one wants to identify it either as a white or a black-dominated flag. This experiment applied the theory directly on a pixel by pixel basis without any kind of features extraction, data compression or dimensionality reduction methods. It showed that, after adding small amount of noise, global voting reversed the results of the original candidate selection while regional voting having different kind of regions size could conserve the original results. It also demonstrated that, as the region size decreases, the stability margin increases. The second experiment was carried out on a face recognition problem. The external noises which occur in face images are typically characterized by clutter or occlusion in imagery. The experimental results showed the superiority of regional voting over national voting for images of lower noise level and also for images of higher noise level. The extension of the original discrete-model-based stability analysis of regional and national voting in the paper [CT03a] was described for more practical continuous model in the paper [CT03b]. The continuous model based analysis reconfirms the earlier conclusion that regional voting with smaller sized regions always demonstrates an improved stability over those with larger sized regions in the presence of both white and concentrated components of noise. This conclusion is still valid in the continuous case providing that the weak distribution assumption is valid [CT03b].

The use of local features is most common in the field of image retrieval and classification [LWW00]. One of the interesting papers, which developed a classification method to classify images and video according to their "type" or "style", used local feature based approaches [Kar03]. The proposed classifier determines the identity of an artist by the style of his/her painting, or detects the activity in a video sequence. It employed local properties of spatial or spatio-temporal blocks of images which were based on the discrete cosine transform (DCT) coefficients. The naive Bayes classifier was used for the learning and classification scheme. This paper attempted to classify every image block first, and then to classify the whole image by a majority vote. The information extracted from this block based process accomplished more than the classification of the entire image. The image was mapped to the different regions, each dominated by a certain style or type. This technique often produces results which are very similar to human perception. This paper showed that the local analysis of the image found more useful information than that present in histogram based approaches, which classified the entire image based on similarity between, for example, cumulative distributions of gradients, or wavelet coefficients. A new similarity measure of images based on region representation was described by Li, Wang and Wiederholdin [LWW00]. An image was represented by a set of regions, generally corresponding to objects, which are described by their local features like color, texture, shape, and location properties in the proposed image retrieval systems. Images were segmented into blocks with 4 x 4 pixels and a feature vector was extracted for each block. These block sizes were chosen to optimize between texture effectiveness and segmentation coarseness. The integrated region matching (IRM) measure for estimating overall similarity between images incorporates properties of all the regions in the images

by a region-matching scheme where one region of an image is matched to several regions of another image. After regions were matched, the similarity measure was computed as a weighted sum of the similarity between region pairs, with weights determined by the matching scheme. The overall similarity approach decreases the effect of imprecise segmentation, helps to clarify the semantics of a particular region, and facilitates a simple querying interface [LWW00].

Local feature based pattern classification also provides the flexibility of using weighting strategies to improve the classification performance [Li03] [AMHW03] [LWW00]. Xuelong Li [Li03] proposed a content based image retrieval algorithm based on running sub-blocks with different similarity weights for image retrieval by movable contents. The algorithm can be extended to retrieve the images with the same content located in different locations. In this approach, the first step was to split the images into different sub-blocks by a dynamic sub-block splitting method based on the size of the query object. The sub-block of the query image was compared with the sub-blocks of the stored images to process histogram matching which facilitates the retrieval of images with the query content located in different areas. Since, different parts in an image contribute difference effects to human vision perception, different weights were assigned to sub-blocks according to their locations. As a result, sub blocks were evaluated not only by their locations in an image but also by simulated perceptions. A similarity matrix was used to measure the similarity between any sub-blocks of the query image and the sampled image. Li claims (in [Li03]) that their algorithm can significantly improve the retrieval performance for certain kinds of objects with little influence of the entire color histogram.

Local features have also found application in face recognition. Several approaches

showed improved performance of local feature based face recognition over global feature based face recognition [HHWP03] [AMHW03]. Heisele *et al* [HHWP03] demonstrated a local component based method and two global methods for face recognition, and evaluated and compared them with respect to robustness against pose changes. The local component-based method employed a face detector that identified and extracted local components of the face. The face detector was designed by a set of Support Vector Machine (SVM) classifiers that detect learned local facial components and a single geometrical classifier. The detected facial components were obtained from the image, normalized in size, and applied as inputs to the SVM classifiers. In addition to local approaches, two global systems were designed to recognize faces by classifying a single feature vector containing the gray values of the whole face image. A single SVM classifier was trained for each person in the database in the first global system. In the second system the images of each person were divided into view-specific clusters. View-specific SVM classifiers were trained on each single cluster. The testing database contained a wide variety of faces rotated up to about 40 degrees. Although the component-based face detector was computationally more expensive than the global face detector, the robustness of the local component based method on the both global systems was clearly shown.

The ability of using different kinds of voting techniques along with local features also makes the local feature based method attractive for different classification purposes. Artiklar *et al* [AMHW03] proposed a local voting network for human face recognition. A template matching-based classifier system, which employed local features called local distance computations and a voting scheme was used for facial image classification. The system has the ability to reject unknown patterns, and provides invariance over small

amounts of translation. The shifting process was used to compute the (local) distance between an input window and a database window. In addition to standard voting methods, they also investigated the feasibility of a weighted voting system. In the local voting method, the local window received only a single vote for the most similar image and all other images did not receive any votes. The opinion here is that the correct person may not be first on the list, but may be second or third. A weighted or fuzzy-approach to voting was also proposed where the vote was not cast as a 0-1 binary decision. Instead, the vote was cast as a real number in the interval [0, 1]. Although the performance of the both methods was nearly the same, the weighted voting offered more flexibility in the sense that there were more operating points to choose from.

3 Performance Measures

In practice, the performance measures of a pattern classification system must be estimated from all the available samples which are split into training and test sets. The classifier is first designed using training samples, and then it is evaluated based on its classification performance on the test samples. The percentage of correctly classified test samples is taken as an estimate of the performance of the classifier. It is very important that the training set and the test set should be sufficiently large and independent for reliable and accurate performance measure. In fact this requirement of independent training and test samples is still often ignored [Bao].

3.1 Overall Accuracy

The performance of a classification task has traditionally been measured by the overall accuracy. *Accuracy* represents the overall correctness of the classifier and the overall *error rate* reflects $(1 - Accuracy)$. Each time a classifier is presented with a case, it makes a decision about the appropriate class for the case. The overall accuracy can be defined as the ratio of the number of correctly predicted cases to the number of cases examined:

$$Accuracy = \frac{number\ of\ correctly\ predicted\ cases}{total\ number\ of\ cases}$$

However, the overall accuracy (or percentage classified correctly) does not provide an exact insight into how well the classifier is performing for each of the different classes. In general, a classifier might perform well for a specific class, which accounts for a large amount of the test data. This will bias the overall accuracy, in spite of low class accuracy for other classes. Thus, it is imperative to consider the individual class accuracy to avoid this kind of bias while assessing the accuracy of a classifier. One of the interesting measures for performance analysis is the Matthews correlation coefficient (MCC) when there are only two classes to be classified [Mat75].

3.2 Matthews Correlation Coefficient (MCC)

Matthews correlation coefficient is an efficient and straightforward way to evaluate performance. Two-class classification problems are frequently used in our daily life. The options are restricted to predict the occurrence or non-occurrence of a single case or hypothesis with two-class classification [Bao]. In these circumstances, the two possi-

ble errors are frequently used: false positives or false negatives. Table 1 illustrates the four possibilities for two-class classification problem where a particular prediction rule is employed [Bao].

Table 1: Two Class Classification Performance

	Class Positive(C+)	Class Negative(C-)
Prediction Positive(R+)	True Positives (TP)	False Positives(FP)
Prediction Negative (R-)	False Negatives (FN)	True Negatives(TN)

We can define MCC and accuracy as follows:

$$MCC = \frac{TP.TN - FN.FP}{\sqrt{(TN+FN)(TN+FP)(TP+FN)(TP+FP)}}$$

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} = \frac{TP+TN}{(C+)+(C-)}$$

The ability of a classifier to detect "true positives" is called as *sensitivity* and is defined as follows:

$$Sensivity = \frac{TP}{TP+FN} = \frac{TP}{C+}$$

The ability of a classifier to avoid "false positives" is called as *specificity* and is defined as follows:

$$Specificity = \frac{TN}{FP+TN} = \frac{TN}{C-}$$

Usually MCC, sensitivity and specifivity are very effective performance measures for the applications when the number of "yes/positive" and number of "no/negative" samples differ greatly.

Chapter III

Methodology

This chapter introduces the methods of local feature based pattern classification approaches. Basic definitions and main theorems for local feature based districted matching approach and the basis of employing districted matching approach for the recognition of translation initiation sites in mRNA sequences are discussed in detail. It also provides the overview of DLSI and proposes its application in image classification problems. The use of local features in DLSI is also proposed.

1 Local Feature Based Approach for 1D Pattern Classification

1.1 Districted Matching Approach

The *districted matching* approach can be defined as a local feature based method where any one-dimensional pattern to be classified is first divided into a number of regions and features are extracted locally from each region. Then, each region is classified based on the local features extracted from it. Finally, the whole 1D pattern can be classified according to a type of voting.

1.1.1 Definition and Model

We can describe the districted matching approach by considering a simple two candidate voting problem where there are only two candidates, A and B, and each voter can vote for only one candidate. Let α and β denote the percentages of the (total) votes

which candidates A and B get from the whole interval in the absence of noise. Now, we suppose that the voting is carried out on an interval $[0, N-1]$ which consists of N unit cells, each having exactly one vote to exercise. Without losing generality, we assume $\alpha + \beta = 1$ and $\alpha > \beta$.

The *undistricted voting* system (also called national voting for 2D cases in [CT03a]) is defined as the voting system where the entire population N of the nation votes either for candidate A or candidate B. A candidate wins *if and only if* he/she gets a majority of the N votes. The *districted voting* system (also called regional voting for 2D cases in [CT03a]) is defined as the voting system where firstly the nation is partitioned into N/r intervals, each of which is called as a region; a population of r cells in each region votes for candidate A or B and a majority of votes determines the winner of the region, and a majority of the N/r winning regions determines the winner for the nation. To simplify the analysis, we further assume that N is divisible by r. We also assume that the two end points of the interval are glued together to form a circle, so that the interval can be partitioned into a total of r different partitions [CNK04].

The definitions of some basic terms used to explain the stability margin of districted and undistricted matching approach are given below.

Definition

- We call a set of noise *anti-A-noise* (or *anti-B-noise*) if all the cells under influence will vote for B (or A) regardless of whether it originally votes for A (or B). The number of the cells under influence is called the *number of noise units.* We consider

two different kinds of noise, namely *white noise* and *concentrated noise*, which will be defined later.

- We call a vote *noise-contaminated* if the vote of a cell happens to undergo a change either from candidate A to B or from candidate B to A under some changes of environmental conditions. The noise-contaminated vote undergoing a change from candidate A to B (or B to A) is especially called *anti-A-noise-contaminated vote* (or *anti-B-noise-contaminated vote*).

- A set of *anti-A-white noise* (*or anti-B-white noise*) is dispersed uniformly over the nation, producing a uniformly distributed *anti-A-noise-contaminated vote* (or *anti-B-noise-contaminated vote*). The "uniform" here means that the number of anti-A-noise-contaminated vote (or anti-B-noise-contaminated vote) is proportional to the difference of numbers of A and B supporters (or A and B supporters) in any reasonably sized area. It is obvious that the union of a set of white noise is also a set of white noise.

- A set of *anti-A-concentrated noise* (or *anti-B-concentrated noise*) is defined as the union of non-overlapped intervals of size n, among which all the cells are under influence of anti-A-noise (or anti-B-noise) and thus the votes for A (or B) will become noise-contaminated votes under the influence of the concentrated noise. The union of these intervals is called a *noise concentrated area*, and n is called *the size of noise blocks*. Intuitively, the white noise is isolated and scattered randomly over discrete "points" of the nation while concentrated noise is distributed over connected, continuous areas which may be randomly distributed across the nation.

- A region is defined to be *anti-A-noise-polluted* (or *anti-B-noise-polluted*) if and only if the conjunction set of the region and the anti-A-noise-concentrated (or anti-B-noise-concentrated) area is not empty.

- In accordance with the above two types of noise, the anti-A-noise-contaminated votes (or anti-B-noise-contaminated votes) comprise the two types depending on the noise type, namely the *anti-A-white-noise-contaminated votes* (or *anti-B-white-noise-contaminated votes*) and *anti-A-concentrated-noise-contaminated votes* (or *anti-B-concentrated-noise-contaminated votes*). Notice that, when both of white noise and concentrated noise coexist, some noise-contaminated votes may belong to both of these two types.

Figure 1 illustrates the relationship between noise concentrated block, noise contaminated cells and noise concentrated area as defined above. In the figure, the size of the nation, N, is 35 cells, is equally divided into 7 regions. Each region, r, contains 5 cells. The size of noise concentrated block, n, is 4 cells. The size of noise concentrated area is $4 \times 4 = 16$ cells. We can also notice that the total number of noise contaminated cells is 20 cells while the total size of noise contaminated region is $6 \times 5 = 30$ cells (6 of the 7 regions are contaminated).

Since we are interested in computing the lower bounds of the voting stability in this case, we only consider the anti-A noise in the analysis. Thus, when we refer to noise, concentrated noise, white noise, or contaminated votes hereinafter, anti-A-noise, anti-A-concentrated-noise, anti-A-white-noise, anti-A-noise-contaminated votes are implied.

We let $\aleph_c$, and $\aleph_w$ denote the number of concentrated-noise contaminated votes, and

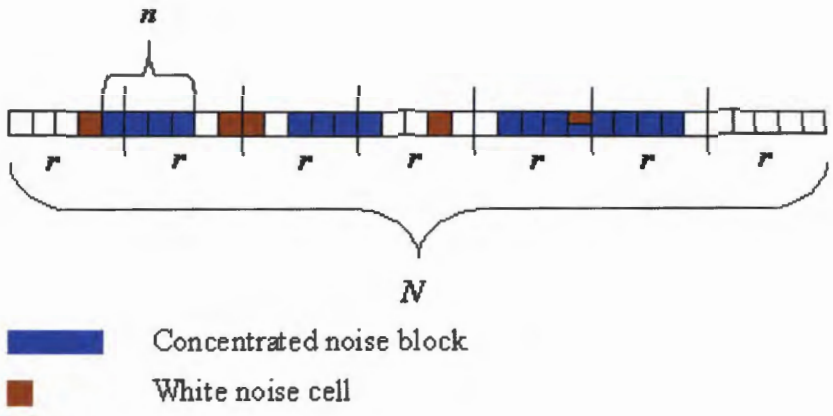

Fig. 1: Nation with noise concentrated blocks, noise contaminated regions and noise contaminated area

the number of white-noise contaminated votes respectively.

In the analysis, we assume that there is only anti-A-noise, as we want to establish a lower bound to a breakdown point in the prevailing situation of $\alpha > \beta$. The result for the districted vote will be established in Theorem 1.2 while the exact bound for the undistricted vote is given in Theorem 1.1.

The following assumption is made in terms of the definitions we introduced.

Assumption *We assume that the size of equally partitioned regions is sufficiently large so that the average distribution assumption holds in each region; where Average Distribution Assumption is defined as, in the absence of noise: the voting distribution of the undisturbed undistricted vote prevails in* any *sufficiently large size areas whether consisting of a continuous part of the nation or of randomly chosen blocks of cells.*

The assumption implies that, in the absence of noise, the global statistical behavior of the ratios of A and B supporters prevail in each of the regions such that there are almost αr cells give vote for A and βr cells give vote for B. We conclude that, if candidate A

(or B) wins in the nation, so does candidate A (or B) in each of the regions.

1.1.2 Main Theorems

Theorem 1.1. *Undistricted voting will preserve the original candidate* A *if*

$$\aleph_c + \aleph_w \times \frac{N - \aleph_c/\alpha}{N} < \frac{\alpha - \beta}{2} N$$

This theorem can be proved by noticing the following two facts:

(1) Among $\aleph_w$ anti-A-white-noise-contaminated votes, $\frac{\aleph_c/\alpha}{N} \times \aleph_w$ votes come from the anti-A-noise-concentrated area.

(2) The undistricted voting is able to preserve the original candidate selection, if and only if the number of overall anti-A-concentrated-noise-contaminated votes is less than $\frac{\alpha-\beta}{2} \times N$.

The detailed proof can be found in [CT05].

Theorem 1.2. *The original candidate selection of the districted voting will be retained if:*

$$\aleph_c < \frac{\frac{n}{r}}{\lceil \frac{n-1}{r} \rceil + 1} \cdot \alpha \cdot N/2 \text{ and } \aleph_w < (\alpha - \beta)/2 \times N.$$

Proof: The proof is similar to the proof of theorem for stability analysis of regional voting in paper [CT05].

Since a concentrated noise block of size n can be partitioned into at most $\lceil \frac{n-1}{r} \rceil + 1$ different regions, we can write,

$$\frac{S_r}{S_c} \leq \left(\left\lceil \frac{n-1}{r} \right\rceil + 1 \right) \frac{r}{n},$$

where S_r is the total size of concentrated noise polluted regions within the nation and S_c denotes the total size of noise concentrated area. We can see that when S_r is less

than half of the total number of regions, districted matching will preserve the original candidate selection in districted voting.

Thus we can write

$$\mid S_c \mid < \frac{\frac{n}{r}}{\lceil \frac{n-1}{r} \rceil + 1} \cdot N/2$$

Substituting the relation of $\aleph_c = \mid S_c \mid \times \alpha$, we get:

$$\aleph_c < \frac{\frac{n}{r}}{\lceil \frac{n-1}{r} \rceil + 1} \cdot \alpha \cdot N/2.$$

On the other hand, when $\aleph_w < (\alpha - \beta)/2 \times N$, the white noise is not enough to reverse the candidate selection in any region that is concentrated noise free [CT05]. Therefore, we have proved the theorem.

Theorem 1.2 shows clearly that to retain the original candidate selection of A in the districted vote, a larger subdivision of the nation, namely partitioning into smaller sized regions, leads to a higher stability, provided that the regions are large enough to hold the Average Distribution Assumption.

To simplify the analysis, we suppose the two ends of the interval are glued together so that there are r different partitionings for partitioning the whole interval into regions of size r. Notice that, for each partitioning partitions the nation into N/r regions, we can see that as a total, these r partitionings partition the nation into N non-equivalent regions, many of which are overlapped, of course. Then, we can define a districted matching scheme for all these N regions rather than N/r regions by one partitioning.

Theorem 1.3. *Generalized districted voting will retain the candidate selection if:*

$$\aleph_c < \frac{n}{r+n-1} \cdot \alpha \cdot N/2 \text{ and } \aleph_w < \frac{\alpha - \beta}{2} N.$$

Theorem 1.3 could be proved by using the technique for proving Theorem 1.2, and the observation that a noise block of size n can pollute at most $n + r - 1$ regions among all the N regions [CT05].

We can refer to a region as Pro-A (A dominated) or Pro-B (B dominated) if A dominates B or B dominates A in that region. Figure 2 illustrates the number of noise-contaminated votes that districted and undistricted voting can accommodate before the original Pro-A decision is reversed. Near equilibrium cases of $\alpha - \beta = 0.02$ are treated in the Figure 2.

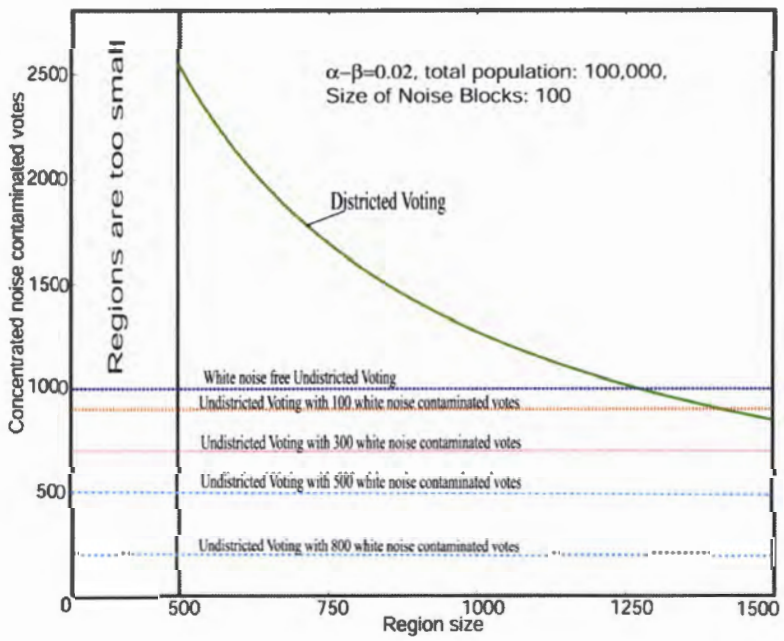

Fig. 2: Numbers of white & concentrated noise contaminated votes a voting system can accommodate

We can see that, as the size of subdivided regions decreases, the number of noise contaminated votes a districted voting can accommodate increases continuously up to a certain point beyond which we could expect that the size of the regions might be too small such that the average distribution assumption might not be valid, although we do not

know a distinct lower boundary of the region sizes for assuring the average distribution assumption.

It may seems that for very large regions with small white noise, the stability margin for the districted voting looks smaller than that of the undistricted voting for concentrated noise. This is due to the upper ceiling operation we have adopted in the analysis where we have regarded all of the concentrated noise polluted regions as Pro-B regions in counting the winning regions by districted voting, so that only the regions that remain entirely *free of noise contamination* are counted to remain Pro-A. In fact, many of the Pro-B transformed regions still remain Pro-A even in the presence of concentrated noise. Evidently this is most serious when the size of regions is large. We can see that, if the size of regions is close to the size of the nation, the districted voting will be close to undistricted voting again. This implies that if the effect of over-estimation of the concentrated noise polluted regions is properly taken into account the stability margin will increase so that the curve representing districted voting close to the right end of figure 2 will move slightly up. Thus, we conclude that the districted voting is always more stable than the undistricted voting as long as the size of regions is large enough to hold Average Distribution Assumption even if the region size is very large; and the districted voting and the undistricted voting will become identical, when the size of regions is so large as that of the nation.

Comparing with the conclusion in [CT03a], where the stability margin for districted voting of 2D pattern is $\aleph_c < (\frac{\frac{n}{r}}{\lceil \frac{n-1}{r} \rceil + 1})^2 \cdot \alpha \cdot N/2$, we can see that the improved performance of the districted vote scheme for 1D pattern is even more substantial than that for 2D objects.

1.2 Application

In the previous section we presented a theoretical model of the districted matching approach. The following sections will present a detailed description of the application of the districted matching approach for a practical problem in bioinformatics.

1.2.1 Problem Statement

Living organisms are made up with proteins they produce according to their genetic information. *Deoxyribonucleic acid* or DNA is called the building block of the life. It contains the information the cell needs to synthesize protein and to replicate itself. DNA is made up of four nucleotide bases called: Adenine (A), Guanine (G), Thymine (T) and Cytosine (C). The information in DNA is first passed on to *Ribonucleic Acid* or RNA in the process of *transcription.* In this process, Thymine (T) in DNA is replaced by Uracil (U) in RNA. Then, in *translation*, this information is used to make proteins. The *replication* ability of DNA is represented by the circular arrow around it in Figure 3 [tis]. Conversely, in some viruses RNA is reverse transcribed into DNA, and RNA is able to replicate itself. These situations are described by the dashed arrows. This flow of the genetic information is called the *central dogma* of molecular biology as shown in figure 3.

The initiation of translation almost always occurs at an AUG codon following the ribosome binding site. This AUG codon is sometimes called as *start codon.* After the translation initiation takes place, the ribosome reads the *messenger RNA* or mRNA triplets and a *transfer RNA* (tRNA) molecule transports the proper amino acid to the

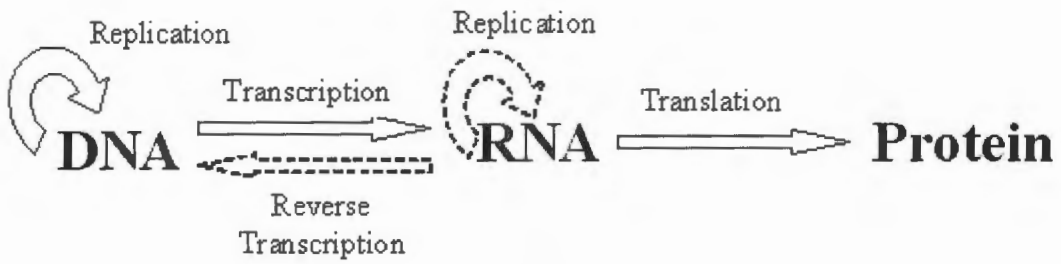

Fig. 3: The central dogma of molecular biology

protein synthesis site. The amino acid is appended to the protein chain, which, by this way, is extended until a *stop codon* is reached [tis].

The recognition of the translation initiation sites (TIS) is important to extract protein sequences from nucleotide sequences. Usually, translation initiation takes place at the first occurrence of AUG codon nearest to the 5-inch end of the mRNA, but in some cases an AUG further downstream is selected. It is observed that less than 10% of all mRNA sequences do not use first AUG as start codon. As a result, if one can obtain complete and error free mRNA sequences then it is possible to predict the translation initiation sites at more than 90% accuracy simply by choosing the first AUG as start codon. Almost 40% of mRNA sequences contains upstream AUGs after extracting annotated GenBank nucleotide data very carefully. The use of unannotated genome data makes the problems of prediction even worse. These errors of sequence analysis cause the prediction of translation initiation sites (TIS) to be a critical task [PN97].

We can consider all these errors as different kinds of noise in a simple two candidate voting system and can model the problem of predicting the translation initiation

sites as a *one-dimensional* (1D) classification problem. It is fairly standard to use a sliding window covering fixed lengths upstream and downstream of an AUG and test if the nucleotide sequence in a window matches to a pattern of start codon. Neural network is one of the very popular approaches for the recognition of start codons, where the inputs are the nucleotide sequences in a sliding window. Taking this method as a general, undistricted matching scheme, we can obtain a districted matching scheme from it: partition the input window into a certain number of sub-windows, use a neural network for each of the sub-windows; then use another neural network whose inputs are the outputs of the previous neural networks for the determination of start codons [CN05].

1.2.2 Implementation

Undistricted Neural Network

A neural network used for the above prediction problem is usually a multi-layer neural network with N inputs, and some (more or less) hidden neurons. We used Pedersen-Nielsen's neural network, shown as NN_U in figure 4. NN_U has 3 layers with some hidden units and 2 outputs. One of the outputs is used for predicting whether the centered AUG of the input is a start codon, while the other is used for predicting whether the centered AUG is a non-start codon. The output of the network is interpreted by believing the output neuron with the highest score. We shall call it an undistricted neural network as opposed to a districted neural network which we will define next.

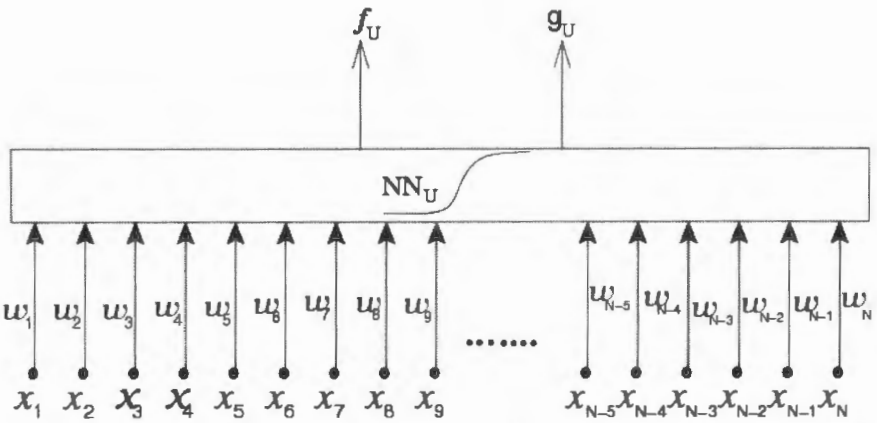

Fig. 4: Undistricted Neural Network

Districted Neural Network

A *districted neural network* is shown in figure 5. A districted neural network constitutes two levels of neural networks: a set of *lower level networks* and a *higher level network*. Each of the lower level neural networks, called a *regional sub-neural network*, takes a block (region) of cells as its inputs; the higher level network, called an *assembling sub-neural network*, takes the outputs of the regional sub-neural networks as inputs. Each of the regional sub-neural networks NN_t $(1 \leq t \leq r)$ uses the same structure as that of an undistricted neural network, but with input $\mathbf{x}^{k_{t-1},k_t}$ which constitutes the cells x_i, for all $k_{t-1} \leq i \leq k_t$ (we let $k_0 = 1$, $k_N = N$). The assembling sub-neural network NN_A has 3 layers with some hidden units, 2 outputs, and $2r$ inputs, where r is the number of total regions.

It is expected that, although it is most likely to come up with much more errors in comparing with the large undistricted neural network, each regional sub-neural network can independently determine a class label for any input array. Of course this may cause many errors because of limited input sizes. But, we can expect that not all the re-

gional sub-neural networks make wrong conclusions at the same time, so the assembling sub-neural network can fuse together these individual regional sub-neural networks to produce a correct answer.

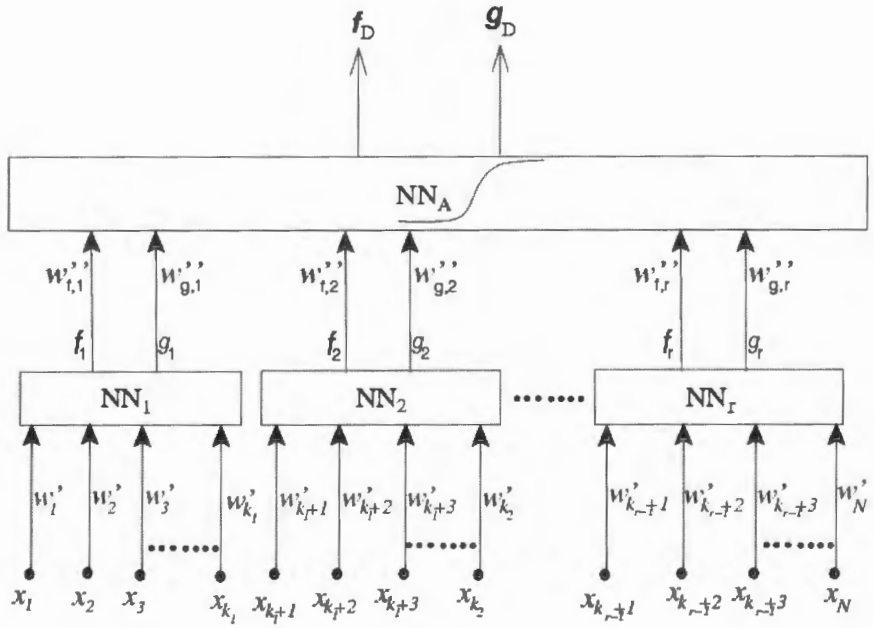

Fig. 5: Districted Neural Network

Training Data Set

Although the training data set used for undistricted and districted neural network is the same, the ways of applying the training dataset in this two kinds of neural network are somewhat different.

Training Data Set for Undistricted Neural Network: We denote the sample set $\mathcal{U}$, of which each sample is of form $(\mathbf{x}^N; (\mathbf{d}_1, \mathbf{d}_2))$, where $(\mathbf{d}_1, \mathbf{d}_2)$ is either $(1, 0)$ or $(0, 1)$ representing $\mathbf{x}^N$ is centered with a start codon or a non-start codon.

Training Data Set for Districted Neural Network: The training set $\mathcal{R}_t$ for

a regional sub-neural network NN_t $(1 \leq u \leq r)$ is constructed as follows: for any $(\mathbf{x}^N; (\mathbf{d}_1, \mathbf{d}_2)) \in \mathcal{U}$, we obtain a sample $(\mathbf{x}^{k_{t-1},k_t}; (\mathbf{d}_1, \mathbf{d}_2))$ for $\mathcal{R}_t$, where $\mathbf{x}^{k_{t-1},k_t}$ constitutes the cells x_i, for all $k_{t-1} \leq i \leq k_t$.

The training set $\mathcal{A}$ for the assembling sub-neural network NN_A is constructed as follows: first, we have each regional sub-neural network NN_t $(1 \leq t \leq r)$ trained using the training set $\mathcal{R}_t$ described above; then, for any $(\mathbf{x}^N; (\mathbf{d}_1, \mathbf{d}_2)) \in \mathcal{U}$, we place a sample $\big(f_{f,1}(\mathbf{x}^{1,k_1}), g_{g,1}(\mathbf{x}^{1,k_1}),\ f_{f,2}(\mathbf{x}^{k_1+1,k_2}), g_{g,2}(\mathbf{x}^{k_1+1,k_2}), \cdots,\ f_{f,r}(\mathbf{x}^{k_{r-1},N}), g_{g,r}(\mathbf{x}^{k_{r-1},N}); (\mathbf{d}_1, \mathbf{d}_2)\big)$ into $\mathcal{A}$, where $\big(f_{f,t}(\mathbf{x}^{k_{t-1},k_t}),\ g_{g,t}(\mathbf{x}^{k_{t-1},k_t})\big)$ $(1 \leq u \leq r)$ is the output of the trained regional sub-neural network NN_t with input $\mathbf{x}^{k_{t-1},k_t}$.

1.3 Discussion

For the districted approach, it is very critical to choose the optimal size of the regions so that it can accommodate the maximum amount of noise contaminated votes. Usually it depends on several other factors. For example, size of noise block, level of white noise, etc. However, these factors are usually application and environment dependent and are not very easy to analyze. Thus, another reasonable solution is to try different sizes of regions to find the best region size (if we don't have enough prior knowledge of the factors involved in the particular application domain) [CT05].

2 Local Feature Based Approach for 2D Pattern Classification

2.1 Differential Latent Semantic Index

This section gives an overview of how Differential Latent Semantic Index (DLSI) works in document classification, on the original papers by Liang Chen, Naoyuki Tokuda and Akira Nagai [CTN01] [CTN03].

DLSI is an efficient method of document indexing that improves the performance as well as the robustness of the document classifier, exploiting both the distances to, and the projections on, a reduced document space [CTN03].

Each document in the DLSI method is represented by the *term by document vector*. A term is defined as a word or a phrase that occurs in at least two documents. Suppose the list of the terms that appear in the documents are $t_1, t_2, ..., t_m$. Each document j in the collection is represented with a real vector $(a_{1j}, a_{2j}, , a_{mj})^T$ with $a_{ij} = f_{ij} \times g_i$, where f_{ij} is the local weight of the term t_i in the document, while g_i is a global weight of t_i across the whole document collection.

2.1.1 Weighting

Weighting strategies play a significant role in the performance of DLSI. *Local weight* of a term represents the significance of the term in the document while the *global weight* indicates the importance of the term applicable throughout the document collections. Several techniques have been proposed for calculating the local and global weights. For example, local weights can be calculated by either raw occurrence counts, boolean, or logarithm of occurrence count. Global weights can be given by no weighting (uniform

weighting), domain specific, or entropy weighting. Although each weighting scheme has its own advantages and disadvantages, experimental outcomes showed that the best approach is to use a logarithmic function for local weighting and entropy for global weighting. The logarithm of the occurrence count reduces large variations between documents and reduces the effects of large differences in frequencies. The entropy weighting scheme assigns higher weights to discriminating terms and lesser weights to terms that carry non-significant information.

2.1.2 Differential Document Matrix and Space

The *Differential document matrix* and the reduced *DLSI space* are the central concepts of DLSI method. The term document vector is normalized as $(b_1, b_2, ..., b_m)$ by the following formula

$$b_i = \frac{a_i}{\sqrt{\sum_{j=1}^{m} {a_{ij}}^2}}$$

The centroid vector $C = (c_1, c_2, ..., c_m)^T$ of a cluster can be calculated in terms of the normalized vector as:

$$c_i = \frac{s_i}{\sqrt{\sum_{j=1}^{m} {s_i}^2}}$$

where $(s_1, s_2, ...s_m)^T$ is a mean vector of the member documents in the cluster.

An *intra differential document vector* T is defined as $T = T_i - T_j$, where T_i and T_j are two normalized document vectors belonging to a same document class; and an *extra differential document vector* T as $T = T_i - T_j$, where T_i and T_j are belonging to different document clusters. The *differential intra- and extra- term by document matrices* are respectively defined by the matrix, each column of which comprises an intra- and extra-

differential document vector respectively.

Using *singular value decomposition* (SVD), any $m \times n$ differential term document matrix D (differential intra- or extra- term by document matrix) of rank $r \leq (q) = min(m, n)$, can be decomposed into a product of three matrices: $D = USV^T$, such that U and V are an $m \times q$ and a $q \times n$ unitary matrices respectively, and the first r columns of U and V are the eigenvectors of DD^T and D^TD respectively. $S = diag(\delta_1, \delta_2, ..., \delta_q)$ where δi are non-negative square roots of eigen values of DD^T , $\delta_i > 0$ for $i \leq r$ and $\delta_i = 0$ for $i > r$.

A new reduced matrix S_k can be obtained by selecting the left-upper corner $k \times k$ matrix ($k < r$) of diagonal matrix S. Similarly U_k and V_k are the matrices obtained by keeping the leftmost k columns of U and V. The product of U_k, S_k and V_k gives a matrix D_k which is approximately equivalent to D. An appropriate value of k must be selected, depending upon the type of application. Generally we choose $k \geq 100$ for $1000 \leq n \leq 3000$. The corresponding k is usually smaller for the differential term by intra-document matrix than that for the differential term by extra-document matrix, because the differential term by extra-document matrix normally has more columns than differential term by intra-document matrix has [CTN03]. SVD produces new reduced intra- and extra-DLSI space. In addition to the global description ability, intra- and extra- DLSI spaces can be effectively used as additive information to improve adaptability to the unique characteristics of the particular differential document vector. On the other hand, SVD is used not only to reduce the dimension of the term by document matrix but also to reduce the noise significantly.

2.1.3 The Posteriori Model

The *likelihood function* $P(x|D)$ of any differential document vector x given D can be estimated by

$$P(x|D) = \frac{n^{1/2} exp(-\frac{n}{2}\sum_{i=1}^{k}\frac{y_i^2}{\delta_i^2}) exp(-\frac{n\varepsilon^2(x)}{2\rho})}{(2\pi)^{n/2}\prod_{i=1}^{k}\delta_i \cdot \rho^{(r-k)/2}}, \quad (1)$$

where $y = U_k^T x$, $\varepsilon^2(x) = \|x\|^2 - \sum_{i=1}^{k} y_i^2$, $\rho = \frac{1}{r-k}\sum_{i=k+1}^{r}\delta_i^2$, and r is the rank of matrix D.

We notice that, the terms $\sum_{i=1}^{k}\frac{y_i^2}{\delta_i^2}$ and $\varepsilon(x)$ respectively describe the projection onto and the distance to a reduced space spanned by the column vectors of U_k from x. If an document I belongs to a cluster centered at C, we could expect $P(x|D_I)$ should be large, and $P(x|D_E)$ should be small, where x is the differential vector of I and C; D_I and D_E are differential intra- or extra- term by document matrices.

When both $P(x|D_I)$ and $P(x|D_E)$ are computed, the *Bayesian posteriori function* can be computed as:

$$P(D_I|x) = \frac{P(x|D_I)P(D_I)}{P(x|D_I)P(D_I) + P(x|D_E)P(D_E)}, \quad (2)$$

where $P(D_I)$ is set to $1/n$, and $P(D_E) = 1 - P(D_I)$, n is number of clusters in the collection.

In this fashion, the Bayesian posteriori likelihood function for the differential document vectors, based on their projections on the DLSI spaces and their distances to the DLSI spaces, provides a most probable similarity measure of a document belonging to a cluster.

2.1.4 LSI and DLSI

It is important to clarify the difference between the DLSI approach and traditional LSI approach. In LSI method the similarity between two documents is calculated by the cosine angle between the projections of a pair of normalized documents vectors in the LSI space [DDL+90]. As described by [CTN01], the cosine measurement of projections of two vectors has the same geometric meaning as the length of the projection of differential document vectors of two vectors in differential latent semantic space. Since LSI is a global dimensionality reduction approach, it has the problem in adapting to particular characteristics of each document. In contrast, DLSI approach takes into consideration the distance to, as well as the projection on, a reduced vector space. Therefore, it is able to capture individual features and much richer information about each document. DLSI has shown improved performance in full text document retrieval and classification compared to the standard LSI based approach.

2.2 Application

We have applied the DLSI approach, which was originally developed for document classification and retrieval, to a new application domain; content based image classification. The new concept of using local features in DLSI has also been proposed here to increase the robustness of the DLSI classifier.

2.2.1 Problem Statement

As digital image and video libraries have become rapidly available to everybody through the Internet, interest in the potential of digital image classification and retrieval has increased enormously over the last few years. Content based image retrieval and classification which retrieve and classify images based on the low-level features, such as color, texture, and shape derived from images, are becoming increasingly active research areas. We study supervised image classification, which is a technique to categorize images based on the available training data [HKZ98]. This topic is so useful for semantic organization of digital libraries and in obtaining automatic annotation of images important for efficient image retrieval systems, that many approaches have already been proposed. Huang *et al* [HKZ98] proposed a method for hierarchical classification of images via supervised learning. They used low level feature-banded color correlograms of the training data to obtain a hierarchical classification tree that can be used to categorize new images. Park *et al* [PLK04] developed a neural network classifier, where the backgrounds are removed from original images and shape-based texture features are extracted for training the neural network. Li, Najmi and Gray [LNG00] proposed a two-dimensional hidden markov model for image classification. Support vector machine (SVM) approaches have also been used in image classification. Goh *et al* [GCC01] combined SVM-based binary classifiers to handle the multiclass image classification problem. However, it seems that no consistently adequate level of performance and user satisfaction have been achieved in this area.

In the area of full text document retrieval and classification, it is noticed that syn-

onymy and polysomy are two major problems that cause surface based retrieval systems to miss important related materials to recall and/or recalling many unrelated documents [CTN01]. Several approaches have been investigated to use the co-relations of terms (words) to weaken the influences of these two problems. We believe co-relationships among features extracted automatically from images should also be taken into consideration in order to improve the performance of image classification.

The LSI approach, which was developed for full document retrieval, and has shown significant effectiveness in content based document retrieval by offering a dampening effect on the synonymy and polysomy problems, has now been applied to image processing by Pecenovic [Pec97], Heisterkamp [Hei02] and Zhao *et al* [ZG02] and proved to be a very promising method for image retrieval. Inspired by the idea of extending the LSI approach from document to image retrieval and classification, we propose the DLSI scheme for image classification. We can model this problem of image classification as a *two-dimensional* (2D) pattern classification.

The major problem of applying DLSI to image classification is that words must be replaced by image features. On the other hand, in order for DLSI to work acceptably, the number of terms must be relatively high. These conditions make the process of feature selection a challenging task. Finally, the idea of occurrence count becomes even more difficult to imagine for image features that usually have numeric values.

2.2.2 Implementation

Features Extraction

The images are represented as vectors of features in content based image classification, and two images are considered to be similar if their feature vectors lay close in the feature vector space. Widely used features include color, texture and shape of objects in the image. For document classification, DLSI represents each document as a term vector, where 'term' can be considered as global feature. In this work, we have used global features as well as local features to construct the feature image vector, as we believe that local features are more robust than global features in the presence of different kinds of noise. Here we first use only color as global features in verifying the efficiency of our DLSI approach in comparing with that of the LSI method and the SVM approach. Then, we extract texture from image blocks as local features. Texture features are used in addition to the color features to improve the accuracy of DLSI based image classification [NCK04].

Global Feature Extraction

For global feature we extract color from the whole image.

Color Features: Color is a very important cue in extracting information from images. Color features are relatively robust and simple to represent. They are invariant to rotation, translation and scaling, but remain sensitive to illumination change and noise. This feature is used in most image classification systems. The distribution of color is a useful feature for image representation. Humans perceive a color as a combination of three basic colors, R (red), G (Green) and B (Blue), which form a color space. Different

color spaces can be generated by separating chromatic and luminance information. To extract color information, a color space must be chosen first. The most commonly used color space is *RGB*. Usually the color image acquisition and recording hardware are designed for this color space. However, the RGB color space model is very complex because of the mutual relation. Thus, it is very difficult to handle the RGB color space in image processing application. On the other hand, *HSV* i.e. H (Hue) S (Saturation) V (Value) color space is more intuitive than RGB color space and very close to human perception. So we have chosen to convert all the images from RGB to HSV color space. As proposed by Zhao and Grosky [ZG02], we extract from each pixel of an image the hue and the saturation values, and quantize them into 10-bin histograms. Then, two histograms are combined into one hue saturation histogram with 100 bins. Consequently, each image is represented with 100 color features.

Local Feature Extraction

For local feature extraction we divide each image into a regular grid of non-overlapping blocks and calculate four important texture descriptors for each image block.

Texture Features: An important set of features for image classification using DLSI technique is the set of texture descriptors. Texture features offer one vital cue for the visual perception and discrimination of image content. Intuitively texture features provide measures of properties such as smoothness, coarseness and regularity.

Although the perception of texture plays a significant role in the visual system for recognition and interpretation, it is quite difficult to adequately model texture. There are several methods to calculate the texture features. A texture descriptor based on a co-

occurrence matrix is quite popular. We can consider the co-occurrence matrix as second order statistical measure of gray level variation which represents the joint probability of gray level occurrence at a certain displacement in an image. The co-occurrence matrices are computed locally within a small window across the image. The choice of window size usually depends on the image and application. The size of the window should be reasonable, so that the extracted information has statistical significance. For the image classification system, we consider four vital texture descriptors which are very easy for humans to differentiate: energy, inertia, entropy and homogeneity.

$$Energy = \sum_i \sum_j p_{ij}$$

$$Inertia = \sum_i \sum_j (i-j)^2 p_{ij}$$

$$Entropy = -\sum_i \sum_j p_{ij} \log p_{ij}$$

$$Homogeneity = \sum_{i,j} \frac{p_{ij}}{1+(i-j)^2}$$

where p_{ij} is an element of the cooccurrence matrix. The entry (i, j) of cooccurrence matrix P_d for an image represents number of occurrences of the pair of gray levels i and j at distance d. *Energy* gives a measure of textural uniformity of an image. The maximum value of energy is obtained when gray level distribution has a constant or a periodic form. *Inertia* which represents image contrast, measures the amount of local variations in an image. *Entropy*, which is inversely proportional to energy, measures the randomness of an image. The entropy of an image is very high when the image is not texturally uniform. *Homogeneity* is inversely proportional to the inertia. It reaches its maximum value when most of the occurrences in gray level co-occurrence matrix are concentrated near

the main diagonal [PCGV02].

Occurrence Count and Weighting

We use the ideas in [Pec97] to find "occurrence counts" for both global and local features such as histogram bins and textures. Basically, the occurrence count O_{ij} of feature i in image j is defined as:

$$O_{ij} = \begin{cases} \left\lceil \frac{val_{ij} - \mu_i}{\sigma_i} \right\rceil & \text{if } val_{ij} \geq \mu_i; \\ 0 & \text{Otherwise} \end{cases}$$

where val_{ij} is the value of the feature i in image j and μ_i and σ_i are the mean and standard deviation respectively of the feature i's value across the training set.

After occurrence count is established, the local and global weighting of features are employed. Following [CTN03] we use logarithmic scaling for local weights and entropies for global weights:

$$f_{ij} = \log(1 + O_{ij}), \quad g_i = 1 - \frac{1}{\log N} \sum_{j=1}^{N} q_{ij} \log q_{ij}$$

where $q_{ij} = \frac{O_{ij}}{d_{ij}}$, O_{ij} is the occurrence count of feature i appear in image j and d_i is the total number of times that feature i appear in the collection, N the number of images in the collection.

Algorithm

Reduced DLSI Space Selection and Classification System Set Up:

(1) Extract numerical features of the images in the training image set.

(2) Calculate and apply global and local weights of each feature, create a feature vector for each image.

(3) Normalize the feature vectors for all the images.

(4) Construct intra differential feature-image matrix $D_I^{m \times n_I}$, such that each of its columns is an intra differential feature vector.

(5) Construct an extra differential feature-image matrix $D_E^{m \times n_E}$, such that each of its columns is an extra differential feature vector.

(6) Decompose $D_I^{m \times n_I}$ and $D_E^{m \times n_E}$ by SVD algorithm into USV form. Find proper values of K_I and K_E to define the likelihood functions $P(x|D_I)$ and $P(x|D_E)$ using the equation 1.

(7) Define the posteriori function

$$P(D_I|x) = \frac{P(x|D_I)P(D_I)}{P(x|D_I)P(D_I) + P(x|D_E)P(D_E)}, \tag{3}$$

where $P(D_I)$ is set to $1/n$, and $P(D_E) = 1 - P(D_I)$, n is number of clusters in the collection.

Automatic Image Classification by DLSI Space Based Classifier: Given an image as a query to be classified:

(1) A feature vector is set up by generating the features as well as their frequency of occurrence in the image, so that a normalized feature vector N is obtained for the

image. For each of the clusters in the image database, repeat the procedure of items (2)-(4) below

(2) Using the image to be classified, construct a differential feature vector, $x = N - C$, where C is the normalized vector giving the center or centroid of the cluster.

(3) Calculate the intra-image likelihood function $P(x|D_I)$, and the extra- image likelihood function $P(x|D_E)$ for the differential image feature vector x.

(4) Calculate the Bayesian posteriori probability function $P(D_I|x)$.

(5) Select the cluster having a largest $P(D_I|x)$ as the recall cluster.

2.3 Discussion

In the case of document classification, we know that the frequency of each term in the document is calculated, which constitutes the term vector to represent each individual document in the document collection. We can consider this term as global feature. For image classification purposes, we use the local texture feature which is extracted from the part of the image (i.e. from the image block). The global color feature is also used to construct the feature vector, as we know in some cases global features like color contains vital information so that we can not avoid them completely. Although it is also possible to compute the color features locally and use them in feature vector, we did not employ them in our method because of some technical difficulties. For example, time limitation. We believe that in the future local color feature will be able to improve the classification accuracy further. Since the use of local features gives better performance

of DLSI classifier, we can say that the local feature based DLSI approach is more robust than global feature based DLSI approach.

Chapter IV

Experiments and Results

This chapter provides the experimental details and results after employing two local feature based methods: districted matching approach and DLSI method in the area of gene recognition and content based image classification respectively.

1 Districted Matching Approach for Start Codon Prediction

This section discusses about the data set and the detailed experimental results of using districted matching approach for prediction of translation initiation sites in mRNA sequence.

1.1 Data Set

The experiment is done on the well-known *Arabidopsis thialiana TIS set* and *vertebrate TIS set* provided by Pedersen and Nielsen [PN97]. Here we compared the original artificial neural network approach proposed by Pedersen and Nielsen to its "districted voting" version. The data were extracted from GenBank [BBLO97], and the possible introns were removed by the splicing of mRNA sequences. Only high quality sequences containing at least 10 nucleotides upstream and 150 nucleotides downstream of the initiation point were selected, and redundancy was reduced so as to avoid over-estimated performance resulting from biased over-represented samples. The Arabidopsis thialiana TIS set contains 523 sequences, and the vertebrate TIS set contains 3312 sequences. Fol-

lowing the setting of [PN97], we generate one data point for each potential start codon (the triplet AUG) on each sequence. Each data point is represented by a sequence window of 203 nucleotides centered around the respective AUG triplet. In case a sequence window "falls off the edge", i.e., the triplet AUG lies less than 100 nucleotides from either end of the available sequence, the positions missing from the 203 nucleotide window are filled with E, the symbol for unknown. This results in 13505 data points for the vertebrate set, and 2048 for the Arabidopsis thialiana set. We use roughly 80% of the data points of each set for training, and the remaining 20% for testing. The vertebrate test set contains a total of 2700 data points of which 666 are start codons, and the A. thialiana test set contains 410 data points of which 107 are start codons.

1.2 Experimental Details and Results

The performances are mainly estimated by Matthews correlation coefficient [Mat75]. All the neural network experiments are written in Matlab 6.

1.2.1 Undistricted Neural Network

The artificial neural network proposed by Pedersen and Nielsen has 3 layers with 30 hidden units and 2 outputs [PN97]. The inputs were presented by encoding nucleotide sequences into a binary string, using a sparse coding scheme where each nucleotide is represented by 5 binary digits (personal communication): A=00001, C=00010, G=00100, T=01000 and E=10000. Thus the neural network has $203 \times 5 = 1015$ inputs. As it is well known in practice that different implementations of a neural network approach might

result in slightly different performance, we re-implement the original version of Pedersen and Nielsen's approach using Matlab 6. We use the same amount of inputs, outputs, hidden neurons as in paper [PN97]. The activation function used for each hidden layer neuron is a/the hyperbolic tangent sigmoid transfer function (*tansig*) and the activation function used for each output neuron is a/the saturating linear transfer function (*satlins*). Gradient descent with momentum and adaptive learning rate back-propagation algorithm (*traingdx*) is used for neural network training.

The best performance that we obtained on the vertebrate set showed a Matthews correlation coefficient of 0.5955 with overall accuracy of 85.52%, sensitivity of 64.41% and specificity of 92.43%. The best performance for the Arabidopsis data set yielded a Matthews correlation coefficient of 0.7058, with overall accuracy of 88.78%, sensitivity of 76.63% and specificity of 93.07%.

The best performances obtained in [PN97] for these two sets are 0.6208 and 0.7122, respectively. We believe that the differences come from different implementations. While we notice that 0.6208 and 0.7122 are still lower than most of the best results we obtained by using districted neural networks, we believe that it should be fair to compare the results using the programs by the same programmer in the same programming environment.

1.2.2 Districted Neural Network

For the districted neural network approach, we should partition the input vectors into blocks. We use a regional sub-neural network for each block, then an assembling sub-neural network to generate the final results. Notice that each data point is represented by

a sequence 203 nucleotides. We partition each data point into r equivalent subsequences, $\text{Win}(1), \text{Win}(2), \cdots, \text{Win}(r)$, and associate these subsequences with same label "start codon", "non-start codon" indicating that the AUG centered at the original window sequence is a start codon, or a non-start codon. When 203 is not divisible by r, we let the last block contains slightly more nucleotides than other blocks do.

For each i $(1 \leq i \leq r)$, we implement a regional sub-neural network, using the same structure as the undistricted neural network described above but with fewer inputs and fewer hidden neurons, for the subsequences $\text{Win}(i)$ of all the data points. As a total, we have r regional sub-neural networks, each of which has 2 output neurons. The numbers of hidden neurons for the regional sub-neural networks are smaller than that for the undistricted neural network. In our experiments, we use 10 hidden neurons for all these regional sub-neural neural networks in all the cases.

For the assembling sub-neural network, we implement a neural network with $2r$ inputs, 2 outputs and r hidden neurons. The inputs of the assembling sub-neural network come from the outputs of the regional sub-neural networks, and the two outputs are the results indicating whether the centered AUG of the concatenated sequence of the inputs of the regional sub-neural networks is a start codon or a non-start codon. The activation function used for each hidden layer neuron is a hyperbolic tangent sigmoid transfer function (*tansig*), and each of the output neuron uses a log-sigmoid transfer function (*logsig*) as its activation function. The training function (*trainlm*) is used to update weight and bias values according to Levenberg-Marquardt optimization. Notice that each regional sub-neural network is trained separately, and the assembling neural network is trained after all the regional sub-neural networks have been trained. The training sets for these

sub-neural networks are generated using the method described in Section 1.2.2 from the same training set used for undistricted neural network. Of course, we use the same testing set for test purposes. The Matthews correlation coefficients and accuracy we obtained by

Table 2: Performances of districted neural networks for vertebrate data set

Number of regions	Matthews correlation	Accuracy	Sensitivity	Specificity
3	0.6260	85.89%	73.57%	89.92%
5	0.6277	86.15%	72.07%	90.76%
7	0.6414	86.19%	76.87%	89.23%
9	0.6246	85.37%	76.72%	88.20%
11	0.5994	85.63%	65.01%	92.38%
13	0.5121	82.74%	56.91%	91.20%
20	0.5159	83.26%	53.15%	93.12%
40	0.5611	84.07%	63.96%	90.66%

using the districted neural networks for vertebrate and Arabidopsis thialiana data sets for different sizes of blocks are recorded in Tables 2 and 3 respectively.

Table 3: Performances of districted neural networks for the Arabidopsis thialiana data set

Number of regions	Matthews correlation	Accuracy	Sensitivity	Specificity
3	0.7558	90.73%	79.44%	94.72%
5	0.7100	89.02%	75.71%	93.70%
7	0.8034	92.43%	85.04%	95.04%
9	0.7159	89.27%	75.70%	94.05%
11	0.7448	90.49%	73.83%	96.36%
13	0.7442	90.49%	70.09%	97.69%
20	0.7243	89.51%	77.57%	93.73%
40	0.6788	88.05%	70.09%	94.39%

We could clearly see that, in most of the cases, the districted neural network approach performs much better than undistricted neural networks. The reduced performances, for the cases when the sizes of regions are 5 (corresponding to the districted neural networks

with 40 regions, respectively), could be taken as the cases when the regions are too small to fully satisfy the average distribution assumption.

1.3 Computational Complexity Analysis

The computational complexity of a neural network depends on several factors. Neural networks are intricate devices, with a number of different implementations and no generally accepted standard form. The computationally important aspects of a neural network include the network topology (the properties of the graph of nodes), the signal propagation method (the flow of signals through the graph), the activation function (used to map inputs to outputs on the neuronal level), the input weights (used to determine the relative values of inputs to a neuron from some other neurons), the update procedure (used during the training process to change weights), nodal complexity (the node count in the network), and the computational model (whether deterministic or probabilistic). Computational complexity of neural network is calculated in terms of how complicated a neural network must be in order to complete the required computation [Kro].

Generally the structural complexity of a neural network depends on the number of the hidden nodes which is also related to input dimension. Often the number of nodes increases along with the increase of the net's input dimension. As the number of nodes increases the structural complexity also increases. Hence the successful reduction of the net's input space can significantly decrease the network structural complexity, whereby the nodal complexity and also time complexity (the network's learning converges faster) [Hua03].

The districted version of neural network consists of several regional sub neural networks. Each regional-neural network takes a sub-input space as its own input. The outputs of regional neural networks are applied to a higher level neural network. So the output of higher level assembling neural network is the combination of sub-neural network's outputs. Since districted approach divides the high dimensional input space into several low-dimensional ones, the number of inputs and hidden nodes required for each regional sub neural network is much smaller than original undistricted version of neural network. In addition to decreasing the structural complexity it also decreases the time complexity, since each regional neural network converges much faster than the original undistricted neural network. In this complexity analysis we did not consider other important aspects, which affect the computational complexity of neural network. Because here we just want to compare the complexity of districted and undistricted neural network. For both the districted and undistricted version, we used the feed-forward backpropagation neural network with same weight function, net input function, and transfer and training functions.

Here we recorded the time required to train districted and undistricted version of neural network only for Arabidopsis thialiana TIS set. For districted neural network we consider blocks with reasonable sizes (5, 7, 9 and 11). We did not record all the cases because of time limitation. We believe that these example cases are enough to give the insight of how the time required for districted neural network depends on the size of block.

We used three iterations of training for each neural network (i.e., train each neural network three times) and recorded the CPU time required by training and calculate

Table 4: Time required for regional sub neural network (in seconds).

No of region	Average time for Training
5	23.829
7	20.499
9	17.258
11	15.0212

Table 5: Time required for assembling sub neural network (in seconds).

Iteration	NN with 5 regions	NN with 7 regions	NN with 9 regions	NN with 11 regions
First	38.89	29.33	34.16	36.47
Second	18.84	31.04	33.06	20.38
Third	20.92	24.27	6.92	28.18
Average Training Time	26.217	28.213	24.7133	28.3433

the average of them. Here are the tables showing that the average times required for each sub-neural network in districted approach and the neural network in undistricted approach.

Table 6: Time required for undistricted neural network (in seconds).

Iteration	Time
First	156.05
Second	87.44
Third	174.44
Average time	139.31

So from the above experimental results we can see, as the number of blocks increases the input size of regional neural network decreases (although we used the same number of hidden nodes for different size of blocks) and the computational time for training the neural network also decreases. The average time required for each regional neural

network is almost $1/n$ of the time required from the undistricted neural network where n is the number of regions. For example if the average time required for training of undistricted neural network is T then the time required for each regional neural network is almost T/n, where n is the number of blocks or regions. Again the time required to train the assembling neural network is relatively small. So if we train the regional neural networks sequentially then the time required for districted matching approach is slightly larger than the undistricted version because of the additional training time required for second level neural network. On the other hand if we train the regional neural networks in parallel then the time complexity of districted neural network will reduce effectively than the undistricted version of neural network. So we can conclude that the proposed districted approach of neural network gives much better prediction accuracy with a same or faster learning speed than undistricted approach.

2 DLSI for Image Classification

This section illustrates the experiments that have been conducted using DLSI for content based image classification.

2.1 Image Collection

We conducted our experiments on a collection 150 COREL images. The image classification community does not yet have enough experience and established systems for a common data set testbed to be used for comparison. Thus we downloaded several images from Internet [dat] which are widely used by image retrieval research groups [LWW01]

Fig. 6: The samples of images in the collection

[LW03]. We created our own image database using 150 images from it, which are suitable for content based image classification purposes. The images are of size 384×256. The image collection is divided into fifteen different semantic clusters: buses, dinosaurs, elephants, horses, roses, cherries, fireworks, sea, sunset, hills, ancient buildings, aero planes, deserts, people and snow– 10 images each, based on their contents. Figure 6 shows 15 typical images in the collection. We choose 5 images from each of the 15 clusters to construct the training set, and use the remaining 75 images for testing.

2.2 Experimental Details and Results

All the experiments of content based image classification are carried out using Matlab 6.

2.2.1 DLSI with Global Features

LSI vs DLSI

To compare effectiveness of traditional LSI and proposed DLSI approach for image classification purpose the LSI-based and DLSI-based classifiers are set up using the above training set, and tested on the testing set. Initially 100 color features are extracted from the images in the image collection so that each image feature vector has 100 components. The value of k in the SVD decompositions of LSI and DLSI algorithms is important because it represents the dimensions of the reduced LSI and DLSI spaces. The classification accuracy on the testing images for LSI and DLSI methods with different values of k are shown in graphically in Figure 7. The image-feature matrix for LSI method is of size 100×75. We could notice that, the LSI approach works best for $k = 35$, where the best accuracy of 70.66% is reached. To apply the DLSI on the same data set we generate intra and extra feature-image matrices of size 100×60 and 100×74 respectively. DLSI method gives the best accuracy of 81.33% when $k_I = k_E = 25$. It is clear that DLSI method is able to reduce the dimensions more effectively and eliminate the noise more efficiently than LSI method. Figure 7 shows some of the query images and retrieved clusters using both LSI and DLSI method.

SVM vs DLSI

We also employed a support vector machine (SVM) approach for the classification problem to demonstrate the performance of our approach. SVM approach which introduced by Vapnik [Vap98] provides state-of-the-art performance in many classification

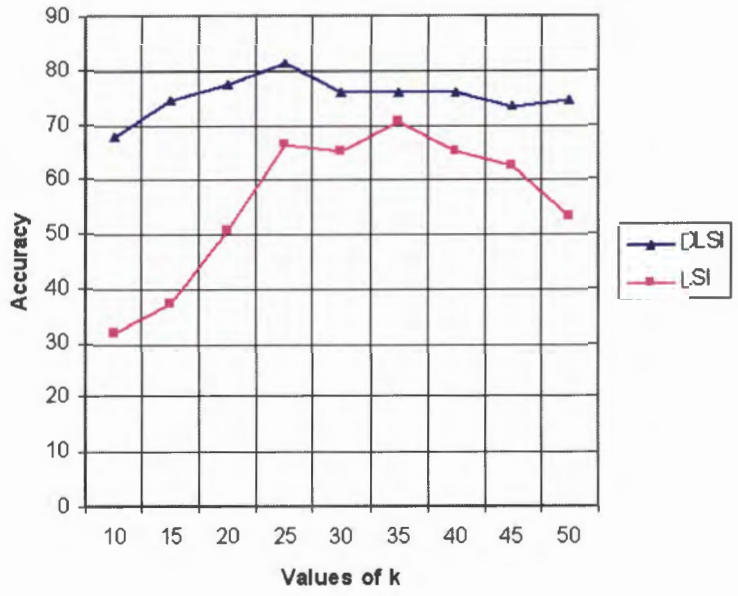

Fig. 7: The classification accuracy of LSI and DLSI for different values of k

problems such as text categorization, hand-written character recognition, face recognition, image classification, and bioinformatics. It has been used for image classification and retrieval by several authors [GCC01] [TC01]. The optimization criterion in SVM is the width of the margin between the classes. Training samples are mapped into a higher dimensional space and SVM tries to find a linear separating hyperplane with the maximal margin in the higher dimensional space. It is mainly a two-class classifier. Therefore we designed 15 binary SVM classifiers for 15 image classes. Each classifier separates one image class from the rest 14 image classes. We used SVM*light* [Joa] developed by Thorsten Joachims [Joa99] for conducting the experiments of support vector machine. The same 100 color features are used for training and testing the SVM classifiers. Best performance is obtained by using polynomial kernel with degree 2. For that case we got the best accuracy of 78.66% while using DLSI we obtained the highest accuracy of

Fig. 8: Test images (top left corner) and the clusters recognized using LSI and DLSI (in best case) respectively

Fig. 9: Test image (top left corner) and clusters recognized by the DLSI approach, without texture features and using texture features respectively.

81.33% (for k=25).

2.2.2 DLSI with Local Features

In order to test the performance of the DLSI method with local texture feature together with global color features, we have divided the whole image into 24 non-overlapping blocks with each block of size 64×64 and calculated 4 texture features: energy, inertia, entropy and homogeneity for each of the block. As a result, the feature vector of each image has 196 components (100 color features and 96 texture features). The classification accuracy reached to 82.66% by doing so. Figure 9 shows the result of DLSI space based classification method after adding texture feature to feature-vector. The experimental result clearly demonstrates that local features are important for DLSI approach and able to improve the performance of global feature based DLSI scheme for image classification.

2.2.3 Computational Complexity

Although the classification time and space complexity increase slightly after adding the local texture features used by DLSI algorithm, the main time consuming part SVD and feature extraction of image database are done off-line manner. Therefore the execution time of DLSI is not very critical for real time image classification. Moreover we can say that the computation time required to classify input image in DLSI approach is almost same as in LSI or SVM approach.

Chapter V

Conclusion

This final chapter presents the summary of the overall thesis and some ideas for the future research work.

1 Summary

In this thesis we have tried to establish the difference between local and global feature based approaches to the problem of pattern classification. Two local feature based classification methods have been investigated. The theoretical analysis of the local feature based districted matching approach is described in detail. The theory shows the robustness of a local feature based method over a global feature based method for 1D pattern classification. In addition to the theoretical approach, experiments are provided on mRNA sequences to predict the translation initiation sites using districted neural network. Experimental results confirm that a districted neural network is able to increase the prediction accuracy considerably.

Differential Latent Semantic Indexing (DLSI), which has been used successfully for full text document retrieval and classification, is employed here to classify the image patterns. Experiments have been conducted on the COREL image database. The results of the experiments proved that the DLSI algorithm is suitable for efficient image retrieval and is more robust than the LSI approach in image classification. Our experiments also proved that DLSI outperforms SVM for content based image classification. Finally, after

combining local feature (texture) with global feature (color) we can improve the classification accuracy of the DLSI method.

2 Future Work

This section presents the ideas that came up during our thesis work which we will investigate in the future.

2.1 Recursive Districted Matching

The districted matching approach can be employed recursively. We can partition each region into smaller regions and perform local feature based districted matching scheme recursively for determining the winner of that region. We can assume that this multi-level districted matching scheme will be more stable against noise.

2.2 Districted DLSI

The districted matching approach can be used to improve the classification accuracy of the DLSI method. In this case the input image x and database images x_{mk} can be divided into small non-overlapping blocks. The classification task is then performed locally between corresponding blocks. We can apply the DLSI approach for each input image block and corresponding blocks of database images for each class. The local block determines the image cluster which gives the higher value of posteriori probability function and cast a vote for that image class. We repeat this process for all local image

blocks and keep track of all the votes v_{mk} received by each cluster in the database. Once the local voting is done, we determine how many votes were cast for each cluster by summing among all the image blocks of that cluster:

$$v_m = \sum_{i=1}^{k} v_{mk}$$

Then finally, we can select the image cluster by majority voting. Since the local analysis of the image encloses more useful information than the global histogram-based methods, we believe that the districted version of DLSI approach will classify the input image more accurately than the undistricted version of DLSI approach.

Certain other aspects, such as adding new local features (for example, a shape feature) to the DLSI approach would make it more flexible.

2.3 Local feature Based Approach for Popular Classification Methods

Detailed experiments on some other popular pattern recognition methods (such as support vector machine, hidden markov model (HMM)) in some common application domains like 3D object recognition and gene pattern classification based on local feature will be very interesting. The investigation of the local feature based method with shifting strategy is also left for our future work.

References

[AMHW03] M. Artiklar, X. Mu, M. Hassoun, and P. Watta. Local voting networks for human face recognition. In *Proceedings of International Joint Conference on Artificial Neural Networks*, Oregon, 2003.

[Bao] H. T. Bao. Knowledge discovery and data mining techniques and practice. Retrieved 10 Jun, 2005, http://www.netnam.vn/unescocourse.

[BBLO97] D. A. Benson, M. S. Boguski, D. J. Lipman, and J. Ostell. Genbank. *Nucleic Acids Research*, 25(1):1–6, 1997.

[CN05] L. Chen and S. Nilufar. A districted neural network for start codon prediction. *International Journal of Bioinformatics Research and Applications*, 2005. Accepted for publication.

[CNK04] L. Chen, S. Nilufar, and H. K. Kwan. Districted matching approach for 1D object classification. In *Proceedings of the 2004 International Symposium on Intelligent Multimedia, Video & Speech Processing*, pages 206–209, Hongkong, 2004.

[CT03a] L. Chen and N. Tokuda. Robustness of regional matching scheme over global matching scheme. *Artificial Intelligence*, 144(1-2):213–232, 2003.

[CT03b] L. Chen and N. Tokuda. Stability analysis of regional and national voting schemes by a continuous model. *IEEE Transactions on Knowledge and Data Engineering*, 15(4):1037–1042, 2003.

[CT05] L. Chen and N. Tokuda. A general stability analysis on regional and national voting schemes against noise —- why is an electoral college more stable than a direct popular election? *Artificial Intelligence*, 163(1):47–66, 2005.

[CTN01] L. Chen, N. Tokuda, and A. Nagai. Probabilistic information retrieval method based on differential latent semantic index space. *IEICE Transactions on Information and Systems*, E84-D(7):910–914, 2001.

[CTN03] L. Chen, N. Tokuda, and A. Nagai. A new differential LSI space-based probabilistic document classifier. *Information Processing Letter*, 88(5):203–212, 2003.

[dat] Retrieved 5 April, 2004, http://wang.ist.psu.edu/docs/related.

[DDL+90] S. Deerwester, S. Dumais, T. Landauer, G. Furnas, and R. Harshman. Indexing by latent semantic analysis. *Journal of the Society for Information Science*, 41:391407, 1990.

[FC90] M. Fleming and G. Cottrell. Categorization of faces using unsupervised feature extraction. In *Proceedings of IEEE International Joint Conference on Neural Networks*, volume 2, pages 65–70, 1990.

[GCC01] K. S. Goh, E. Chang, and K. T. Cheng. SVM binary classifier ensembles for image classification. In *Proceedings of the 10th international conference on Information and knowledge management*, pages 395–402, 2001.

[Hei02] D. R. Heisterkamp. Building a latent semantic index of an image database from patterns of relevance feedback. In *Proceedings of 16th International*

conference on Pattern Recognition, volume 4, pages 132–135, Quebec City, Canada, 2002.

[HHWP03] B. Heisele, P. Ho, J. Wu, and T. Poggiob. Face recognition: component-based versus global approaches. *Computer Vision and Image Understanding*, 91:621, 2003.

[HKZ98] J. Huang, S.R. Kumar, and R. Zabih. An automatic hierarchical image classification scheme. In *Proceedings of 6th ACM International Multimedia Conference*, pages 219–228, 1998.

[HS01] S. Hua and Z. Sun. Support vector machine approach for protein subcellular localization prediction. *Bioinformatics*, 17(8):721–728, 2001.

[Hua03] R. Huang. A divide-and-conquer fast implementation of radial basis function networks with application to time series forecasting. In *Proceedings of 4th international conference on intelligent data engineering and automated learning*, Hong Kong, 2003.

[JDM99] A. K. Jain, R. P. W. Duin, and J. Mao. Statistical pattern recognition: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1):4–37, 1999.

[Joa] Thorsten Joachims. SVMlight: Support vector machine. Retrieved 30 May, 2004, http://svmlight.joachims.org.

[Joa99] T. Joachims. Making large-scale SVM learning practical. *Advances in Kernel Methods–Support Vector Learning*, pages 213–232, 1999.

[Kar03] D. Karen. Recognizing image 'style' and activities in video using local features and naive bayes. *Pattern Recognition Letters*, 24:2913–2922, 2003.

[Kro] J. Kroll. Introduction to neural networks and computational complexity. Retrieved 3 Sep, 2004, http://www.cs.tufts.edu// jkroll/neuralcomp.html.

[Li03] X. Li. Image retrieval based on perceptive weighted color blocks. *Pattern Recognition Letters.*, 24:1935–1941, 2003.

[LNG00] J. Li, A. Nijami, and R. M. Gray. Image classification by the two dimensional Hidden Markov Model. *IEEE Transactions on Signal Processing*, 48(2):517–533, 2000.

[LW03] J. Li and J. Z. Wang. Automatic linguistic indexing of pictures by a statistical modeling approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(9):1075–1088, 2003.

[LWW00] J. Li, J. Z. Wang, and G. Wiederhold. IRM: Integrated region matching for image retrieval. In *Proceedings of 8th ACM International Conference on Multimedia*, pages 147–156, 2000.

[LWW01] J. Li, J. Z. Wang, and G. Wiederhold. SIMPLIcity: Semantics-sensitive integrated matching for picture libraries. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 23(9):947–963, 2001.

[Mat75] B. Mathews. Comparison of the predicted and observed secondary structure of t4 phage lysozyme. *Biochim. Biophys. Acta*, 405:442–451, 1975.

[NCK04] S. Nilufar, L. Chen, and H.K. Kwan. A DLSI approach for content based image classification. In *Proceedings of IEEE international conference on computational intelligence for measurement systems and applications*, pages 138–143, Boston, USA, 2004.

[PCGV02] M. Partio, B. Cramariuc, M. Gabbouj, and A. Visa. Rock texture retrieval using gray level co-occurrence matrix. In *Proceedings of 5th IEEE Nordic Signal Processing Symposium*, Norway, 2002.

[Pec97] Z. Pecenovic. Intelligent image retrieval using latent semantic indexing. Master's thesis, Swiss Federal Institute of Technology, Lausanne, Vaud, 1997.

[PLK04] S. B. Park, J. W. Lee, and S. K. Kim. Content based image classification using a neural network. *Pattern Recognition Letters.*, 25:287–300, 2004.

[PN97] A. G. Pedersen and H. Nielsen. Neural network prediction of translation initiation sites in eukaryotes perspectives for est and genome analysis. In *Proceedings of the 5th International Conference on Intelligent Systems for Molecular Biology*, volume 5, pages 226–223, 1997.

[TC01] S. Tong and E. Chang. Support vector machine active learning for image retrieval. In *Proceedings of the 9th ACM international conference on Multimedia*, pages 107–118, Ottawa, Canada, 2001.

[tis] Translation initiation site prediction. Retrieved 13 Jun, 2005, http://mlkd.csd.auth.gr/TIS.

[TP91] M. Turk and A. Pentland. Eigenfaces for recognition. *Cognitive Neuroscience*, 3(1):71–86, 1991.

[Vap98] V. N. Vapnik. *Statistical Learning Theory.* John Wiley & Sons, 1998.

[ZG02] R. Zhao and W. I. Grosky. Negotiating the semantic gap: from feature maps to semantic landscapes. *Pattern Recognition*, 35:593–600, 2002.

Appendix

List of Publications

Refereed Journal Articles

1. L. Chen and S. Nilufar. A districted neural network for start codon prediction. *International Journal of Bioinformatics Research and Applications*, 2005. Accepted for publication.

2. L. Chen, R. Chen, and S. Nilufar. Improving the Performance of 1D Object Classification by Using the Electoral College. *Knowledge and Information Systems*, June, 2005. Accepted for publication.

Other Refereed Contributions

1. S. Nilufar, L. Chen, and H. K. Kwan. A DLSI approach for content based image classification. In *Proceedings of IEEE international conference on Computational Intelligence for Measurement Systems and Applications*, pages 138-143, Boston, USA, 2004.

2. L. Chen, S. Nilufar, and H. K. Kwan. Districted matching approach for 1D object classification. In *Proceedings of the 2004 International Symposium on Intelligent Multimedia, Video & Speech Processing*, Hongkong, 2004.

3. S. Nilufar, L. Chen, and H. K. Kwan. A Fuzzy Query Based Image Retrieval System. In *Proceedings of third International Conference on Communications, Circuits and Systems*, Hong Kong, May 27-30, 2005.

4. S. Nilufar, L. Chen, and S. A. Siddique. Efficient Image Retrieval System based on Differential Latent Semantic Index. In *Proceedings of the IASTED International*

Conference on Computational Intelligence, Calgary, Canada, July 4-6, 2005.