Development of Advanced Techniques for Pattern Recognition

A Thesis

submitted in partial fulfillment of the requirements for the award of the degree of

Doctor of Philosophy

in

Computer Science & Engineering Department

by

Anshu Singla
(Reg. no.: 951211003)

Thapar University
Patiala-147004, Punjab, India
November, 2016

Candidate Declaration

I hereby certify that the work, which is being presented in the thesis, entitled **Development of Advanced Techniques for Pattern Recognition**, in partial fulfillment of the requirements for the award of the degree of **Doctor of Philosophy** and submitted to the institution is an authentic record of my own work carried out during the period **January, 2013 to November, 2016** under the supervision of **Dr. Swarnajyoti Patra**.

The matter presented in this dissertation has not been submitted elsewhere for the award of any other degree or diploma from any institution.

Date: 25/09/17

Anshu Singla
Candidate

This is to certify that the above statement made by the candidate is correct to the best of our knowledge.

Date: 25.09.17

Dr. Swarnajyoti Patra
Supervisor

The Ph.D. Viva-Voice examination of **Anshu Singla**, Research Scholar has been held on ____________________ .

(Supervisor)

Mita Nasipuri
(External Examiner)

(Chairperson of Research Committee)

*.........dedicated to my husband **Mr. Yogesh Singla** and my lovely daughters **Charvi & Sachika***

Abstract

Pattern Recognition (PR) is the ocean of paradigms and its applications exist almost in every domain. In this modern computing world, each domain like biometric systems, medical diagnosis, banking, remote sensing, voting, forecasting etc. has huge amount of data in the diverse form. The main objective of PR is to recognize, classify, and analyse such data to make inferences. The PR algorithms fall under the umbrella of machine learning which can be broadly classified in two categories : supervised, and unsupervised algorithms. In supervised learning, the training of model depends only on the labeled data available. In contrast, unsupervised learning paradigms are developed based on the unlabeled data. There exist huge number of supervised and unsupervised learning paradigms in literature and can be classified into different categories. In the present work, the objective is to develop few advanced pattern classification techniques based on both the paradigms which can be applied on various applications of different domains for effective classification. More specifically, the present dissertation accomplishes the following objectives:

1. Developed a semi-supervised learning technique that selects the transductive samples by incorporating new criteria in sample selection process. The proposed technique out performs in two real situations: **i)** when the initial training samples are biased and, **ii)** when the initial training samples set is poor.

2. A new fast partition based batch mode active learning technique based on SVM classifier has been developed which gives high accuracy even if the initial SVM is poor. A novel partitioning method has been designed which first divides the unlabeled samples into partitions in one-dimensional feature space according to their distribution in the original feature space. Then to select the most informative samples from the unlabeled pool, one sample from each partition is selected based on an uncertainty criterion defined by exploiting SVM classifier. The number of unlabeled samples selected at each iteration of active learning is determined automatically and depends on the number of non-empty partitions generated.

3. Developed thresholding techniques that mitigate all the limitations of existing traditional thresholding techniques based on histogram of the image. A novel energy curve of the image has been designed which includes the spatial contextual information of image. It has better discriminatory capabilities in comparison to histogram of the image. This energy curve has been utilized to develop thresholding techniques that determine the number of objects present in the image automatically

and gives the optimal threshold values for image segmentation.

Extensive experimental work has been carried out to develop number of novel pattern classification algorithms. To assess the effectiveness of the newly designed algorithms results have been compared with the respective state-of-the-art techniques cited in the literature. Experiments have been performed over the number of standard datasets for each technique designed.

Keywords: Active Learning, Genetic Algorithm, Histogram, Image Segmentation, Support Vector Machine, Thresholding Technique, Transductive Inference.

Acknowledgements

First and foremost, I would like to express my deep gratitude to my supervisor **Dr. Swarnajyoti Patra** for his invaluable advice and encouragement at every step of my PhD program. Without his unfailing support and belief in me, this thesis would not have been possible. His contribution to this thesis goes well beyond his role as an academic supervisor and includes constant support on a personal level without which this journey may never have been completed. And for this, I am truly grateful.

I also express my sincere gratitude to **Dr. Maninder Singh**, Head of Computer Science and Engineering Department, Thapar University, for his kind help and cooperation. I would like to express my gratitude to **Dr. Rajesh Kumar** for his constant motivation and encouragement to complete my thesis work. I express my gratitude and special acknowledgement to **Dr. Harish Garg**, **Dr. Deepak Garg**, **Dr. Prashant Singh Rana**, **Dr. Pritpal Singh**, **Dr. Maninder Kaur**, **Dr. Anju Bala**, and **Mr. Shatrughan Modi** for their constant motivational and technical support in completion of my thesis work. I also wish to thank my research committee members, all my colleagues and non-teaching staff of the institute for their help and support.

I would like to express my sincere and deep gratitude to my husband **Mr. Yogesh Singla** for his love, encouragement, care and support. Finally lot of love to my both loving daughters **Ms. Charvi** and **Ms. Sachika** who always slept in night with their father during my thesis work and instructed me with love that I can continue with my thesis work.

Anshu Singla

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ABC	Ant Bee Colony
ACSTT	Automatic Context Sensitive Thresholding Technique
AL	Active Learning
CC	Cross Correlation
DB	Davies Boulding
DNA	DeoxyriboNucleic Acid
DT	Decision Tree
ECCS	Energy Curve based Context Sensitive
FODPSO	Fractional Order Darwinian Particle Swarm Optimization
GA	Genetic Algorithm
HBMO	Honey Bee Mating Optimization
HMM	Hidden Markov Model
HSI	Hyper Spectral Image
HYTA	HYbrid Thresholding Algorithm
ISOLET	Isolated Letter Speech Recognition
ISVM	Inductive Support Vector Machine
k-nn	K Nearest Neighbors
LIBSVM	LIBrary of Support Vector Machine
MATLAB	MATrix LABoratory
MBF	Modified Bacterial Foraging
ML	Machine Learning
MLC	Maximum Likelihood Classifier
MPTSVM	Modified Progressive Transductive Support Vector Machine
MRF	Markov Random Field
MS	Marginal Sampling
MS-ABD	Marginal Sampling Angle Based Diversity
MS-cSV	Marginal Sampling Closest Support Vector
MS-RP	Marginal Sampling Region Partitioning
MSE	Mean Squared Error
OAA	One Against All
OAO	One Against One
OCR	Optical Character Recognition
PC	Personal Computer
PCA	Principal Component Analysis

PCS	Pattern based Context Sensitive
PR	Pattern Recognition
PSO	Particle Swarm Optimization
PTSVM	Progressive Transductive Support Vector Machine
RBF	Radial-Basis Function
SAR	Synthetic Aperture Radar
SMS	Short Message Service
SOM	Self Organizing Map
SSL	Semi Supervised Learning
SVM	Support Vector Machine
TSVM	Transductive Support Vector Machine

Chapter 1

Introduction and Scope of Thesis

1.1 Pattern Recognition

Pattern is a regular and intangible form which can be defined as a point in n-dimensional feature space. Pattern Recognition (PR) is one of the most important aspect in real world that involves identification of patterns (objects), finding similarities among patterns, matching new patterns with the existing patterns. Recognition of patterns is a basic necessity of every human being to deal with the real life problems like to read newspaper one should be able to recognize the alphabets, to make painting one should be able to recognize different colors. Pattern Recognition also plays vital role in designing critical real time applications say detection of radar signals, tumor in human body, biometrics for highly secure identification, predictions for equity market, natural disasters like volcano eruption, and so on. It is a branch of machine learning and defined as science of making inferences based on perceptual data or given patterns [1]. For instance, from the given number of different shaped celestial objects, a student has to recognize round shape objects. To recognize such objects/patterns, one must know the different representations of round shape i.e. oval, circle, elliptical or spherical. Further, from those round shape selected objects, he has to recognize different types of stars. Based on the features like size, density, temperature, weight the round shape objects (perceptual data) can be further classified in different types of stars like red, blue or yellow dwarf, neutron star etc. This classification can be used further to make inferences say about life of star, future fusions etc. The universal nature of PR has encouraged the number of task specific methodologies and enriched the number of links with other disciplines. Consequently, vast number of challenging and computationally demanding applications are emerging in different disciplines [2], few of which are enumerated in Table 1.1.

The basic meaning of pattern is regular and repeated arrangement. For e.g.: A grid which comprise of regularly spaced horizontal and vertical lines, those horizontal and vertical lines are known as patterns. Pattern can also be defined as customary way of behaviour or operation. For e.g.: a trajectory followed by flight, a speech signal etc. Mathematically, the pattern X can be represented by the feature vector such that $X =$

Table 1.1: Examples of Pattern Recognition applications in different disciplines.

Discipline	**Application**	**Input Pattern**	**Pattern Classes**
Biology	Sequence analysis [3, 4]	DNA/Protein sequence	Known types of genes/ patterns
Engineering	Document classification [5, 6]	Text document	Semantic categories like weather, sports, economy etc.
	Speech recognition [7]	Sound wave	Spoken words
	Text classification [8, 9]	Words image	Classification of characters.
	Face recognition [10, 11]	image	Verification of faces.
Earth Science	Geology: Rocks classification [12]	Microscopic images	Well known categories of rocks e.g. granite, Lime stone etc.
	Remote Sensing [13, 14]	Multi spectral images	Regions categorised as pure water, turbid water, concrete area, habitation, vegetation etc.
	Land cover classification [15, 16, 17]	High resolution images like Synthetic aperture radar (SAR) images	Detect changes by acquiring series of images like maps etc.
Agriculture	Analysis of crops [18, 19]	Multi-spectral images	Land categories, growth pattern of crops
Security	Personal identification [20]	Fingerprints	Authorized users

$(x_1, x_2, ..., x_d)|x_i \epsilon \Re^d$ where d gives the dimensionality of the feature vector. Consider an example of Library which has vast collection of books. Each book can be abstracted as pattern X represented by the feature vector which may consist of number of features like book title, author's names, cost, edition, publisher, no. of pages, type of book: reference book, text book, hand book etc. In this example, feature vector of instance of book consists of $d = 6$ features. Thus, the pattern of book X can be represented as $X = (x_1, x_2, ..., x_6)|x_i \epsilon \Re^6$ where x_1 represents book title, x_2 represents author's name and so on. It is worth noting that all the properties of a pattern may not be considered in feature vector. Only the essential properties of a pattern (used for classification) are considered in feature vector and rest all information regarding that object is ignored. For e.g.: To categorize/ classify the books as per specialization no. of pages of a book is not relevant attribute of book hence should not be considered in feature vector.

PR methods involve mathematical formulation of objects described by features and can be applied mostly in all the disciplines. This mathematical formulation depends on the type of problem. The PR deals with vast number of algorithms which can be broadly classified in four categories: **1)** Template Matching, **2)** Statistical, **3)** Neural Networks, and **4)** Structural. Depending on the nature and domain of the problem each PR algorithm can be further classified and there exist vast number of such algorithms [21]. The broad classifications and respective subclassifications are shown in Figure 1.1 and the brief description is as given below:

1. **Template Matching:** It is the simplest form of PR technique where the input pattern is matched against reference patterns say prototype. This matching may require certain operations like translation, rotation or scaling. Based on selected matching, the input pattern is classified in class with respect to the best prototype matched. Deformation models [22, 23, 24, 25] may be required if the images of patterns to be matched are distorted. This can be categorised as feature based and area based approach [26].

    (a) **Feature based approach:** Feature based approaches are based on the extraction of salient structures (features like points, curves, edges, lines etc). For e.g., Edge based methods [27] are based on the principal that the shape of an object is defined exclusively by the shape of its edges. Therefore instead of matching the whole pattern, one can extract its edges and match only the nearby pixels, thus avoiding unnecessary computations. Other subcategories include spatial relations, invariant descriptors, pyramids and wavelets and relaxation methods [28].

    (b) **Area based approach:** Area based approach is sometimes also referred as correlation methods [29, 30, 31]. This approach has its many different variants like:

        - Cross-correlation methods are based on the principal that if there exist a match between two images then the correlation between these two different images is very high.
        - Fourier based methods [32, 33] are preferred over Cross-correlation methods if the image is distorted or noisy. This principal increases the computational speed and indicates that the correlation in spatial domain among different images can be obtained by applying Fourier transforms.
        - Mutual information based method [34] is a very popular and powerful measure used to find any global relationship of intensities between two input

  images taken from unknown sources. Matching can be based on mutual information like absolute differences, left/right consistency checking and hole filling etc.

  - Optimization based method used to find a good (not necessarily perfect) solution to problem. It reduces the search time by orders of magnitude with respect to an exhaustive search. The algorithm can be applied to the different problems e.g., traffic signs, can be recognized by a navigating robot, locating object in an image etc. [35].

2. Statistical: This is the classical PR approach [36, 37] based on the mathematical formulas and methods. In statistical PR system, each pattern can be represented in d-dimensional space where d is the number of features associated to each pattern/object. The key feature to construct statistical model is that the variability is represented using probability distributions. The model should consider both random and systematic variations. Statistical methods can be broadly classified as: Probabilistic and Geometric approach described below briefly:

   (a) Probabilistic Approach: This approach can be further classified as: Parametric and non-parametric as shown in Figure 1.1. Parametric is applied when the probability distribution of patterns is known in advance [38, 39, 40]. In contrast, for non-parametric approach, the distribution is not known completely and require certain assumptions on the basis of which the patterns are classified. Recent algorithms cited in literature [41, 42] are employing non-parametric approach.

   (b) Geometric Approach: In this approach, best decision boundary (e.g., linear or quadratic) is identified based on which the training patterns are classified. Such boundaries can be constructed using, for example, a mean squared error (MSE) criterion. The direct boundary construction approaches are supported by Vapnik's philosophy [43]. There exist large number of approaches in this category like SVM, Decision trees and many more as shown in Figure 1.1. The main objective is to identify the decision boundary or region which may discriminate the patterns from one category to another. This decision boundary is obtained with the help of discriminant function. To determine the accurate discriminant function, the study of distribution of the patterns of each category is essential.

3. Neural Networks: Organization of brain is the key motivation behind this approach [44]. Patterns are assumed to be interconnected and defined as neurons. The in-

terconnections are used to send or receive signals among neurons. Neural networks are like black box with good learning and generalization capabilities [45, 46]. It's parallel distributed nature gives the high computing power. The neural network approach is based on two aspects : **i)** structure of neurons i.e. network architecture **ii)** learning algorithm. Neural networks can be used in number of different application domains [47, 48, 49, 50, 51]. There exist family of neural network architecture that comprises of: **a)** Feed forward network, **b)** Feedback/Recurrent network, **c)** Radial-basis function network (RBF), and **d)** Self-Organizing Map network (SOM) which are briefly described below:

   (a) Feed forward network: Generally in neural network, the neurons are organised in the form of layers. The information flow is uni-directional and no feedbacks are provided. The input layer comprises of all the source nodes. When the input layer projects onto an output layer or computation nodes and no information flows backward that is from output layer towards input layer, it is said to be feed forward neural network. Feed forward neural networks can be further categorised as single layer [52] and multilayer network [53]. The input layer do not perform any computation hence not counted. The intermediary computation nodes between the source layer and output layer is said to be hidden layer. If there is only input and output layer, it is said to be single layer feed forward network. In contrast, if we have input layer, hidden layer and output layer it is said to be multi-layer feed forward neural network.

   (b) Feedback/Recurrent network: The signals can travel in both the directions. It has atleast one feedback loop which improves the learning capabilities [54].

   (c) Radial-basis function (RBF) network: The model classifies the pattern by transforming it into high dimension space in a non-linear form [55]. It has many applications like face recognition [56].

   (d) Self-Organizing Map (SOM) network: The model classifies the pattern by transforming high-dimensional data into lower-dimensional space maintaining original topological relationship [57]. It is an unsupervised neural network and has large number applications [58, 59].

4. Structural: This approach maintains the structural information to classify each pattern and also has capacity to give reasons for which the pattern is ineligible to be in another class/es [60, 61]. This approach can describe a large set of complex patterns using small sets of simple pattern primitives and grammatical rules. This approach is very useful in the domains where the patterns have definite structure

and can be captured as a set of rules like sound waves, textured images, shape analysis. Syntactical approach is one form of structural approach in PR [62]. The implementation of this approach is quiet challenging as it requires detection of primitives and inference from the training data.

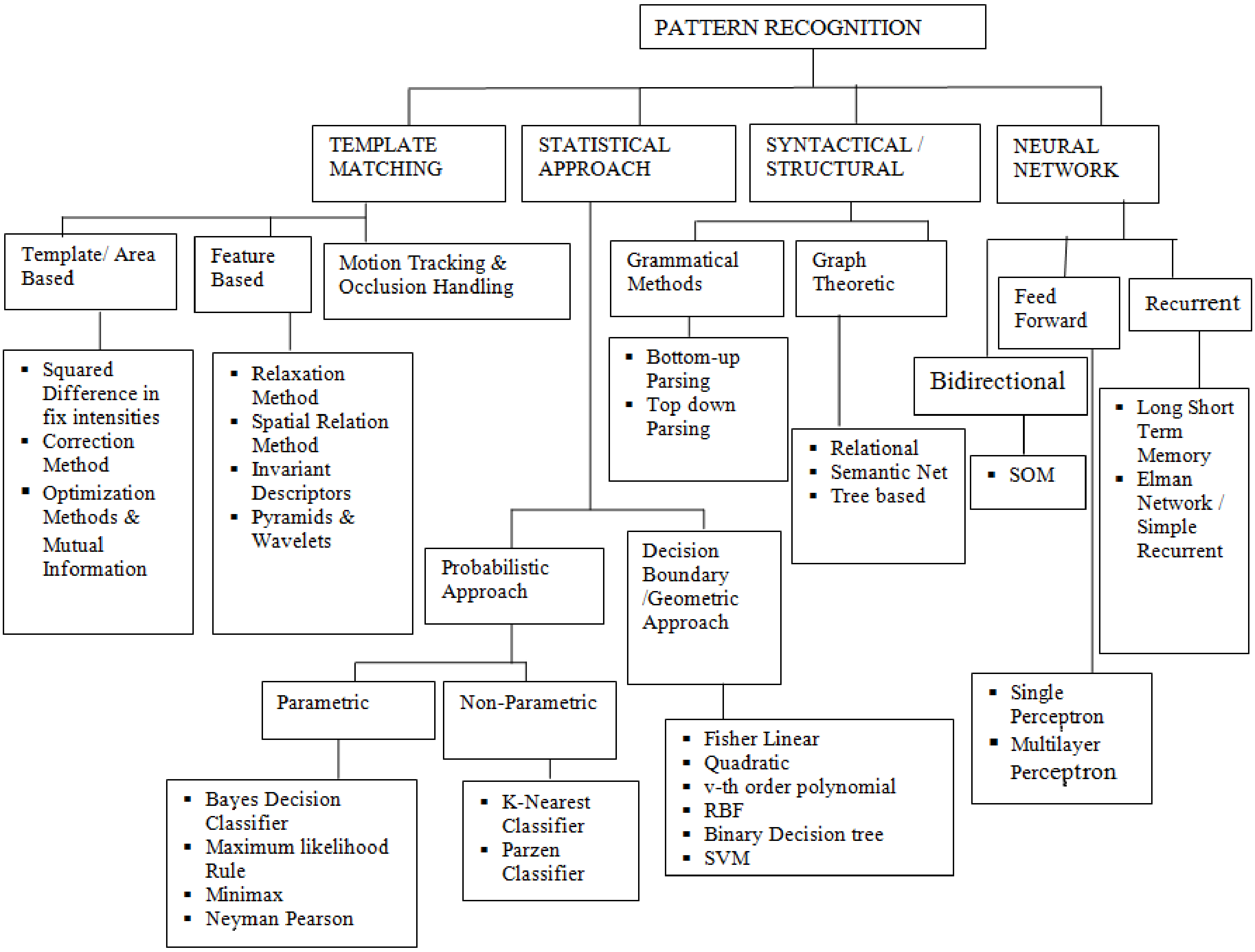


Figure 1.1: Different types of PR approaches and few existing algorithms.

The main goal of all aforementioned techniques is classification of objects. And the objective of classification problems is to identify the characteristics that indicate the group (category of the target variable) to which each case belongs. The groups/classes can be defined by some rules, class boundaries, or some mathematical function/s. The basic stages of the PR are shown in the Figure 1.2. Initially, the sensors capture the data about objects/patterns to be classified. For e.g.: To classify the stars in the space, different sensors like camera to capture images from different directions in space, telescope for position measurement, angular diameters etc. are used. In the next stage, features are generated using the raw data of objects collected from different sensors such as color of stars from captured images, parallax method used to measure distances, depth, size of star from telescope observations and so on. This stage is known as feature generation. In this stage, one has to be very careful that no relevant information about the object should be lost. Once this information is obtained, feature extractor discards the redundant and irrelevant information which may not be effective for object recognition. For e.g.: light emitted by star and distance of the star from the earth, both are highly correlated hence either of them may be selected and the other may be discarded to go further for classification. The attributes selected by the feature extractor is known as feature vector. Each pattern is now represented by its feature vector in feature space. This multidimensional feature space is given as input to a classifier. Data sensing, feature generation and feature extraction is beyond the scope of this thesis. It is assumed that the final output of these phases is a feature vector which represents the input data of the classifier design. The aim of the classifier is to divide the feature space into the number of classes/regions. For instance in case of stars represented by multidimensional feature space, the classifier aim is to assign the category/region (red dwarf, blue dwarf, yellow dwarf, neutron star) to each star pattern. Depending on the problem domain, dataset and the output required, the classifier algorithm may fall in either of three categories as shown in Figure 1.2 : **i)** Classification, **ii)** Regression and, **iii)** Clustering described in subsequent subsections.

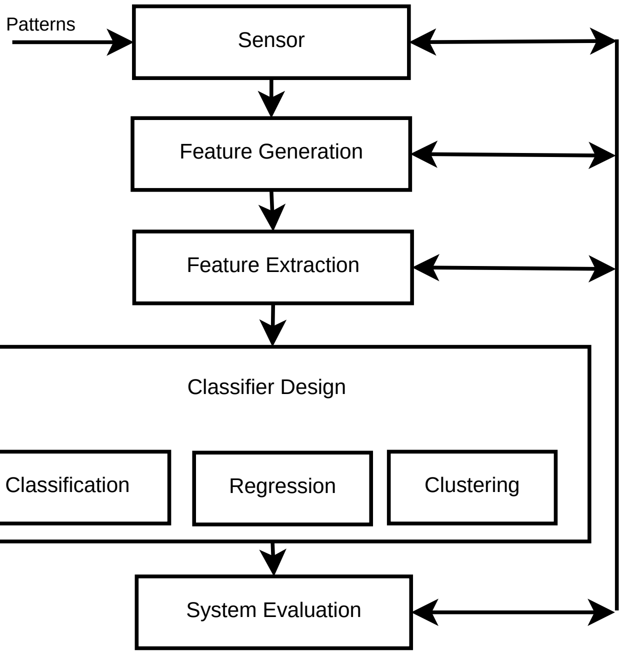


Figure 1.2: Basic stages followed to develop classification system.

1.1.1 Classification

Classifying and analyzing the patterns based on the training or labeled patterns is known as classification. The algorithm produces an inferred function, which can be used to map new patterns. An optimal scenario will allow the algorithm to correctly determine the class labels for unseen patterns. Mathematically, let X be the domain of patterns to be classified and the codomain of such patterns be Y. Given a set of training sample $\{(x_i, y_i)\}_{i=1}^{n}$, the learning algorithm will train a function $f : X \mapsto Y$, with the goal that f(x) predicts the true label y on future pattern x. Depending upon the nature of domain Y, the learning algorithm can be further categorised into two types: classification and regression. If the domain of Y is discrete the problem is categorised as classification problem. If the domain of Y is continuous in nature, it is said to be regression problem which is explained in subsequent subsection. In classification, the classes are generally encoded by integer numbers given as $Y \epsilon \{1, ..., C\}$ where C is the number of classes. If the number of classes is two i.e.$C = 2$, then the problem is defined as binary problem and we say, $Y = \{-1, +1\}$. If $C > 2$, the problem is said to be multi-class problem.

1.1.2 Regression

These learning algorithms are very similar to the classification algorithm with the difference that the Y is continuous in nature. Regression analysis is widely used in prediction and forecasting [63]. The parameters of a regression algorithm are generally estimated using the method of least squares. There exist several other methods like Bayesian linear regression [64], Least square percentage regression [65] for situations where reducing percentage errors is required, Least absolute deviations which is more robust in the presence of outliers, leading to quantile regression [66], Non-parametric regression [67] used when a large number of observations exist computationally intensive, and so on.

1.1.3 Clustering

Organization and grouping of patterns based on similarities and differences among patterns so that meaningful conclusions can be derived is known as clustering [68]. These conclusions analyse the patterns and assign labels to the patterns when the classes are not predefined. The main objective of clustering is to maximize the homogeneity within each cluster formed which means that the patterns belong to the same cluster should be more similar to each other as compare to the patterns that belong to different clusters. Clustering is recognized by different nomenclatures under different domains like unsupervised learning in PR, partitioning in graph theory, typology in social sciences, and so on. Mathematically, let X be the set of n unlabeled patterns represented by $X = \{x_1, x_2, ..., x_n\}$. Then partition of X into k-clusters say $\{C_1, ..., C_k\}$ referred as k-clustering should meet the following three conditions:

1. $C_i \neq \phi$ such that $1 \leq i \leq k$
2. $\cup_{i=1}^{m} C_i = X$
3. $C_i \cap C_j = \phi$ for $i \neq j$ and $1 \leq i, j \leq k$

Clustering algorithms can be classified in three categories [69, 70]: overlapping (so-called non-exclusive), partitional, and hierarchical clustering. In overlapping algorithms, the clustering of patterns is soft or fuzzy that is each pattern belong to one or more clusters with respect to different degrees [71]. Hierarchical and partitional clusterings, represents a hard partitioning of the data set into a different number of mutually disjoint subsets [70]. Manifold applications exist for clustering like image processing [72], market research [73], web mining [74], genes analysis [75], strategic management [76].

1.2 Pattern Recognition Algorithms

The performance of the PR algorithms depend on number of factors like data set, size of training set, dimensions of dataset, tuning parameters, and model used for PR system. As PR is the subset of machine learning, the aforementioned algorithms of PR shown in Figure 1.1 can be broadly subdivided into following machine learning techniques:

1. Supervised learning
2. Unsupervised Learning

The association of the PR algorithms with the learning techniques can be pictorially represented in the Figure 1.3.

The main objective of the present thesis is to develop advanced PR techniques which can be applied in different application domains. The brief description of aforementioned learning techniques and problem domains in which the work has been done is elucidated in the following subsections:

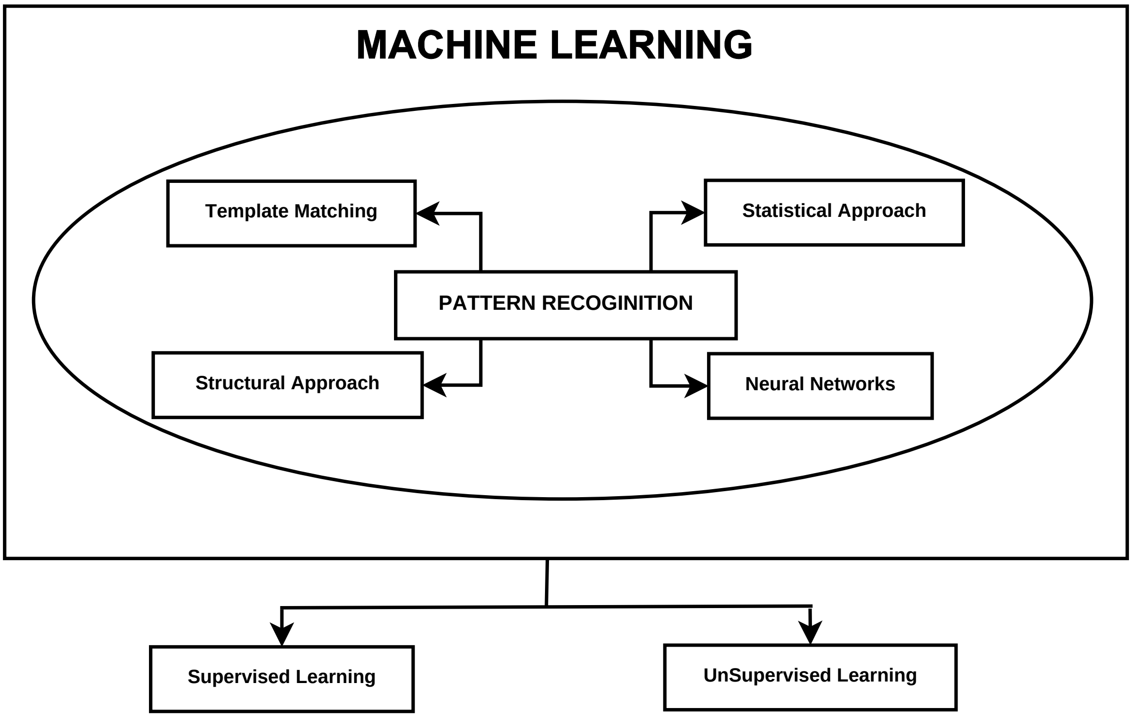


Figure 1.3: Pattern Recognition versus Machine Learning.

1.2.1 Supervised Learning

In supervised learning paradigm, the input pattern is identified as a member of a predefined class; supervised learning is the machine learning task of inferring a function from the labeled data. In supervised learning, there exist few training samples that consist of feature vector and the class to which they belong. Such samples are called as labeled data. A supervised learning algorithm analyzes the labeled data and generates a function, which can be used to map new unseen patterns/samples. The Figure 1.4 shows the labeled samples (patterns represented as points in space) and three different classes to which they belong to. Based on these labeled samples the function is inferred so that the unlabeled samples can be classified in future.

Supervised learning is also referred as learning with a teacher. Both the classifier and the supervisor are having exposure to the training data (labeled data) and classes to which the data belong. Thus, the classifier and supervisor with in-built knowledge can assign the label to unseen pattern. The system is evaluated and the feedback is given to retrain the classifier for better accuracy. This iterative process aims to emulate the teacher and provide better classifier which can label unseen pattern without teacher. Both the classification and regression algorithms described in subsections 1.1.1 and 1.1.2 respectively, are supervised learning algorithms. The literature on PR contains vast

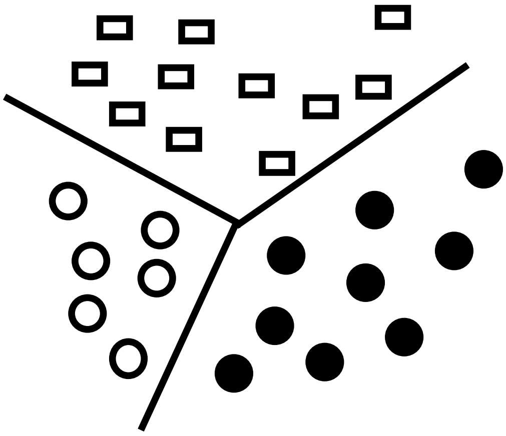


Figure 1.4: Patterns belong to three different classes are shown in two-dimensional feature space.

number of supervised learning algorithms [77]. The brief description of few is summarised below:

- Decision Trees (DT): It is a non-parametric classifier presented in [78]. It's structure consist of one root node and number of internal and terminal nodes. The data is recursively divided down the decision tree as per the framework of classification algorithm. It is widely used in different domains like land cover change assessment [79], fault diagnostic [80], feature subset selection [81], speech tagging [82] and many more.

- Discriminant Analysis: Discriminant analysis [83] is used for both data classification and data reduction. This supervised learning method maximizes the ratio of between-class variance to the within-class variance in any particular data set thereby guaranteeing maximal separability [84]. It has wide variety of applications in different domains like for face recognition problem [85], break down estimation [86] etc.

- Naive Bayes: It is a probabilistic classifier based on inductive learning [87]. It estimates a set of probabilities for the given data set as per Bayes theorem. Generally in real time applications, all the attributes of dataset independently gives the class variable value which is the basis of Naive bayes algorithm [88]. It also serve applications in different domains like disease prediction [89], network intrusion detection [90] etc. It has many variants like hidden Naive Bayes, selective Bayesian classifiers, naive Bayes tree, tree-augmented naive Bayes [91].

- Maximum likelihood classifier (MLC): It is a parametric classifier based on the Bayesian theorem [38]. It is applicable to the number of different domains like genes analysis [92], modulation [93], change detection [94] etc.

- Support vector machines classifier: The support vector machines (SVMs) are a set of related learning algorithms used for classification and regression. Like the DT classifiers, the SVM is also non-parametric classifier. It has been employed in the present dissertation thus explained in details in Section 2.2.

- K-Nearest neighbor: This algorithm is a non-parametric method used for both the classification and regression. In both cases, the input consists of the k closest training samples in the feature space. In this algorithm, the pattern x is assigned the class or label for the majority of it's k-nearest neighbors from a sample pattern. Nearest Neighbor algorithms depend critically on the distance metric employed like Euclidean distance, Manhattan's distance etc. It is frequently used in Geographical information systems [95].

- Ensembles: The number of different classifiers are pooled to make final classification decision. This provides an extra degree of freedom and yields better accuracy as compare to single classifier [96].

Other than this basic categorisation, an algorithm to classify the transmission line faults, based on combined use of neural network and fuzzy logic has been presented in [97]. Jens *et al.* in [98] considered three different statistical approaches to perform land cover analysis: maximum likelihood classification (ML), SVM , and iterated conditional modes (ICM), using a Markov random field. Alec *et al.* in [99] presents the classification of the sentiment of Twitter messages using distant supervision. In [100] Prostate cancer localization using supervised classification technique has been presented. The techniques used to transform multi-beam acoustic data into spatial layers that can be used for predictive habitat modelling has been presented in [101]. Yang *et al.* [102] evaluates six supervised classification techniques to effectively and economically control cotton root rot plant disease. In [103], a new binary supervised non-parametric classification approach based on iterative partitioning of multidimensional feature space into variably-sized and nested hyper-cubes to map land cover has been presented.

Supervised learning is an effective way to classify the data and plays vital role in PR systems but it has its own constraints and limitations: **i)** Supervised learning paradigms, require large amount of labeled samples to train the model. But in modern and real applications, abundant of unlabeled data is available where as labeled data is scarce [104]. In such situations, the generation of labeled samples increases the cost as well as time complexity to train the model. **ii)** Supervised learning paradigms are fully dependent on the availability of labeled samples. As discussed above, the generation of labeled samples is time consuming and expensive. The situations may rise when the available labeled samples may not represent the full statistics of all the classes present [105]. As a result, the performance of the model may be affected adversely. **iii)** The manual selection of labeled training samples may introduce redundancy in the training set of the classifier which further increases the computational time of the classifier without providing the additional information. In the situations of scarce labeled samples availability, two approaches: semi-supervised learning (SSL) and active learning (AL) are very fruitful. Semi-supervised and Active learning techniques aim to improve the performance of supervised learning techniques by exploiting unlabeled samples. But both these techniques are complimentary to each other in terms of selection of unlabeled samples in successive iterations. Both these techniques are elucidated in subsequent subsections.

1.2.1.1 Semi-supervised Learning

Semi-supervised learning makes use of both labeled and unlabeled data for training - typically a small amount of labeled data with a large amount of unlabeled data. Semi-supervised learning paradigm fall between supervised learning paradigm (with completely labeled training data) and unsupervised learning paradigm (where no labeled training data is available). The acquisition of labeled data for a learning problem often requires a skilled human agent or a physical experiment. Also, many machine-learning researchers have found that unlabeled data, when used in conjunction with a small amount of labeled data, can produce considerable improvement in learning accuracy. This is why the semi-supervised learning paradigms are widely used to extend supervised or unsupervised learning so that additional information of other learning paradigm can be included, to have better performance.

In semi-supervised learning paradigm, the learner first trains the classifier with the labeled data available scarcely. It can be seen from the Figure 1.5 that very small amount of labeled and large amount of unlabeled samples is present. Initially, function is inferred on the basis of very few labeled samples available. In successive iterations, the selection criteria is incorporated in such a way that most confident unlabeled samples along with their respective predicted labels are included to retrain the classifier.

Mathematically, the training data consist of both 'l' sized labeled samples $\{(x_i, y_i)\}_{i=1}^{l}$ and 'u' sized unlabeled samples $\{(x_i)\}_{i=1}^{u}$ of patterns. Assuming that $u \gg l$, the objective of semi-supervised classification is to train the function f from both the labeled and unlabeled samples, to improve the classification accuracy as compare to supervised learning paradigm applied on the labeled data alone. The study of semi-supervised learning

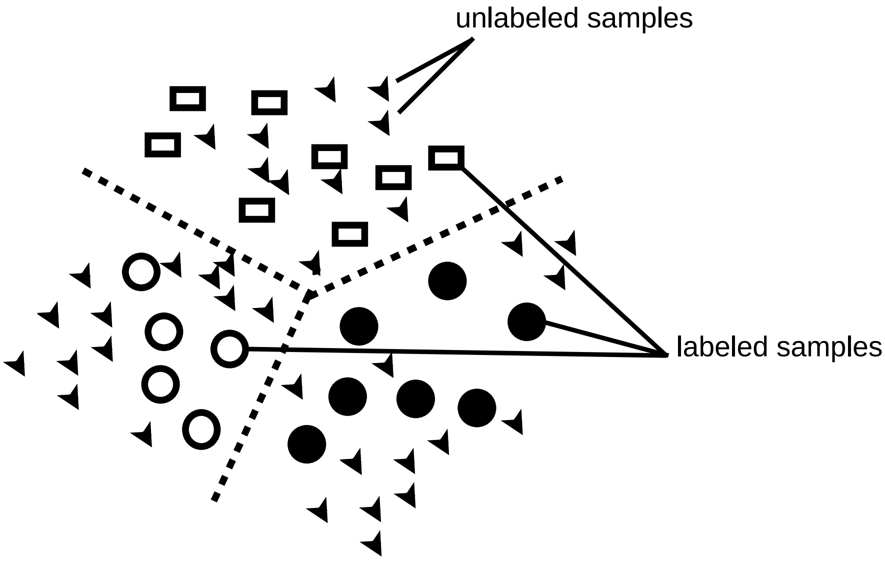


Figure 1.5: Distribution of the labeled and unlabeled patterns in the two-dimensional feature space. The correct labeling of unlabeled patterns play vital role to achieve better decision hyperplane.

bridges the gap between machine learning and human learning. This paradigm can be classified further as inductive and transductive learning.

- **Inductive learning:** Let $\{(x_i, y_i)\}_{i=1}^{l}$, $\{(x_j)\}_{j=l+1}^{l+u}$ be the training set, than a function f is defined as $f : X \mapsto Y$ such that f is expected to be a good predictor on the future dataset, beyond $\{(x_j)\}_{j=l+1}^{l+u}$.
- **Transductive learning:** In supervised learning paradigm, when the objective is to predict the labels of the unlabeled patterns present in the training dataset instead of labeling the future test data, it is defined as transductive learning. Mathematically, Let $\{(x_i, y_i)\}_{i=1}^{l}$, $\{(x_j)\}_{j=l+1}^{l+u}$ be the training set, than a function f is defined as $f : X^{l+u} \mapsto Y^{l+u}$ such that f predicts the unlabeled data $\{(x_j)\}_{j=l+1}^{l+u}$ accurately. Remember that f is only defined on the training set provided and is not required to predict future data set.

There exists wide variety of semi-supervised learning paradigms cited in literature [106] such as self-training, probabilistic generative models, co-training graph based methods, semi-supervised support vector machines (SVM), and so on. In [107], Joachims solved the quadratic optimization problem to implement the transductive SVM with an application to text classification. This algorithm is effective when the ratio between the unlabeled positive and negative samples is known at the beginning of transductive learning. Inoue and Ueda [108] present a semi-supervised method for effectively using unlabeled sequential data in the learning of Hidden Markov models (HMMs). Chapelle and Zien [109] proposed a semi-supervised method that optimizes the SVM objective function by using gradient descent technique to find out the decision boundary in low density regions of the kernel space. Lorenzo and Percello presented a novel context-sensitive semi-supervised SVM classifier [110], aimed at addressing classification problems where the available training set is not fully reliable, i.e., some labelled samples may be associated to the wrong information class (mislabelled patterns). In [111] a novel method based on local spline regression is presented for semi-supervised classification. A novel semi-supervised classification algorithm based on tolerance of rough set and ensemble learning [112] has been developed for text classification. Su *et al.* in 2013 proposed a Semi-supervised learning paradigm for cell segmentation [113], which is a fundamental task for various cell behavior analysis. The semi-supervised fuzzy c-means algorithm and SVM are used for clustering and classification in [114] to develop better classifier. Hyper spectral image (HSI) classification is one of most important application domain where the high dimensionality and heterogeneity is major challenge. Camps *et al.* in [115] presented the summary of different type of SSL used for HSI. Ahmed *et al.* developed a new SSL paradigm in [116] with the help of frequent itemset and ensemble learning to classify Short Message Service (SMS) data into ham and spam.

1.2.1.2 Active Learning

In real time situations, it is very difficult to obtain proper labeled samples. This may increase the cost, time or redundancy when the training samples are selected manually. In order to decrease the cost, time, redundancy and to optimize the classifier performance, the small size, non-redundant and most informative training set should be considered to design classifier. Active learning techniques are the solution to this problem. It is a well defined statistical approach of machine learning. In passive learning, the learner has no interaction with the expert and the goal of learning paradigm is to produce accurate prediction only from the labeled sample data. On the other hand in AL, the learner requests the expert/oracle to label few unlabeled samples (most uncertain and non-redundant) to produce accurate prediction. It is an iterative learning paradigm where the supervisor annotates the newly selected unlabeled sample/s (most informative samples) iteratively with the help of query function as shown in Figure 1.6. Those selected samples are added to the training set to retrain the classifier and then process repeats as shown in Figure 1.6 till the stopping criteria meets.

The design of the query function is generally based on two principles: **i)** Uncertainty and, **ii)** Diversity. Uncertainty criterion determines the most uncertain samples from the unlabeled pool because the samples with the least classification confidence are most in-

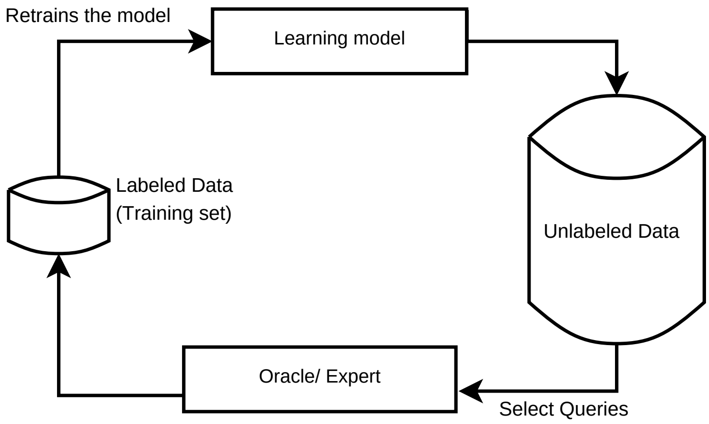


Figure 1.6: Active learning.

formative. Diversity criterion helps in determining the non-redundant unlabeled samples which reduces the cost and time. In semi-supervised learning paradigms, no expert like human annotator is required as the most confident unlabeled samples are selected. In contrast, active learning paradigms require an expert like human annotator which assigns the label to the most informative samples. Another major difference in SSL and AL paradigms can be observed in the selection criterion of unlabeled samples in each iteration. In semi-supervised learning techniques, the most confident unlabeled samples are selected in each iteration where as in active learning techniques the least confident unlabeled samples are selected in each iteration to retrain the classifier and to improve the accuracy of the classifier. Figure 1.7 shows the decision hyperplane and marginal hyperplanes obtained from SVM classifier using available labeled samples. In SSL technique, the most confident samples i.e. closest to marginal hyperplanes are selected to retrain the classifier where as in AL technique the least confident unlabeled samples i.e. nearest to the decision hyperplane are selected to retrain the classifier. Although the objective of both the paradigms is same i.e. to improve the accuracy of the classifier.

There exist number of variants of AL in literature [117] which can be categorised on the basis of query function: **1)** Committee-based paradigms **2)** Posterior probability based paradigms **3)** Large margin based paradigms. The AL principle works efficiently with the number of different learning paradigms like PCA [118], bayesian network [119, 120], feed forward neural network [121, 122], mixture of Guassians and weighted regressions [123], Support vector machines (SVM) [124],and so on . In [125] Begum *et al.* presented a novel AL technique for compound classification that can be used to detect land-cover transitions. A novel change-detection-driven transfer learning approach to update land-cover maps by classifying remote sensing images is presented in [126]. In article [127], a novel learning algorithm called active deep network (ADN) is proposed to classify

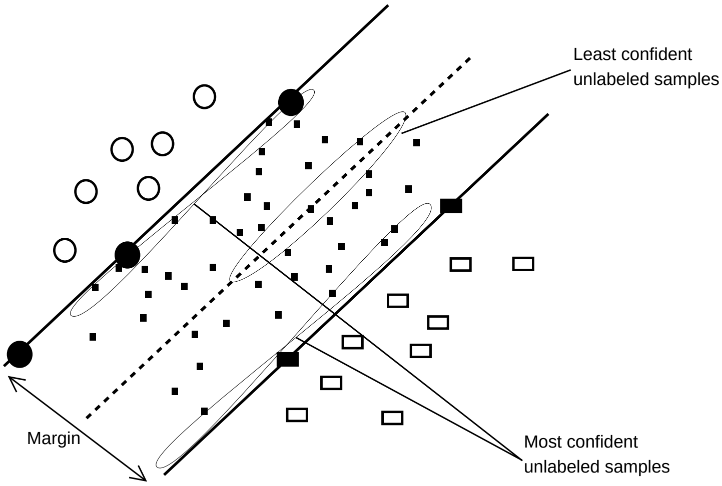


Figure 1.7: Representation of least and most confident unlabeled samples.

sentiments by employing active learning paradigm. Camps *et al.* [115] presented the summary of different type of AL methods as this paradigm has gained much popularity in last few years and focusses on difficult areas to improve the performance with fewer samples. In this dissertation, advanced active learning algorithm for pattern classification based on SVM is proposed.

1.2.2 Unsupervised Learning

In machine learning, unsupervised learning refers to the problem of determining hidden structure in unlabeled data. Since the examples given to the learner are unlabeled, there is no error or reward signal to evaluate a potential solution. This distinguishes unsupervised learning from supervised learning. It can be observed from the Figure 1.8 that all the samples are unlabeled and have no information about the class or category to which they belong. All the unlabeled samples are represented in feature space as if they all belong to same class. Further based on the similarities and dissimilarities among samples different clusters are formed as shown in Figure 1.8. In supervised learning, we have the prior information about the number of classes but in case of unsupervised learning no such prior information exist.

Unsupervised learning is closely related to the problem of density estimation in statistics. However unsupervised learning also encompasses many other techniques that seek to summarize and explain key features of the data. In this type of problem, we are given a set of feature vectors X and the goal is to untangle the underlying similarities and cluster (group) the similar vectors together. Clustering as discussed 1.1.3 is one of major form of unsupervised learning and used in many social sciences and engineering applications, such

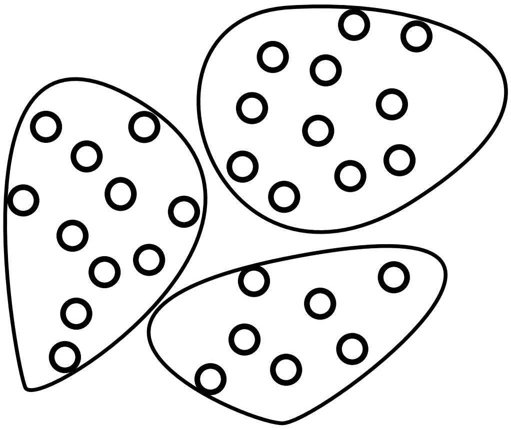


Figure 1.8: Patterns shown in the two-dimensional feature space are divided into three groups based on their similarities and dissimilarities.

as remote sensing, image segmentation, speech coding. Broadly there exist two types of clustering: hierarchical clustering and partition clustering [128]. Hierarchical clustering groups data objects with a sequence of partitions, either from singleton clusters to a cluster including all individuals or vice versa. Hierarchical procedures can be either agglomerative or divisive: agglomerative algorithms begin with each element as a separate cluster and merge them in successively larger clusters; divisive algorithms begin with the whole set and proceed to divide it into successively smaller clusters [69]. In contrast, partition algorithms find all the clusters simultaneously as a partition of the data and do not impose a hierarchical structure. Generally, hierarchical clustering algorithms have higher complexity than the partition clustering algorithms. In literature, a huge number of clustering algorithm has been developed in diverse fields. Among those K-means is one of the simplest partition algorithm widely employed in different applications. Anil K. Jain summarizes the K-means algorithm developments along with major data clustering approaches in [129]. [130] presented a novel MRI image segmentation technique based on fuzzy c-means clustering. Bruzzone *et al.* in [131] analyzed the difference image by considering the spatial-contextual information included in the neighbourhood of each pixel. In particular, an approach based on Markov random fields (MRFs) that exploits inter-pixel class dependency was presented. The said technique required knowledge of the statistical distributions of the changed and unchanged pixels in the difference image. A new multistage method using hierarchical clustering for unsupervised image classification is presented in [132]. Evolutionary algorithms also has number of applications in data clustering. A survey in 2009 presents the important aspects in context of evolutionary data clustering along with multi-objective evolutionary clustering [70]. A new energy conservation based clustering algorithm [133] has been proposed for wireless sensor networks in 2011. It determines suitable size of clusters, achieves equalization of node lifetimes and reduced energy consumption levels. An effective algorithm for the sparse space clustering [134] clusters the data-points that resides in the union of low-dimensional subspaces. This technique has an advantage of direct dealing with noise, sparse outlying and missing entries. The MinMax k-Means algorithm to minimize the maximum intra-cluster variance was proposed in [135]. Banerjee *et al.*, in [136] proposed cluster ensemble based method which combine multiple data partitions generated by different clustering algorithms into a single robust solution for land-cover classification of remotely sensed multi-spectral satellite images.

Unsupervised learning is also known as data driven approach where the paradigm is fully dependent on data set as a result the computational burden may increase with the increase in size of data set. This computational burden may be reduced by employing SSL paradigms discussed in Section 1.2.1.1. In this dissertation, the domain of image segmentation is given consideration to develop advanced algorithms using unsupervised learning which is explained in the following subsection.

1.2.2.1 Image Segmentation

An image can be represented in two-dimensional coordinate system say $f(x, y) = l_{xy}$, where l_{xy} denotes the feature value (intensity, depth, gray level etc) at spatial coordinate (x,y) [137]. Mathematically, a two-dimensional matrix can be used to represent an image in computer system. Each row and column position of matrix represents a pixel of the image and the value at each position of matrix represents the value of the feature captured for the given image [138]. Therefore, the digital image I can be represented by the Equation (1.1) given below:

$$I = \{l_{xy} | 1 \leq x \leq m, 1 \leq y \leq n\} \tag{1.1}$$

where $m * n$ is the size of the image. Digital Image processing involves the following six fundamental steps:

- Image Acquisition, where the electrical signal received from the imaging sensors are digitized to store the image in computer memory.
- Image Enhancement, where the objective is to process original image so as to produce new image which will be more suitable for a particular application. The enhancement processes are different for different type of images and broadly fall under two domains: Spatial and frequency which are out of the scope of thesis.
- Image Restoration, where the inverse process is applied to recover the original

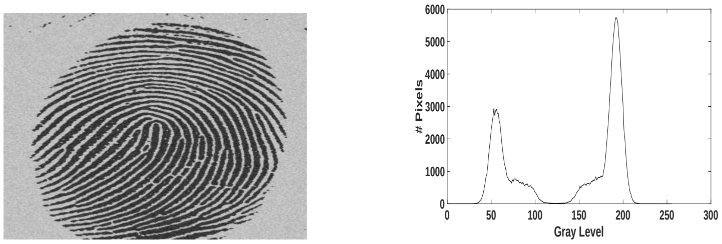
image.

- Image Compression ,where the space used to store the image is reduced by removing redundant data of the image used to store the digitized image.
- Image Segmentation defined as the process of assigning same label to those pixels that share certain common characteristics.
- Recognition that implies identification of object present in the image.

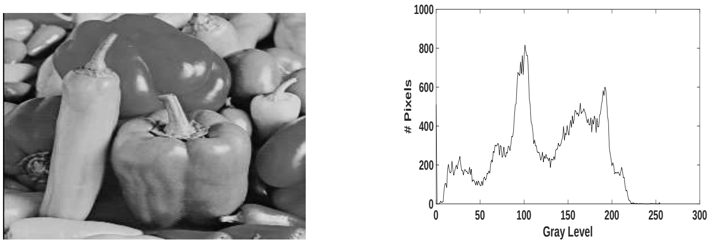
Image segmentation is another vast area of image processing where large number of algorithms exist [139, 140]. The solution of image segmentation problems is domain specific [141, 142, 143, 144]. It is defined as dividing regions homogeneously and uniformly based on certain characteristics say tone, gray level, texture etc. There exist wide variety of image segmentation techniques which can be classified in four categories: **a)** texture based where elementary and complex patterns are analyzed [145], **b)** thresholding based that considers the gray value of individual pixels [146], **c)** clustering based methods that look for discontinuities [147, 69]and **d)** region based methods that analyze the gray values in larger homogenous areas [148]. Thresholding is one of the effective technique for image segmentation and widely used in the area of image processing applications such as **1)** Optical character recognition (OCR), **2)** automatic visual inspection of defects, **3)** video change detection **4)** moving object segmentation **5)** medical imaging and so on. In literature, large number of thresholding techniques [149, 150] exists. In the present thesis, thresholding techniques have been designed to contribute towards the development of advanced techniques for PR.

Thresholding: Generally, different objects present in the image are represented with different intensities. The histogram of those images can be viewed as combination of different modes (peaks). Let us illustrate it with the help of an example. If only one object is present in the image then the pixels can be grouped into two groups say light and dark intensity. The histogram of such images will have two dominant modes where as if there are two objects in the image than the pixels will be grouped into three groups and it's histogram will have three dominant modes. Figure 1.9 shows two images: Fingerprint, and Peppers. It can be observed from the Figure 1.9(a) that it has only one object thus the histogram of this image has two dominant modes. In contrast the Figure 1.9(b) has more than one object present in it, thus the histogram of this image has more than two dominant modes.

The objective of thresholding is to identify such gray level (T) so that all the pixels with the gray-level greater than T will be assigned intensity of black and all the pixels with



(a) Fingerprint (b) Histogram



(c) Peppers (d) Histogram

Figure 1.9: (a) and (c) depicts the images fingerprint and peppers and (b) and (d) depicts the histogram of the respective images.

gray level below this T will be assigned intensity of white or vice-versa. The value of T is known as threshold and the process of determination of this threshold is known as thresholding. Mathematically, let l_{xy} denotes the gray level of pixel (x,y) of the image. The thresholded image f(x,y) is defined by Equation (1.2) given below:

$$\begin{aligned} f(x,y) = {} & 0 \;\; ; \;\; l_{xy} > T \\ = {} & 1 \;\; ; \;\; l_{xy} \leq T \end{aligned} \tag{1.2}$$

If single object is present in the image, thresholding will produce two homogenous regions hence known as bi-level thresholding. On the other hand, if $n > 1$ objects are present in the image, thresholding technique determines $n+1$ homogenous regions and the objective is to determine n thresholds to segment the image, hence known as multi-level thresholding. Thresholding methods can be further categorised broadly in six categories[146]. Those methods are briefly described as follows:

- Histogram shape-based: The peaks and valleys of the smooth histogram are analyzed by using various tools like convex or concavity analysis, wavelet transforms [14], spectral analysis [48] and so on.
- Clustering-based methods: The gray image is divided into two clusters, one cluster represents the background and the other one represents the object. Guassian model may also be employed in which gray image is modeled as mixture of two

Guassian distributions i.e. background and foreground. Number of different types of clustering techniques are employed for image segmentation [151]

- Entropy-based methods: Entropy of the distribution of gray levels in an image is analysed. Different entropy methods are used like maximization of entropy, cross entropy.
- Object attribute-based methods: Thresholding is performed based on the attribute similarity between gray level image and the segmented image. The attribute may be edges, shapes, intensity etc.
- Spatial methods: These techniques employ higher-order probability distribution, correlation functions, co-occurrence probabilities, two- dimensional entropy, random sets etc.
- Locally adaptive methods: The threshold values are dependent upon some local statistics like range, variance, contrast, gradient, and so on.

A good survey of different types of thresholding techniques have been presented in [150, 152, 146]. Thresholding technique can also be considered as optimization problem. In [153] Hammouche *et al.* presented comparative study of six optimization algorithms when applied to solve multilevel thresholding problem. A new multilevel algorithm based on the honey bee mating optimization (HBMO) algorithm is proposed in [154]. Another fast multilevel thresholding technique is cited in [155]. This technique is based on PSO which can be applied in complex image processing such as automatic target recognition system. In 2011, a novel histogram based thresholding technique [156] based on fuzzy C-mean clustering for color image segmentation was presented. It has low complexity and improves the compactness of the clusters forming uniform regions. Algorithm based on ABC was proposed in 2011 [157]. Multilevel thresholding for histogram-based image segmentation using modified bacterial foraging (MBF) algorithm is focussed in [158]. The Hybrid Thresholding Algorithm (HYTA) [159] exploits the benefits of the combination of fixed and adaptive thresholding methods. It transforms an input color cloud image into a normalized blue/red channel ratio image that can keep a distinct contrast, even with noise and outliers.

1.3 Preliminaries

Training Data: It is defined as 'experience' provided to the learning process. It is a collection of patterns $\{X_i\}_{i=1}^{n} = \{X_1, X_2, ..., X_n\}$ sampled independently.

Labeled and Unlabeled Data: Finite set of distinct values known as classes is defined as label. For instance, a ball may be a basket ball, volley ball, foot ball, tennis ball. Thus the finite set $\{Basketball, Volleyball, Football, Tennisball\}$ are the classes of ball. The classes are generally encoded by integer numbers given as $y \epsilon \{1, 2, ..., C\}$ where C is the number of classes. The sample which has the prior information about the class to which they belong are known as labeled data. In contrast, the samples which have no prior information about the class or category to which they belong are known as unlabeled data.

Test Data: To estimate the future performance of learning paradigm, a set of labeled data is kept separate. This set of labeled data is not used during training of the algorithm and hence provides unbiased estimation for future performance. This set of labeled data kept separately is known as test data.

Model Evaluation: One fundamental question that arise here is how do we measure the generalisation performance of a classifier or the capability of a learning algorithm. In literature, there exist large number of techniques to evaluate the efficiency of the learning paradigm. Broadly speaking, three main metrics to evaluate the efficiency of algorithm are: **i)** Accuracy to predict or classify, **ii)** Time complexity, and **iii)** Space complexity. The most common metric among all is accuracy.

- Accuracy:The percentage of number of samples in test data classified correctly gives the accuracy of the classifier. There are three main different approaches to accuracy estimation and model selection:
  - hold-out,
  - bootstrap,
  - K- fold Cross validation

  The present work employed cross validation to measure the robustness of the classification algorithm which is described below:

- K-fold Cross Validation: The original sample is randomly partitioned into k equal size sub-samples. Of the k sub-samples, a single sub-sample is retained as the validation data for testing the classifier, and the remaining k-1 sub-samples are used as training data. The cross-validation process is then repeated k times (folds), with each of the k sub-samples used exactly once as the validation data. Then those k results obtained from the folds are averaged (or otherwise combined) to produce a single estimation. The advantage of this method over repeated random sub-sampling is that all observations are used for both training and validation, and each observation is used for validation exactly once. In case of SVM, the cross validation process can be used to determine the control parameter denoted as C. A precise definition of the value of the C parameter is of extreme importance for the classification accuracy. If the value of C is too small, then the resulting discriminant function of the classifier may be poorly fit. On the other hand, if the value of C is quiet large then the classifier may overfit the data instances, thus resulting in low generalization capabilities.

1.4 Scope of the Thesis

The foundation of advanced PR techniques proposed in the present thesis can be categorised in three parts: **i)** Tranductive SVM based semi-supervised technique, **ii)** Active Learning technique, and **iii)** Thresholding techniques for image segmentation.

The first part of the present thesis deals with the development of semi-supervised pattern recognition technique. Supervised techniques require sufficient labeled samples to train the classifier, and the classification results rely on the quality of the labeled samples used for learning. However, the collection of labeled samples is time consuming and costly, and the available training samples are often not enough for an adequate learning of the classifier. Semi-supervised learning is a popular approach to deal with such problems. It exploits unlabeled data, together with the labeled data, to build better classifiers. As a result, under proper assumptions, it requires less human effort for labeling. Thus, it becomes of great interest both in theory and practice. In Chapter 2, we have investigated many semi-supervised learning techniques based on transductive SVM (TSVM) for pattern classification cited in literature [107, 109, 160, 161, 162, 163, 164]. It was observed that the semi-supervised techniques based on progressive transductive support vector machine (PTSVM) iteratively select transductive samples that are closest to the SVM margin bounds. More specifically, these techniques determine a decision hyperplane passing through low density region of the kernel space. This may result in selection of wrong patterns (i.e., patterns that when included in the semi-supervised learning can be associated with a wrong label) as transductive samples, especially when poor initial training sets are available or when available training samples are biased. To mitigate this limitation, a new technique has been proposed in Chapter 2. The proposed technique considers the distance from SVM margin bounds, the properties of the k-nearest neighbors approach, and the cluster assumption property of the data. To assess the effectiveness of the proposed method, we compared it with other PTSVM methods cited in the literature [162, 164]. Experimental results confirmed the effectiveness of the proposed technique.

The second part of the present thesis deals with the development of fast batch-mode active learning technique for pattern classification. Supervised learning algorithm needs proper labeled samples to train the classifier. The labeled samples called training set should fully represent the statistics of all classes. To classify patterns large number of supervised classifiers as discussed in Section 1.2.1 are cited in literature [165, 38, 78, 166, 167, 84]. Supervised learning has its own limitations: **i)** non-availability of adequate amount of labeled samples, **ii)** costly to collect the labeled samples, and **iii)** time complexity due to redundancy as explained in Section 1.2.1. To overcome such limitations AL is another approach. In Chapter 3, we have designed a fast batch mode AL technique which transforms the original feature space into 1-D feature space and then divides the unlabeled samples into different partitions generated around the mean of unlabeled samples. To select most informative samples, one sample from each non-empty partition is selected based on an uncertainty criterion defined by employing SVM classifier. The effectiveness of the proposed technique is measured by comparing it with four state-of-the-art techniques [168, 169, 170, 171] cited in the literature. The experimental analysis proved that the proposed technique is robust and computationally less demanding.

The third part of the thesis deals with the development of unsupervised PR technique to solve image segmentation problems described in Section 1.2.2.1. Unlike the above mentioned techniques, unsupervised methods do not require labeled samples for training. It uses similarity criterion to divide the patterns into different groups/clusters. Image segmentation is one of the most popular unsupervised technique researchers widely adopted to distinguish different objects present in the image. In the Chapters 4 and 5, advanced thresholding techniques for image segmentation has been designed. It has been observed that histogram based thresholding is one of the most popular method for image segmentation [146]. Due to the advantage of smaller storage space, fast processing speed and ease in manipulation, histogram thresholding techniques have drawn a lot of attention during the last couple of decades. Since thresholding is a well-researched field, there exist many algorithms for determining an optimal threshold of the image. Traditional thresholding techniques based on the histogram of the image suffered with three major limitations [172, 173, 174]: **i)** Unable to considered contextual information for selecting optimum threshold; **ii)** Inefficient for multilevel thresholding (computationally demanding and complicated to implement); and **iii)** Unable to find out optimal number of thresholds automatically.

In Chapter 4, we proposed a thresholding technique based on the energy curve of the image we defined in [175]. The energy curve has similar characteristics as that of histogram and unlike histogram of the image, it incorporates the spatial contextual information of the image. As a result, the energy curve of the image is smoother than that of histogram of the image and hence has better discriminatory capabilities. Using this energy curve, the proposed technique automatically determines the optimum number of thresholds for image segmentation. In the proposed technique, first concavity analysis technique [176] is applied on the energy curve to detect all the potential thresholds. Then a threshold elimination technique based on cluster validity measure is proposed to determine the optimum number of thresholds. To assess the effectiveness of the proposed technique the segmentation results obtained using energy curve of the image are compared with those obtained using histogram of the image. Experimental results exhibits the effectiveness of the proposed technique.

The technique presented in Chapter 4 employs concavity analysis technique [176]. Since concavity analysis technique may skip some valleys of the energy curve, it may fail to detect the optimal number of potential thresholds. To mitigate this limitation, a novel initial threshold selection technique is proposed in Chapter 5. In this Cshapter, a fast context sensitive threshold selection technique is presented to solve the image segmentation problems. In lieu of histogram, the proposed technique employs recently defined energy curve of the image for thresholds selection. It determines the number of potential thresholds and the bounds where the optimal value of each potential threshold may exist. Finally, employs GA to detect the optimal value of each potential threshold from their respective defined bounds. The proposed technique incorporates spatial contextual information of the image in threshold selection process without loosing the benefits of histogram based techniques. Computationally it is very efficient and able to determine the optimal number of segments in the input image. To assess the effectiveness of the proposed technique the results obtained are compared with four state-of-the-art methods cited in the literature [175, 177, 178, 179]. Experimental results on large number of images confirmed the effectiveness of the proposed technique.

Chapter 2

A Novel Classification Technique Based on Progressive Transductive SVM Learning [1]

2.1 Introduction

Supervised learning needs only labeled data for training. The classification results rely on the quantity and quality of these labeled samples. However the generation of proper labeled samples is often difficult, expensive and time consuming, as this requires the effort of experienced human annotators. On the other hand, unlabeled data are relatively easy to collect, although they have no use in supervised learning. Two popular machine learning approaches for dealing with this problem are active learning and semi-supervised learning. Active learning expands the original training set according to an interactive process that involves a supervisor who can assign the correct label to unknown data points [180, 175]. The goal of the active learner is to select the most informative samples so as to accurately learn from the fewest such additionally labeled data. In contrast, semi-supervised learning exploits unlabeled data, together with the labeled data, to build better classifiers [181]. As a result, under proper assumptions, it requires less human effort for labeling. Thus, it becomes of great interest both in theory and in practice.

The transductive support vector machine (TSVM) is a semi-supervised version of SVM [43]. During the training phase, it gradually (iterative process) searches a reliable separating hyperplane in the kernel space by taking into account both labeled and unlabeled samples. The unlabeled data can be used as an additional source of information about margin for SVM. In transductive learning the goal is to find a labeling of the unlabeled data, so that a linear boundary has the maximum margin on both the original labeled data and the (now labeled) unlabeled data (defined here as transductive data for convenience). This task can improve the generalization performance of SVMs, especially when poor training sets are available or when the available training samples are inadequate [107].

In the literature many semi-supervised techniques based on TSVM exist for pattern clas-

---

[1] This chapter has been published in *"Pattern Recognition Letters , Elsevier"*, Vol 42, pages 101-106, 2014 .

sification [107, 109, 160, 161, 162, 164]. All these methods try to find out a decision hyperplane passing through low density region of the kernel space. A good review of semi-supervised approaches can be found in [182]. Joachims in [107] solved the quadratic optimization problem for the implementation of the TSVM with an application to text classification. This algorithm is effective when the ratio between the unlabeled positive and negative samples is known at the beginning of transductive learning. Chapelle and Zien [109] proposed a method that optimizes the transductive SVM objective function by using gradient descent technique to find out the decision boundary in low density regions of the kernel space. Sindhwani and Keerthi [160] proposed a fast algorithm for linear TSVM, suitable for large scale text applications. In published literature [161], an additional criterion is included with the standard objective function of the TSVM and then a genetic algorithm is used for optimizing this objective function. A progressive TSVM algorithm was proposed in [162] that iteratively selects a positive and a negative sample as transductive samples from the available unlabeled samples that are inside the SVM decision margin and have minimum distance from positive margin and negative margin, respectively. The method also uses dynamic adjustment to reduce the miss labeling of selected transductive samples. Bruzzone *et al.* [164] modify the algorithm presented in [162] to select a batch of positive and negative patterns as transductive samples from the available unlabeled patterns at each iteration of the transductive learning process.

The progressive TSVM (PTSVM) algorithms existing in the literature [162, 164] iteratively select the most certain patterns as transductive samples from the available unlabeled patterns that are inside the SVM margin bounds. The certainty of a sample is measured by considering only its distance from the nearest SVM margin bound. This may result in a high probability of selecting wrong patterns as transductive samples (the actual labels of the patterns are different from the labels automatically assigned to them), especially when the initial decision hyperplane is poor i.e., passes through a wrong region of the kernel space. Thus, the final classification accuracy may be degraded. In this chapter, we developed a novel semi-supervised technique based on PTSVM learning that mitigates the above-mentioned limitation. The proposed technique not only uses the distance from the nearest SVM margin bound but it also exploits the properties of k-nearest neighbors (k-nn) approach and the cluster assumption to selects the most certain samples as transductive samples at each iteration of the learning process.

The rest of the chapter is organized as follows. The concept of inductive and transductive SVM learning is presented in Section 2.2. Section 2.3 describes the limitations of the existing PTSVM based approaches. The proposed technique is presented in Section 2.4. Section 2.5 provides the detailed description of the data sets used in the experiment and the results obtained on the considered data sets. Finally, Section 2.6 draws the conclusion of this work.

2.2 Support Vector Machine Classifier

Before presenting the proposed semi-supervised technique based on PTSVM learning, we briefly recall the main concepts associated with both the inductive SVM and transductive SVM learning. Among all the existing classifiers, support vector machines (SVMs) revealed one of the very effective and robust in the solution of many classification problems [183, 43]. The success of SVM in pattern classification is justified by three main general reasons: **(i)** their intrinsic effectiveness, with respect to traditional classifiers, which results in high classification accuracies and very good generalization capabilities (low sensitivity to the Hughes phenomenon [184]); **(ii)** the convexity of the objective function used in the learning of the classifier, which results in a unique solution (i.e., the system cannot fall into sub-optimal solutions associated with local minima); and **(iii)** the possibility of representing the convex optimization problem in a dual formulation, where only non-zero Lagrange multipliers are necessary for defining the separation hyperplane (which is a very important advantage in the case of large datasets).

2.2.1 Inductive SVM Learning

Let $X = \{(x_1, ..., x_n)|x_i \in \Re^d\}$ be the set of n available training samples and $Y = \{(y_1, ..., y_n)|y_i \in \{-1, +1\}\}$ be the set of associated labels. The standard SVM learning also called inductive SVM (ISVM) learning, tries to separate the data in the input space with the available training data by defining a hyperplane:

$$f(x) : wx + b = 0 \tag{2.1}$$

such that the distance between the closest vectors to the hyperplane is maximum. The maximal geometrical margin generated by the hyperplane is

$$\varphi(w) = \frac{2}{\parallel w \parallel} \tag{2.2}$$

In the case of linearly non-separable training data, the objective function of the ISVM learning is to find out a hyperplane by solving the following quadratic optimization problem

$$
\begin{aligned}
& \max_{\alpha}\left\{\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}y_iy_j\alpha_i\alpha_jK(\mathbf{x}_i,\mathbf{x}_j)\right\}\\
\text{subject to:}\quad & \sum_{i=1}^{n}y_i\alpha_i=0\\
& 0\leq\alpha_i\leq C \qquad (2.3)
\end{aligned}
$$

where ξ_i and α_i represents slack variables and Lagrangian multipliers, respectively. K(., .) is a kernel function that implicitly models the classification problem into a higher dimensional space where linear separation between classes can be approximated, and C is a regularization parameter that allows one to control the penalty assigned to training errors [175].

2.2.2 Transductive SVM Learning

In the above framework, to address the issue of the available biased/poor training set, a transductive SVM approach has been proposed by involving unlabeled samples into the training phase [43]. Let $X^* = \{(x_1^*, ..., x_m^*)|x_i \in \Re^d\}$ be the set of m unlabeled samples and $Y^* = \{(y_1^*, ..., y_m^*)|y_i \in \{-1, +1\}\}$ be the corresponding predicted labels. At the initial iteration, the standard ISVM is used to obtain a separating hyperplane using the training set X only. Then, depending on the distance from this hyperplane labels are assigned to the unlabeled samples which are thus called semi-labeled data. After that, according to a defined criterion transductive samples chosen from the semi-labeled patterns are included into the original training set X. The resulting training set is used at the following iterations to find a more reliable discriminant hyperplane. This hyperplane separates (X, Y) and (X^*, Y^*) with the maximal margin and is derived as follows:

$$
\begin{aligned}
& \varphi(w)=\min_{w,\xi_i,\xi_u^*}\left\{\frac{1}{2}\parallel w\parallel^2+C\sum_{i=1}^{n}\xi_i+C^*\sum_{i=1}^{d}\xi_u^*\right\}\\
\text{subject to:}\quad & \forall_{i=1}^{n}: y_i(wx_i+b)\geq 1-\xi_i, \xi_i>0\\
& \forall_{i=1}^{m}: y_i^*(wx_i+b)\geq 1-\xi_u^*, \xi_u^*>0 \qquad (2.4)
\end{aligned}
$$

In order to handle the nonseparable training and transductive data, similarly to the ISVMs, the slack variables ξ_i and ξ_u^* and the associated penalty values C and C^* of both the training and transductive samples are introduced. In the learning process of the

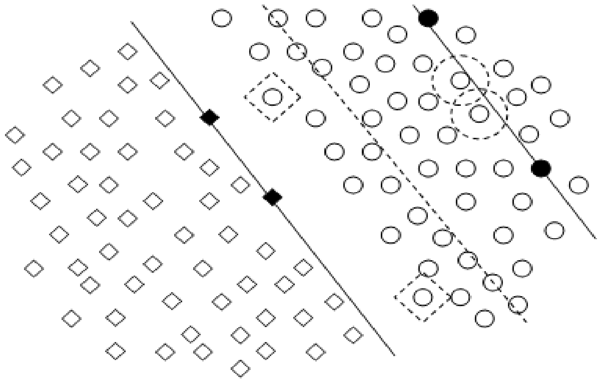
TSVMs, the purpose of C and C^* is to control the number of misclassified samples that belong to the original training set and to the unlabeled set, respectively. On increasing their values, the penalty associated with the errors on the training and transductive samples increases. In other words, the larger the regularization parameter are, the higher is the influence of the associated samples on the selection of the discriminant hyperplane. $d (d \leq m)$ is the number of transductive samples chosen at each iteration of transductive learning.

2.3 Limitations of the existing methods

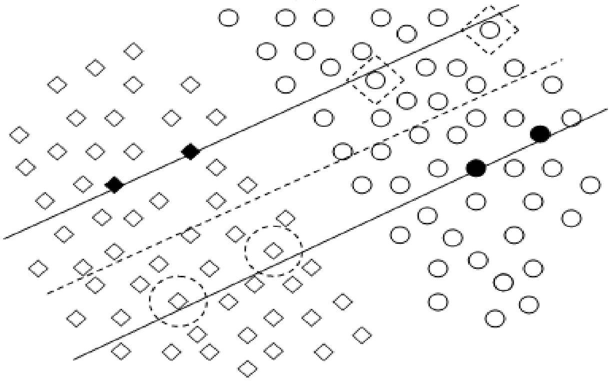
The semi-supervised techniques based on PTSVM existing in the literature [162, 164], iteratively select transductive samples from semi-labeled patterns that are inside and closest to the SVM margin bounds. This results in a high probability of selecting wrong patterns as transductive samples when the initial SVM decision hyperplane is very poor i.e., in cases where it passes through wrong region of the kernel space. Figure 2.1(a) depicts a situation where one SVM margin bound is near to the actual decision hyperplane (i.e., it passes through low density region of the kernel feature space) and the other one is far from it. Now, if we apply the conventional PTSVM methods to select transductive samples, they will select some semi-labeled patterns as transductive samples that also shift the margin bound which passes through low density region towards wrong direction. As a result a poor decision hyperplane will be generated. The main reason is that these algorithms do not impose any explicit constraint to the transductive sample selection process so that the margin bound which passes through high density region can only be shifted towards low density region. Figure 2.1(b) depicts another extreme situation where both positive and negative margin bounds pass through both the classes. Also in this case the conventional PTSVM methods may fail to select the appropriate transductive samples. As a result, the classification performance can be degraded. In this chapter we developed a novel semi-supervised technique based on PTSVM learning that addresses both the above-mentioned issues during the selection of trasductive samples at each iteration of the learning process.

2.4 Proposed TSVM Method

As regards the selection of transductive patterns, two points should be considered: **1)** choose the informative samples, and **2)** select the samples with an expected accurate labeling. In the proposed work the informative samples are selected by considering only the unlabeled patterns inside the SVM margin bounds. To select the proper transductive samples (with expected accurate labeling), we not only considered the distance from the SVM margin bounds, yet we also exploits k-nn technique and the cluster assumption. The details of the proposed transductive samples selection procedure is given below:



(a)



(b)

Figure 2.1: Patterns belonging to class "-1" and "+1" are shown as white squares and circles, respectively. Initial labeled samples for class "-1" and "+1" are shown as black squares and circles, respectively. The separation hyperplane is shown as a dashed line, whereas the solid lines define the margin. The dashed squares and circles highlight the transductive patterns selected by the 1st iteration of conventional progressive TSVM methods labeled as "-1" and "+1", respectively (a) when one decision margin is near to the actual decision hyperplane and the other one is far from it; (b) when both positive and negative decision margins pass through the feature space of both the classes.

Let us consider at the beginning only few training samples $X = \{(x_1, ..., x_n)|x_i \in \Re^d\}$; with their associated labels $Y = \{(y_1, ..., y_n)|y_i \in (-1, +1)\}$. Let us assume that a large number of unlabeled samples $X^* = \{(x_1^*, ..., x_m^*)|x_i \in \Re^d\}$ are available. First, we train the SVM classifier using the training set X only (inductive learning) to find out the decision hyperplane $f(x)$. Then, depending on the distance from this hyperplane, pseudo labels are assigned to all the unlabeled samples, thus obtaining semi-labeled samples. Let N be an integer variable. We fix the value of $N = min\{N_{SV}^+, N_{SV}^-\}$, where N_{SV}^+ and

N_{SV}^{-} represent the number of positive and negative margin support vectors, respectively, obtained after the first inductive learning iteration. After that, the set ψ^{+} of positive candidate transductive samples is initialized by choosing N semi-labeled samples from X^{*} that lie inside the margin bound and are closest to the positive margin. Similarly the negative candidate set ψ^{-} is initialized by choosing N semi-labeled samples from X^{*} that lie inside the margin bound and are closest to the negative margin. If we select all the semi-labeled samples in ψ^{+} and ψ^{-} as transductive samples as in the existing techniques then the problem shown in Figure 2.1(a) may arise (i.e., the decision hyperplane may shift towards wrong direction). To mitigate this kind of problem, here first we compute the average distance d^{+} and d^{-} of all samples in ψ^{+} and ψ^{-} from the SVM decision hyperplane as follows:

$$d^{+} = \frac{\sum_{x_i \in \psi^{+}} f(x_i)}{N} \tag{2.5}$$

$$d^{-} = \frac{\sum_{x_i \in \psi^{-}} |f(x_i)|}{N} \tag{2.6}$$

Then a threshold value t is obtained as $t = min\{d^{+}, d^{-}\}$. After finding the value of threshold t we update the positive candidate set $\psi^{+} = \{x_i | x_i \in \psi^{+}; f(x_i) > t\}$ and negative candidate set $\psi^{-} = \{x_i | x_i \in \psi^{-}; |f(x_i)| > t\}$. This help us to incorporate cluster assumption criterion to choose appropriate transductive samples. Due to the second problem shown in Figure 2.1(b), there may be some semi-labeled patterns that are closest to the positive (or negative) margin but belongs to ψ^{-} (or ψ^{+}). To select appropriate transductive samples from $\psi^{\pm}$, in this work we propose to apply a k-nn technique in the SVM kernel space. For each pattern $x_i^{*} \in \psi^{\pm}$, we find out the k-nearest labeled samples from X and then assign a class label to x_i^{*} according to k-nearest neighbor rule. After that according to the k-nn rule, the patterns from ψ^{+} and ψ^{-} selected as transductive patterns those have positive and negative class label, respectively. To compute the distance between a pattern $x_i^{*} \in \psi^{\pm}$ and $x_j \in X$ in the kernel space we use the same kernel function as used in the SVM classifier. Accordingly we try to select transductive samples that are most certain. The certainty of each unlabeled samples is measured by applying cluster assumption criterion and by exploiting the properties of the SVM and the k-nn techniques. Note that compared to the existing methods, the proposed technique spends some additional time to find out the k-nearest neighbors of each sample in $\psi^{\pm}$. Since a limited number of samples are in $\psi^{\pm}$, this does not take significant additional time. Let ψ_t be the set of transductive samples. Initially the set is empty. After selecting few semi-labeled samples as transductive samples we add them into the transductive set ψ_t and retrain the SVM classifier using the available labeled set X and ψ_t. The process is iterated until some patterns are inside the margin bound or a threshold value on the number of patterns in the margin is reached. If the label of a transductive pattern at iteration $itr + 1$ is different from the one at iteration itr, such pattern is removed from the transductive set and reassigned to the unlabeled set. The regularization parameter for the transductive patterns C^* is increased in a linear way, depending on the value of the regularization parameter for the labeled patterns C and a parameter g. In our algorithm, the initial value of C^* is defined as $C^* = g \times C$, where g is a weight parameter computed as $g = \frac{1}{l} \times itr$ (l is the growth rate and is a user defined parameter). In the first l iterations, the value of g increases linearly and reach value equal to 1 at l^{th} iteration. After that the value of g does not change by increasing the iteration number. It is worth noting that to solve multiclass problem we adopt one-against-all binary SVM architecture. Thus, if there are n classes then n TSVM are defined. Each TSVM solves a binary classification problem defined by one information class against all the others. The details of the proposed technique for solving multiclass problem is shown in Algorithm 2.1.

Algorithm 2.1: Proposed transductive learning technique

```
Let $X$ and $X^*$ denote the available labeled and unlabeled samples and $n$ represent the number of classes.
For s=1 to n
    Step 1: Set $itr = 0$ and transductive candidate set $\psi_t = \phi$.
    Repeat
        Step 2: Train the $s^{th}$ OAA binary SVM with the available labeled samples $X$ to
            generate the decision function $f_s(.)$.
        Step 3: Fix $N = \min(N^+_{SV}, N^-_{SV})$.
        Step 4: Define the positive candidate set $\psi^+$ and negative candidate set $\psi^-$ by
            selecting $N$ samples from $X^*$ that lies inside the margin bound and are closest to the
            positive and negative margins, respectively.
        Step 5: Compute average distances $d^{\pm}$ and threshold $t$ as follows:
            $d^+ = \frac{\sum_{x_i \in \psi^+} f(x_i)}{N}$, $d^- = \frac{\sum_{x_i \in \psi^-} |f(x_i)|}{N}$ and $t = min\{d^+, d^-\}$.
        Step 6: Update
            $\psi^+ = \{x_i^* \mid x_i^* \in \psi^+, |f_s(x_i^*)| \geq t\}$; $\psi^- = \{x_i^* \mid x_i^* \in \psi^-, |f_s(x_i^*)| \geq t\}$
        Step 7: For each $x_i^* \in \psi^{\pm}$, compute the distance to all the samples $x_j \in X$ in SVM
            kernel space. Then according to k-nearest neighbor rule assign either positive or
            negative class label to each $x_i^* \in \psi^{\pm}$.
        Step 8: Update
            $\psi^+ = \{x_i^* \mid x_i^* \in \psi^+$ and positive label assigned by $k - nn$ rule$\}$;
```

```
        $\psi^- = \{x_i^* \mid x_i^* \in \psi^-$ and negative label assigned by $k-nn$ rule$\}$
        Step 9: Update $\psi_t = \{\psi^+ \cup \psi^-\}$ and $X^* = X^* - \psi_t$
        Step 10: $itr = itr + 1$;
        Step 11: If $itr \leq l$ then update $g = \frac{1}{l} \times itr$ and $C^* = C \times g$.
        Step 12: Train $s^{th}$ binary SVM with updated labeled set $(X \cup \psi_t)$.
        Step 13: For a sample $x_i^* \in \psi_t$, if sign of $f_s^{itr}(x_i^*)$ and $f_s^{itr-1}(x_i^*)$ is different
         then update $\psi_t = \psi_t - x_i^*$ and $X^* = X^* \cup x_i^*$.
    Until $x_i^* \in X^*$ is found inside the margin bound or a threshold value on the number
    of patterns in the margin is reached.
End For
```

2.5 Experimental Results

2.5.1 Description of Data Sets

In order to assess the effectiveness of the proposed technique, seven data sets with significantly different properties were used in the experiment. The first one is a toy data set which is made up of four linearly separable classes as shown in Figure 2.2. The second one is a more complicated vowel data set [185]. The rest five data sets (Iris, Diabetes, Letter Recognition, Ionosphere and ISOLET) are taken from UCI machine learning repository among those that are widely used as benchmark for pattern classification tasks [186]. For all data sets, first only few available labeled samples were randomly selected as initial training set X, and the rest were stored in the unlabeled pool X^*. Table 2.1 shows the details of all the above mentioned data sets.

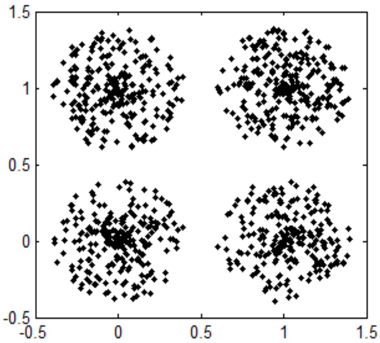


Figure 2.2: Linearly separable toy data set.

Table 2.1: Number of patterns, features and classes for the seven different data sets.

Data sets	**Patterns**	**Features**	**Classes**
Toy	1000	2	4
Vowel	871	3	6
Iris	150	4	3
Diabetes	768	8	2
Letter Recognition	20000	16	26
Ionosphere	351	34	2
ISOLET	6238	617	26

2.5.2 Design of Experiments

In our experiments we adopted an SVM classifier with RBF kernel. The SVM parameters $\{\sigma, C\}$ were derived by applying the cross-validation technique [187]. The cross-validation procedure aims at selecting the best values for the parameters of the initial SVM. The same RBF kernel function is also used to implement the kernel k-nn technique. The value of k for kernel k-nn is also automatically computed using cross-validation technique. To gradually consider the influence of transductive samples for defining the actual decision hyperplane, the initial value of the regularization parameter for the selected transductive patterns C^* should be small since at initial stage of learning the SVM decision hyperplane is poorly defined. At subsequent iterations, since labeling confidence increases, the value of C^* will also increase. This is done by defining the value of the growth rate parameter l. In our experiments, the value of growth rate l is fixed to 10 for all the considered data sets.

To assess the effectiveness of the proposed technique we compared it with two other existing methods: i) the progressive transductive SVM (PTSVM) method [162]; and ii) the modified progressive transductive SVM (MPTSVM) method [164]. In PTSVM, at each iteration of transductive learning, a positive and a negative semi-labeled pattern are selected as transductive samples from the available samples that are inside the SVM margin bound and are closest to the positive and negative margins, respectively. In MPTSVM, at each iteration, a batch of positive and negative semi-labeled patterns are selected as transductive samples using the same criterion as used in PTSVM. The batch size is determined automatically depending on the number of available positive and negative support vectors at a particular iteration of the learning process.

The multiclass SVM with the standard OAA architecture has been manually implemented by using the LIBSVM library (for MATLAB interface) [188]. All the algorithms presented in this chapter have been implemented in MATLAB.

2.5.3 Results

In order to understand the effectiveness of the proposed technique, in the first experiment we compared the performance of the proposed method with the existing PTSVM and MPTSVM methods. For all the four data sets, initially only few labeled samples were considered in the training set. The transductive learning process was repeated for 10 trials with 10 different initial training sets (generated randomly) to reduce the random effect on the results. Table 2.2 shows the average overall classification accuracies and standard deviations provided by different methods starting with different numbers of initial labeled samples for the toy, the iris, the diabetes and the vowel data sets. From the table one can see that for the toy data set, the proposed technique yielded a classification accuracy of 100% starting with only 8 initial labeled samples, whereas both the PTSVM (95.90%) and the MPTSVM (97.10%) techniques failed to achieve the same accuracy under the same conditions. For the vowel, the iris, the diabetes and the letter recognition data sets, the proposed technique starting with different numbers of initial labeled samples always resulted in higher classification accuracy than the other techniques. For example, considering the iris data set with 18 initial labeled samples, the proposed technique resulted in a 96.47% classification accuracy, whereas the best accuracy obtained by the existing literature methods is 94.87%. Similarly, for the vowel, the diabetes and the letter recognition data sets, one can see that the proposed technique always achieved at least 2% higher classification accuracies compared to those obtained by the literature methods under the same conditions. Moreover, from Table 2.2 one can see that the standard deviation of the accuracy provided by the proposed approach with respect to the different trials is always smaller than those of the other techniques. This confirms the better stability of the proposed method versus the choice of initial training samples. Since the proposed technique selects the transductive samples not only considering the distance from the SVM margin bounds, yet it also exploits the properties of the k-nn technique and the cluster assumption, it reduces the probability of selecting wrong transductive samples during the learning. Thus, it is more robust to solve classification tasks. For the ionosphere and the ISOLET data sets, the proposed technique only slightly improved the accuracy as compared to the literature techniques. This may be due to the fact that the cluster assumption criterion incorporated by the proposed technique may be less effective for these high dimensional data sets.

The second set of experiment was devoted to analyze the performance of the proposed technique by varying the value of the regularization parameter for transductive patterns C^*. As explained in Section 2.4, the initial and incremented (after completion of an iteration) value of C^* are computed based on the value of the user defined growth rate

Table 2.2: Average overall classification accuracy (acc) and its standard deviation (std) obtained on ten runs starting with different number of initial labeled samples ($|X|$).

Data sets	$\|X\|$	**Proposed**		**PTSVM**		**MPTSVM**	
		acc	**std**	**acc**	**std**	**acc**	**std**
Toy	8	100	0.00	95.90	3.68	97.10	1.83
Vowel	42	76.04	2.42	74.06	2.66	74.08	3.16
	56	77.93	2.14	75.92	2.66	75.98	2.85
Iris	9	95.12	3.83	93.47	5.21	93.53	4.69
	18	96.47	3.41	94.87	4.15	94.87	4.27
Diabetes	11	69.42	3.62	65.47	5.94	66.99	5.44
	22	70.68	3.87	67.59	7.23	67.50	6.86
Letter Recognition	280	68.93	2.79	65.81	2.58	66.32	2.17
	561	77.32	1.21	71.74	1.45	73.37	1.82
Ionosphere	54	89.56	2.46	88.80	3.30	89.46	2.67
	69	92.02	1.51	91.33	1.38	91.34	1.18
ISOLET	208	82.21	2.50	81.53	2.73	81.72	2.28
	338	87.04	1.52	86.27	1.78	86.48	1.67

parameter l. Thus, in this experiment for all considered data sets l was varied in the range 5, 10, 15 and 20. Table 2.3 shows the classification accuracies obtained by the proposed technique for different values of l. From the table one can see that the classification accuracy is not significantly affected. Thus, the value of C^* can be computed by assigning the value of l in wide range.

Table 2.3: Average classification accuracy provided by the proposed approach for different values of the user defined parameter l used to compute the value of C^*.

Data sets	$\|X\|$	**Proposed technique**			
		l=5	**l=10**	**l=15**	**l=20**
Toy	8	100	100	100	100
Vowel	56	77.71	77.93	78.12	77.83
Iris	18	96.42	96.47	96.58	96.51
Diabetes	22	70.75	70.68	70.62	70.57
Letter Recognition	561	77.27	77.32	77.05	76.96
Ionosphere	69	91.81	92.02	92.16	92.08
ISOLET	338	86.87	87.04	87.13	86.95

2.6 Discussion and Conclusion

In this chapter we have proposed a novel semi-supervised technique based on PTSVM for solving pattern classification tasks, which overcomes the limitations of the existing

PTSVM based methods. The existing techniques select the transductive samples by exploiting only the properties of the SVM classifier. They do not take adequately into account the low-density region of the feature space as well as the possible poor initial training set in the definition of the criterion for selecting transductive samples. As a result, the probability of selecting wrong transductive patterns (the actual labels of the patterns are different from the automatically assigned labels) becomes high and the classification performance may result degraded. To overcome this problem, the proposed technique not only exploits the properties of the SVM classifier, yet it also exploits a k-nn technique and the cluster assumption for selecting accurate transductive samples.

To empirically asses the effectiveness of the proposed method we compared it with other PTSVM based approaches existing in the literature using a toy data set and six real data sets. By this comparison we observed that the proposed method provided better accuracy compared to the existing techniques on the considered data sets.

Compared to the existing methods the proposed technique needs some additional time for using the k-nn technique. However, since the proposed technique computes the k-nearest neighbors of few samples this does not take significant additional time.

Chapter 3

A Fast Partition-based Batch-mode Active Learning Technique using SVM Classifier [2]

3.1 Introduction

Active learning (AL) is one of the machine learning techniques extensively used in recent years for pattern classification [123]. In AL paradigms, first the classifier is trained with the small number of labeled samples. Then the most informative samples are queried from the unlabeled pool and the training set is updated on the basis of a supervisor who assigns the labels to the selected samples. The steps of training and updating the training set are repeated until the most of the unlabeled samples can be classified with reasonably high confidence. As a result, the labeling of non-informative samples is avoided which increases the quality of the training set and reduces the labeling cost to the larger extent.

The basic motive of AL paradigms is to design a query function by incorporating a criteria which selects the most informative samples from the unlabeled pool for labeling. Numerous AL paradigms exist in the literature which can be broadly classified into three categories based on their query functions: **i)** Committee-based heuristics, where the learner uses disagreement among the committee members to determine the uncertain samples for labeling [189, 190, 191]. **ii)** Posterior probability-based heuristics, where the learner employs estimation of posterior probability density function for sample selection [123, 192, 105]. **iii)** Large margin-based heuristics, where the distance from the separating hyperplane generated by SVM classifier is used to select samples for labeling [193, 124, 194, 171, 195, 169, 196, 197, 198, 199]. Depending on the number of samples to be selected at each iteration, two paradigms of AL exist in the literature: **1)** Selects only one informative sample at each iteration [189, 123, 190, 191, 193, 192, 105, 124, 171] and, **2)** Selects a batch of informative samples at each iteration [170, 169, 200, 196, 197, 198, 199, 168]. To speed up the classification process, batch-mode AL techniques are more familiar in the pattern recognition literature.

---

[2] This chapter has been published in *"Soft Computing, Springer"*, 2017 .

Most of the batch-mode active learning techniques exist in literature select the most informative samples by using query function which first incorporates uncertainty criterion followed by the diversity criterion. The uncertainty criterion is associated with the confidence of the supervised algorithm for the correct classification of the considered samples while the diversity criterion aims at selecting a set of unlabeled samples that are as more diverse as possible in the feature space which reduces the redundancy among the samples selected at each iteration [198]. The batch-mode active learning techniques presented in the literature [170, 196, 168, 199, 169], first selects $m(m > 1)$ most uncertain samples from the unlabeled pool U which are closest to the one-against-all SVM decision hyperplanes. Then the batch of $h(1 < h < m)$ samples is selected from the already selected m most uncertain samples by applying a diversity criterion. Brinker in [170] proposed a diversity measure that consider the angles between the induced classification hyperplane. Tuia *et al.* in [169] incorporated the diversity criterion in active learning technique by selecting samples which are closest to distinct support vectors. Clustering-based diversity measures have been incorporated in [196, 199] to design the query functions for active learning techniques. Huo and Tang [168] proposed a diversity criterion by partitioning the one-against-all SVM decision hyperplanes. Since all the above mentioned techniques first select m most uncertain samples based on current SVM decision hyperplanes and then apply the diversity criteria to select h most informative samples to retrain the classifier, there is high possibility of biased shifting of SVM decision hyperplanes when the poor initial labeled samples are available (i.e. if the initial decision hyperplane is poor). In this work, the authors proposed a novel active learning technique which is fast and robust to select most informative samples.

The proposed active learning technique first employs a novel partitioning method which partitions all the unlabeled samples around the mean of the unlabeled pool U in one-dimensional (1-D) space. This partitioning of unlabeled samples is independent of the classifier decision hyperplane. After partitioning, at most one sample from each non-empty partition (this helps to select diverse samples) which is nearest to the SVM decision hyperplane (this helps to select most uncertain samples) is selected for labeling.

The contribution of the proposed algorithm is summarized in threefolds: **i)** The proposed technique first partitions the unlabeled pool U so that the informative samples are selected from all the uncertainty regions of the feature space. This mitigates the biased shifting of SVM decision hyperplane when the poor initial labeled samples are available, i.e., the initial decision hyperplanes are poor. **ii)** The partitioning method employed in the proposed technique, partitions the unlabeled data in 1-D space, i.e., transforms the original feature space into 1-D feature space. Hence, the diversity criterion used to define query function of the proposed technique is faster than the existing criteria. As a result, proposed technique is computationally less demanding. **iii)** In most of the existing batch-mode AL techniques, the number of samples selected at successive iterations is fixed and given as input by the user. There exist no specific criteria to decide how many samples should be selected at each iteration to retrain the classifier. In the proposed technique, the number of samples selected at successive iteration depends on the number of non-empty partitions generated and determined automatically. The experimental results of proposed technique using four different UCI repository data sets compared with four different state-of-the-art techniques show the effectiveness of the proposed technique.

This chapter is organized into six sections. In Section 3.2, a brief overview of active learning paradigms have been recapitulated. Section 3.3 presents the overview of SVM classifier. The proposed technique is presented in Section 3.4. Section 3.5 presents the detailed description of the experimental settings and the analysis of results obtained. Section 3.6 concludes the proposed work and the ways ahead in future.

3.2 Active Learning

Active learning is an iterative learning procedure where the supervisor repeatedly annotate new sample/s selected by the query function. The AL paradigm can be represented by quintuple (G, Q, S, L, U) [201]. The classifier G is trained on the samples present in the training set (L). The query function Q selects the most informative samples from the unlabeled pool U. The supervisor S assigns the true class label to the samples selected using Q from U. Those newly classified samples are added in the pool L. Further the classifier G retrains the updated training set L and the steps of querying and retraining are repeated till the stopping criterion is met. The generalized form of AL process is presented in Algorithm 3.1

Algorithm 3.1 Standard AL algorithm

```
The classifier $G$ is trained with the initial training set $L$.
Classify the pool of unlabeled samples in $U$.
repeat
    Using query function $Q$, query a set of samples from the pool $U$.
    The selected samples are labeled manually by the supervisor $S$.
    The newly labeled samples are added to the training set $L$.
    The classifier $G$ is retrained with updated training set $L$.
until stopping criterion is satisfied.
```

3.3 SVM Overview

The authors present a fast batch-mode AL technique to solve pattern classification task using SVM classifier. In the proposed technique, the SVM classifier is used for both the classification purpose and the uncertainty criterion involved in the design of query function of AL. SVM is a supervised classifier based on statistical learning theory [202] and has excellent performance in many real-world applications. The set of support vectors is the main characteristic of SVM classifier [166].

SVM is a supervised binary classifier that aims to separate the d-dimensional input feature space into two sub-spaces(one for each class) using a separating hyperplane. Assume that the training set consists of N labeled samples $(x_i; y_i)$ for i = 1 to N such that $x_i \epsilon \Re^d$ and $y_i \epsilon \{+1, -1\}$ denotes the associated labels which represent classes ω_1 and ω_2. Then SVM finds an optimal hyperplane which separates the two classes correctly and also maximizes the margin which is defined as the distance between the nearest point and the hyperplane. Thus the training phase of the SVM classifier can be standardized as an Lagrange's optimization problem and can be represented by following dual representation:

$$maximize \quad \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \phi(x_i) \phi(x_j) \tag{3.1}$$

$$subject\ to \quad \sum_{i}^{N} \alpha_i y_i = 0 \quad and \quad 0 \leq \alpha_i \leq C, \ \ \forall i = 1, \ldots, N \tag{3.2}$$

where α_i is the lagrange multiplier, C is a constant used to penalize the training errors of samples. According to Karush-Kuhn-Tucker conditions [203], most α_i is zero, only the training data that lie on or inside the margin plane have non-zero α_i, and such samples are known as support vectors(SV). The function $\phi(x)$ is a mapping function that transforms the training pattern x from its original feature space to a higher dimensional feature space and the dot product $\phi(x_i)\phi(x_j)$ given in Equation (3.1) is computed in the transformed feature space. Two major problems may arise to compute the dot product between the pairs of training patterns in the transformed feature space. First, how do we choose the mapping function $\phi(x)$? Second, the computations involved to compute the dot product in higher dimensional space are heavy. Both these problems can be solved using a mathematical trick known as kernel trick. Instead of computing the dot product on the transformed higher dimensional feature space, it turns out that it is mathematically equivalent to apply a kernel function, $K(x_i, x_j)$, to the original input feature space, i.e., $(K(x_i, x_j) = \phi(x_i)\phi(x_j))$. In other words, wherever the expression of dot product of

$\phi(x_i)\phi(x_j)$ appears in the training algorithm, one can replace it with $K(x_i, x_j)$ [204]. Mercer conditions [205, 202] ensure about the candidate kernel whether it is dot product in feature space or not. If the candidate kernel is not a dot product that means it is not admissible for use in SVM. For a given test sample x, the decision function $f(x)$ is computed as:

$$f(x) = \sum_{x_i \epsilon SV} \alpha_i y_i K(x, x_i) + b \tag{3.3}$$

where b is the perpendicular distance of the hyperplane from the origin, known as bias. The sign of the decision function $f(x)$ defined in Equation (3.3) is used to predict the class label of x. For more details, one can refer to [206, 207, 208]. SVM can also be used for multiclass problems by adopting ensemble of binary SVMs such as: one-against-all (OAA) and one-against-one (OAO). In this work, we adopted the OAA architecture which is a parallel architecture made up of n binary SVMs, one for each information class. Every SVM solves a two-class problem defined by one information class against all rest of the classes. Thus, if we have n classes, for a test sample x, n functional distances $\{f_1(x), f_2(x), ..., f_n(x)\}$ will be obtained. Now the pattern x will be assigned to j^{th} class if $j = \arg \max_{i=1,2,...,n} \{f_i(x)\}$.

3.4 Proposed Technique

The authors proposed a fast batch-mode active learning technique by defining a novel query function that selects variable number of most informative samples from the unlabeled pool U to retrain the SVM classifier at successive iterations of AL. As mentioned before, most of the batch-mode SVM-based AL techniques existing in literature [170, 196, 168, 199, 169], first, select $m(1 < m)$ uncertain samples based on the current SVM decision hyperplanes. Then applies the diversity criteria to select h informative samples from m samples for labeling. Such techniques suffer with the limitation of the biased shifting of hyperplanes, when the poor initial labeled samples are available. As a result, more samples need to be labeled to obtain better classification results. To mitigate this limitation, in the present work, we proposed a fast and novel partitioning method to partition the available unlabeled samples in such a way so that the diverse samples can be obtained from the whole feature space. The novel partitioning method proposed in the present technique is elucidated in next subsection.

3.4.1 Proposed partitioning method

In order to select the diverse unlabeled samples from the unlabeled pool U and to decrease the redundant samples among the selected samples, we proposed a novel partitioning method that divides the available unlabeled samples into different partitions in the feature space. Let $U = \{x_1, x_2, x_3, ..., x_l\}$ be the pool of unlabeled samples. The mean of unlabeled pool U say, $\bar{x}_m$ is computed as $\bar{x}_m = \frac{1}{l}\sum_{i=1}^{l} x_i$ and the set $D = \{d_1, d_2, ..., d_l\}$ is obtained by computing the distance of each unlabeled sample $x_i \in U$ from the mean $\bar{x}_m$. The minimum, average and maximum values in D say d_{min}, d_{mean} and d_{max}, respectively, will be used to partition the samples in U. To this end, first the elements in D are sorted and then divided into two subsets D_1 and D_2 such that D_1 consist of all those unlabeled samples with distance within $[d_{min}, d_{mean}]$ and D_2 within $(d_{mean}, d_{max}]$. These two subsets D_1 and D_2 help us to adjust the width of each partition. The width of each partition of unlabeled samples in the subsets D_1 and D_2 represented by W_1 and W_2 is computed as:

$$W_1 = \frac{1}{|D_1|} \sum_{i=2, d_i \in D_1}^{|D_1|} (d_i - d_{i-1}) \tag{3.4}$$

$$W_2 = \frac{1}{|D_2|} \sum_{i=2, d_i \in D_2}^{|D_2|} (d_i - d_{i-1}) \tag{3.5}$$

Thus the width of partitions of each subset D_1 and D_2 can be formed using Equations (3.6) and (3.7) given below:

$$L_P(i) = d_{min} + (i-1)W_1 \quad and \quad U_P(i) = d_{min} + (i)W_1 \tag{3.6}$$

$$L_P(i) = d_{mean} + (i-1)W_2 \quad and \quad U_P(i) = d_{mean} + (i)W_2 \tag{3.7}$$

where L_P and U_P represent the lower and upper bounds of each partition. Let P be the set of partitions generated by using Equations (3.6) and (3.7). Now each non-empty partition P_i may consist of different number of samples depending on the width of the partition and the distance from the mean sample $\bar{x}_m$.

For illustration,consider the randomly generated unlabeled pool U of 29 samples with two-dimensional feature space as shown in Figure 3.1. Following steps illustrate the details of the proposed partitioning method:

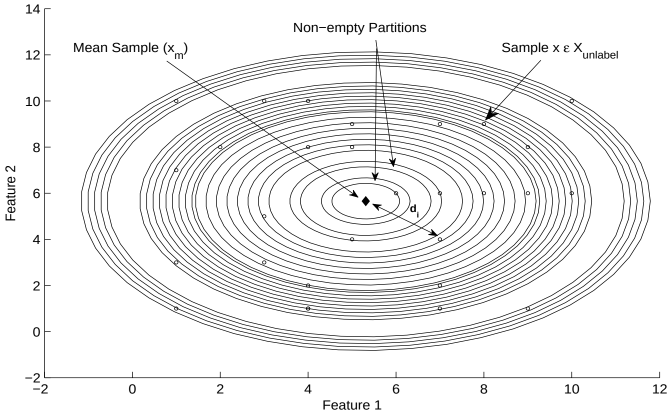


Figure 3.1: Illustration of the proposed partitioning method.

- Compute the mean sample $\bar{x}_m$ of U i.e. $\bar{x}_m = (5.31, 5.66)$, represented by the diamond in Figure 3.1. Compute $|d_i|$ of each sample from the mean sample $\bar{x}_m$. The set D consist of 29 absolute distances from $\bar{x}_m$. Sort D and compute d_{min} and d_{max} obtained as 0.77 units and 6.39 units which provides the information that the first partition can be found 0.77 units and the boundary of last partition can be found 6.93 units far from $\bar{x}_m$. The computation of distance from the mean sample $\bar{x}_m$ of unlabeled samples basically transformed the original feature space into 1-D feature space which reduces the complexity and determines the diverse(non-redundant) samples faster.

- To determine the width of each partition, the set D is divided into two subsets such that D_1 consist of all the distances less than or equal to d_{mean} i.e $D_1 \epsilon [0.77, 3.96]$ and D_2 with the distances greater than d_{mean} i.e $D_2 \epsilon (3.96, 6.39]$. By using Equations (3.4) and (3.5), we obtain two widths of partitions say, $W_1 = 0.239$ units for the samples with $d_i <= d_{mean}$ and $W_2 = 0.148$ units for the samples with $d_i > d_{mean}$. The boundaries of each partitions are generated using Equations (3.6) and (3.7).

- Once the partitions are obtained, the unlabeled samples belong to U are mapped to their respective partitions. The partitions with no samples are completely discarded. The non-empty partitions with in the range $[d_{min}, d_{mean}]$ have width of approximately 0.239 units and those with in the range $(d_{mean}, d_{max}]$ have width 0.148 units approximately. It can also be observed that the total number of partitions formed were 30 but the non-empty partitions were only 17. In Figure 3.1 only those 17 non-empty partitions are represented by using concentric circles around the mean sample x_m.

3.4.2 Selection of informative samples

In AL the goal of the query function is to select most informative samples from the unlabeled pool U, so that minimum number of unlabeled samples need to be annotated without compromising classification accuracy. In the proposed query function, the diversity criterion is incorporated by selecting at most one unlabeled sample from each non-empty partition. Diversity criterion alone is not sufficient to design a good query function. Uncertainty criterion that identifies the most uncertain samples (samples which have lowest classification confidence) plays significant role in query function. In this work, SVM with OAA architecture has been exploited to incorporate uncertainty criterion in query function. As explained before, in OAA architecture if we have n classes, n decision hyperplanes $\{f_1(), f_2(), ..., f_n()\}$ one corresponds to each information class is obtained. Considering these n decision hyperplanes, several techniques exist in the literature to define uncertainty criterion. Among them, the well-known marginal sampling (MS) approach [194] is used in the proposed technique. In MS approach, the sample with the smallest distance to any of the decision hyperplane is considered to be most uncertain and determined as follows:

$$x_j = arg \min_{x_k \in U} \{ \min_{i=1,...,n} \{|f_i(x_k)|\}\} \tag{3.8}$$

where $|f_i(x_k)|$ represents the absolute value of $f_i(x_k)$. In the proposed technique, a batch of informative samples are chosen by selecting one most uncertain sample from each non-empty partition. In details, let $\{P_1, P_2, P_3, ..., P_k\}$ be the k non-empty partitions of the unlabeled pool U obtained as described in Section 3.4.1. Then k most informative samples one from each partition is selected as follows:

$$x_l = arg \min_{x_j \in P_l} \{ \min_{i=1,...,n} \{|f_i(x_j)|\}\} \quad l = 1, 2, ..., k \tag{3.9}$$

After selecting a batch of informative samples using the proposed diversity and uncertainty criteria, a true label is assigned to the selected samples and the SVM classifier is retrained with the updated training set. The process is repeated till the stop criterion is satisfied. The generalized form of the proposed technique is given in Algorithm 3.2.

Algorithm 3.2 Proposed AL algorithm

```
Input:
    L:The labeled training set.
    $U$:The unlabeled sample pool.
    G:The supervised classifier (SVM).
    Q:The query function.
    Stopping Criterion
Begin:
The classifier G is trained with the initial training set.
Classify the unlabeled samples of the pool $U$.
repeat
    Divide the unlabeled pool $U$ into partitions $\{P\}$ as described in Sect. 3.4.1.
    Discard the empty partitions.
    Using proposed query function $Q$, query a sample from each non-empty partition $P_i$ as described in Sect. 3.4.2.
    Assign label to the selected samples with the help of the supervisor $S$.
    Add newly labeled samples to the training set $L$ and update the $U$.
    The classifier $G$ is retrained with the updated training set $L$.
until (Stopping criterion)
```

3.5 Design of Experiments

3.5.1 Data Sets

To visualize the effectiveness and robustness of the proposed technique, four different UCI data sets have been used. The first data set which is widely used for classification of patterns is Ionosphere. It was collected by Goose Bay, Labrador and consist of 351 free electrons patterns with 34 input features and two overlapping classes. The second famous UCI Iris Plants data set has 3 classes of Iris plants with 50 instances of each class. One class is linearly separable from the other two classes, but the rest two are not linearly separable from each other. The third UCI data set is Pima collected by Vincent Sigillio, The John Hopkins University. It consists of 768 patterns of Pima Indian Heritage female patients with age 21 years or above. Each instance has 8 attributes and two overlapping classes: tested negative or positive for diabetes. The fourth UCI data set named Vehicle was gathered at the TI in 1986-1987 by JP Siebert. This data set is used to discriminate between silhouettes of model cars, vans and buses and consists of 946 instances of vehicles. Each instance has 18 attributes and can be classified in four classes: OPEL, SAAB, BUS, and VAN .

For the considered data sets, instances were randomly divided into the training and test sets represented as T and TS respectively. The subset of instances of the training set T selected randomly as initial training set L and the remaining instances were considered under the unlabeled pool U. Table 3.1 provides the description of each data set along with the samples for different classes in corresponding learning sets (L, TS, U).

Table 3.1: The number of instances for each class in the: initial training set (L), test set(TS) and unlabeled pool(U) for the considered UCI data sets

Data Set	**Classes**	**No.of Samples**		
		L	TS	U
Iris	Setosa	2	15	33
	Versicolour	2	15	33
	Virginica	2	15	33
	Total	6	45	99
Ionosphere	good	8	225	150
	bad	4	126	84
	Total	12	351	234
Pima	Non-Diabetic	4	150	346
	Diabetic	2	80	186
	Total	6	230	532
Vehicle	OPEL	3	64	145
	SAAB	3	65	149
	BUS	3	65	150
	VAN	3	60	136
	Total	12	254	580

3.5.2 Experimental Settings

In the experiments, the authors have adopted SVM classifier with radial basis function (RBF) kernel. The SVM parameters $\{\sigma, C\}$ were derived by applying a grid search according to a cross-validation technique. To assess the effectiveness of the proposed technique, it has been compared with the four state-of-the-art SVM-based batch-mode active learning techniques: **1)** the MS-RP [168], **2)** the MS-cSV [169], **3)** the MS-ABD [170], and **4)** the MS [171]. In all these techniques, the SVM classifier is used for both the purposes: **i)** classification and, **ii)** to incorporate the uncertainty criterion to design query function. In the MS approach, a batch of $m(m > 1)$ unlabeled samples closest to the one-against-all SVM decision hyperplanes are selected for labeling at each iteration of AL. On the other hand, after selecting m most uncertain samples by employing MS criterion, the MS-RP, MS-cSV and MS-ABD techniques select $h(h < m)$ informative samples from the m samples by using XOR-based partition, closest support vector and cosine angle-based diversity criterion, respectively. To compare the proposed technique with the existing batch-mode state-of-the-art active learning techniques, the batch size is taken as 8. All the active learning algorithms have been implemented in MATLAB (R2012a). The LIBSVM library [209] for MATLAB interface has been used to implement multiclass SVM with OOA architecture.

3.5.3 Results Analysis

To exhibit the effectiveness of the proposed technique, three different experiments were performed. The first experiment exhibits the robustness of the proposed technique by comparing its performance with the four other state-of-the-art techniques on the four real data sets. To compare the performance of proposed technique with the existing techniques three metrics have been considered (i) average overall classification accuracy ($\overline{OA}$), (ii) its standard deviation(s) and (iii) average kappa accuracy ($\overline{k}$). Overall classification accuracy (OA) is defined as the ratio of the number of samples classified correctly to the total number of training samples present in the test set. Standard deviation of overall classification accuracies obtained for different runs measure the stability of the technique with respect to the initial available labeled samples. Low standard deviation indicates the robustness of the AL technique also when the biased initial labeled samples are available. The kappa accuracy (k) is another measure to compute how closely the samples in the test set are classified by the classifier. Mathematically, both overall classification (OA) and kappa (k) accuracies are computed by generating a confusion matrix of the test set. Let $CM = cm_{ij}, 1 <= i <= c, 1 <= j <= c$ be the generated confusion matrix for a test set containing N samples with c classes, where, cm_{ij} indicates the number of samples of class i that were labeled by the classifier as class j. Then overall classification accuracy and kappa accuracy is computed as follows:

$$OA = \frac{\sum_{i=1}^{c} cm_{ii}}{N} \times 100 \tag{3.10}$$

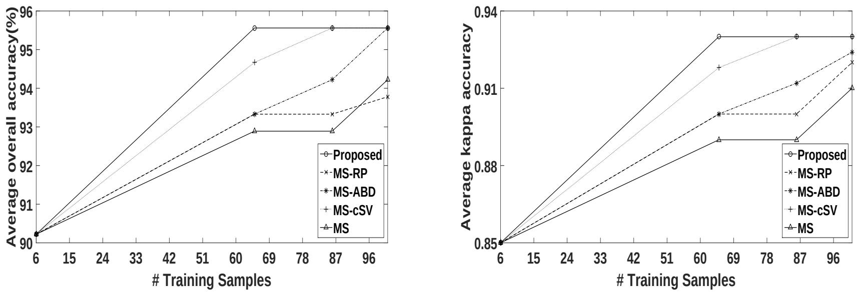
$$k = \frac{N \times \sum_{i=1}^{c} cm_{ii} - \sum_{i=1}^{c}(\sum_{j=1}^{c} cm_{ij} \times \sum_{j=1}^{c} cm_{ji})}{N^2 - \sum_{i=1}^{c}(\sum_{j=1}^{c} cm_{ij} \times \sum_{j=1}^{c} cm_{ji})} \tag{3.11}$$

In the experiment the proposed active learning process was repeated for 15 runs each with different randomly generated initial training sets. Since the number of samples selected at successive iterations depend on the number of non-empty partitions generated around the mean of the unlabeled samples. All the 15 training sets generated randomly may not have same number of samples selected at each iteration. Thus, those five training sets, where the number of samples selected is same (with ±1 difference) have been considered to compute the average overall classification accuracy ($\overline{OA}$) and average kappa accuracy

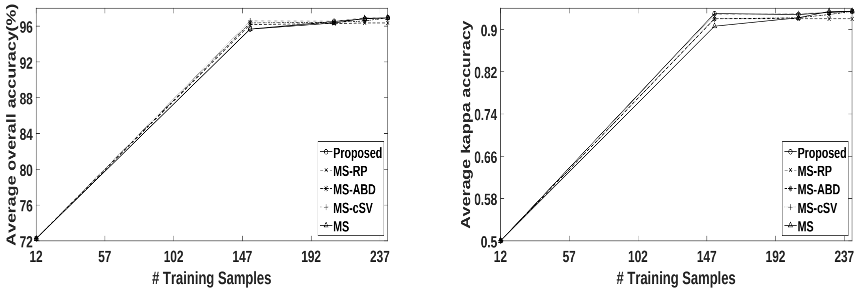
($\overline{k}$) for each data set. Table 3.2 reports the results of average overall classification accuracy ($\overline{OA}$), its standard deviation(s) and the average kappa accuracy ($\overline{k}$) for the given number of samples selected to retrain the model. Figure 3.2 presents the average overall classification and kappa accuracies achieved by the proposed and the existing techniques versus the number of samples included in different iterations for the considered data sets. It can be observed that in the iris data set the average overall classification accuracy has been improved by 2.38% and kappa accuracy has been improved by 3.3% when the training sample size is 65 and has been achieved in first iteration of the proposed algorithm. In case of ionosphere data set, although the accuracy is comparable but the variations in accuracy for five runs of the algorithm is very less as standard deviation is comparatively low with respect to other existing techniques. Similarly for other data sets, the proposed technique produced better results than the existing techniques.

To validate the robustness of the proposed technique, very less number of samples in the initial training sets were considered as shown in Table 3.1. For the iris and pima data sets, the number of initial samples taken is only 6 and for the ionosphere and vehicle data sets, only 12 labeled samples were added randomly to the initial training set. The small size of initial training set with biased samples for each data set ensures that the initial SVM has very low accuracy as shown in Figure 3.2. This shows that the proposed technique is independent of the initial SVM as the non-redundant samples are selected by allocating equal weighted partition to each unlabeled sample followed by selection of most uncertain sample from each non-empty partition considering all the uncertainty regions of the feature space.

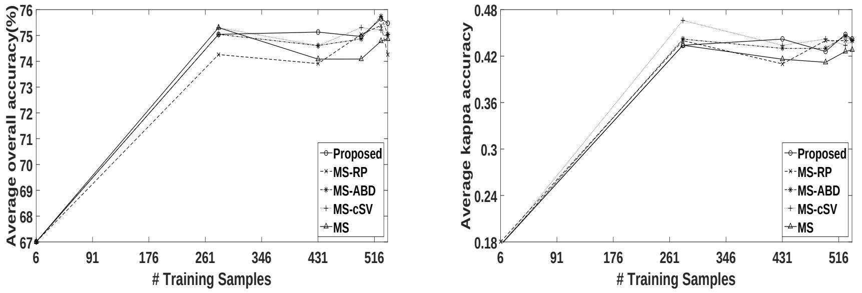
Second experiment measures the average computational time of the proposed and the existing techniques for the same training data sets considered for the first experiment. The experiments were carried out on a PC [INTEL(R) Core(TM)2 Duo 2.0 GHz with 2.0 GB RAM]. Table 3.3 shows the computational time in seconds taken by the investigated techniques for all the four data sets for the newly selected samples. From Table 3.3, one can observe that the required computational time has been reduced by minimum 3% and can be as high as 37% in many cases. The proposed algorithm takes very less computational time because the n-dimensional original feature space has been transformed into one-dimensional space and the diversity criteria incorporated using partitioning method is independent of the decision hyperplanes of the SVM classifier. Also the number of informative samples selected at successive iterations is not fixed but equals to the number of non-empty partitions formed using diversity criteria, which may be larger than the batch size fixed by user thus reduces the time complexity of the algorithm.



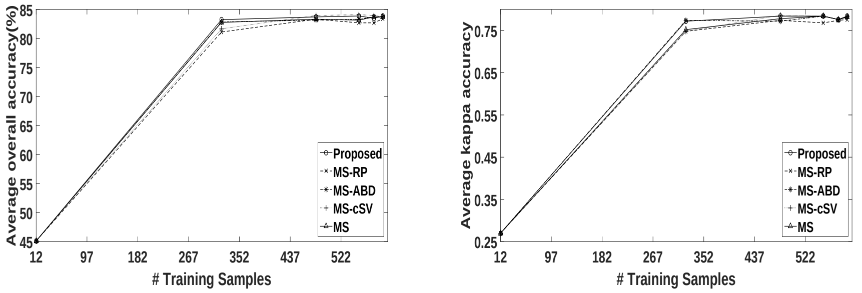
Iris



Ionosphere



Pima



Vehicle

Figure 3.2: (a), (c), (e), and (g) depicts the average overall classification accuracy ($\overline{OA}$). (b), (d), (f), and (h) depicts the average kappa accuracy ($\overline{k}$) versus the number of training samples obtained by the proposed, MS-RP MS-ABD, MS-cSV and MS techniques for the considered real data sets.

Table 3.2: Average overall classification accuracy ($\overline{OA}$), its standard deviations(s) and average kappa accuracy($\overline{k}$) obtained on five runs for different training data size at different iterations for the considered real data sets of the proposed and existing techniques: MS-RP, MS-ABD, MS-cSV, and MS. The best results obtained from the proposed technique are highlighted in boldfaces.

Data Set	$\lvert L\rvert$	**Proposed**			**MS-RP [168]**			**MS-ABD [170]**			**MS-cSV [169]**			**MS [171]**		
		$\overline{OA}$	s	$\overline{k}$	$\overline{OA}$	s	$\overline{k}$	$\overline{OA}$	s	$\overline{k}$	$\overline{OA}$	s	$\overline{k}$	$\overline{OA}$	s	$\overline{k}$
Iris	65	**95.56**	**0.00**	**0.93**	93.33	0.00	0.90	93.33	0.00	0.90	94.67	1.22	0.92	94.67	1.99	0.84
	86	**95.56**	**0.00**	**0.93**	93.33	0.00	.90	94.22	1.22	0.91	95.56	0.00	0.93	93.78	1.86	0.90
	101	**95.56**	**0.00**	**0.93**	93.78	1.86	0.92	95.56	0.99	0.92	95.56	0.00	0.93	94.67	1.99	0.92
Ionosphere	152	95.67	0.25	0.93	96.18	1.26	0.92	96.36	0.71	0.92	96.58	0.20	0.93	95.66	1.27	0.91
	207	**96.55**	**0.13**	**0.93**	96.29	0.45	0.92	96.41	0.33	0.92	96.55	0.69	0.93	96.35	0.24	0.92
	227	96.81	**0.24**	**0.93**	96.35	0.37	0.92	96.58	0.45	0.93	96.92	0.37	0.93	96.87	0.40	0.93
	242	**96.92**	**0.37**	**0.93**	96.35	0.37	0.92	96.92	0.37	0.93	96.92	0.37	0.93	96.92	0.37	0.93
Pima	281	75.04	**0.24**	0.43	74.26	2.03	0.44	75.04	3.32	0.44	75.30	1.25	0.47	75.30	0.84	0.43
	431	**75.13**	**0.78**	**0.44**	73.91	1.27	0.41	74.61	1.64	0.43	74.61	1.09	0.43	74.08	1.33	0.42
	496	74.96	**0.79**	0.43	75.04	1.0	0.44	74.87	1.32	0.43	75.30	0.36	0.44	74.09	1.40	0.41
	526	75.65	0.82	**0.45**	75.39	1.09	0.44	75.74	1.08	0.45	75.22	0.44	0.43	74.78	1.34	0.43
	536	**75.48**	**0.39**	**0.44**	74.26	0.89	0.44	75.04	0.50	0.44	75.04	0.50	0.44	74.87	1.28	0.43
Vehicle	322	**83.23**	0.87	**0.77**	81.10	1.34	0.74	82.84	1.54	0.77	81.65	0.53	0.75	82.76	1.61	0.75
	480	83.70	1.77	0.78	83.31	1.61	0.77	83.15	2.42	0.77	83.94	1.54	0.79	83.31	2.37	0.78
	522	83.86	**2.03**	**0.78**	82.68	1.21	0.77	83.31	2.10	0.78	84.09	2.24	0.78	83.15	2.11	0.78
	577	83.47	2.03	0.78	82.68	1.21	0.77	83.70	1.35	0.78	83.94	2.07	0.78	83.70	1.35	0.78
	592	**83.86**	**1.08**	**0.78**	83.39	1.40	0.78	83.70	1.35	0.78	83.70	1.35	0.78	83.70	1.35	0.78

Table 3.3: Average computational time(in seconds) over 5 runs of the proposed and existing active learning techniques for the considered real data sets.

Data Set	$\|L\|$	**Proposed**	**MS-RP**	**MS-ABD**	**MS-cSV**	**MS**
Iris	65	0.03	0.32	0.14	0.57	0.17
	86	0.05	0.46	0.34	1.14	0.38
	101	0.10	0.55	0.41	1.28	0.45
Ionosphere	152	0.12	7.70	2.23	5.70	2.41
	207	0.23	8.05	3.31	7.99	3.53
	227	0.32	8.64	3.74	8.82	3.97
	242	0.54	9.80	4.05	9.41	4.28
Pima	281	0.32	5.36	9.31	31.41	9.92
	431	0.72	11.75	26.21	77.56	27.03
	496	1.41	17.57	37.03	108.70	37.89
	526	2.28	20.47	42.51	124.65	43.35
	536	3.34	20.82	44.49	130.12	45.39
Vehicle	322	0.34	8.02	11.75	55.08	14.74
	480	0.75	17.06	28.47	110.07	33.46
	522	1.33	20.28	37.72	137.01	43.05
	577	2.02	23.42	41.19	146.92	46.60
	592	2.73	24.12	42.60	150.99	48.06

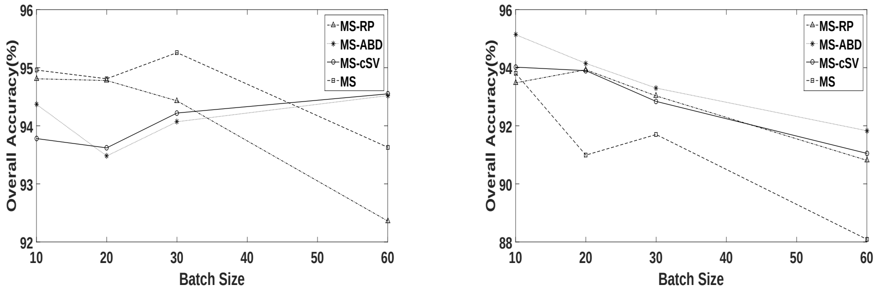
Third experiment validates the need to determine that how many samples from unlabeled pool at successive iterations should be selected to retrain the classifier. The proposed algorithm was repeated for 15 runs each with randomly generated different initial training sets. The batch size of the existing techniques is varied and the overall average classification accuracy is computed for the fixed number of training samples. Table 3.4 shows the average classification accuracy ($\overline{OA}$) by varying the batch size of the existing techniques: MS-RP, MS-ABD, MS-cSV, and MS. It can be observed that the average overall classification accuracy generally decreases when the batch size increases as shown in 3.3. This is because of the redundancy among the selected samples increases. Thus, selection of optimal batch size is a critical issue in the AL process. In the proposed technique, the partitioning method ensures that the number of unlabeled samples selected will be non-redundant even if the large number of samples are selected in an iteration. This is mainly due to the reason that from each non-empty partition one most uncertain sample is selected. Thus, in the proposed technique the automatic selection of batch of samples at each iteration of AL ensures that the technique is adaptive.

Table 3.4: Average overall accuracy ($\overline{OA}$) over batch size of the existing investigated active learning techniques for the considered real data sets.

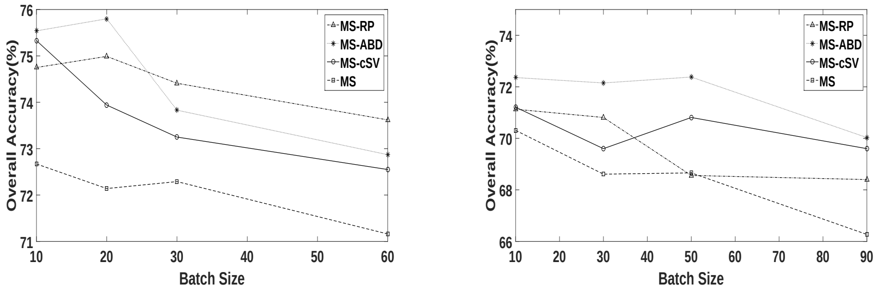
Data Set	$\|L\|$	**Batch Size**	**MS-RP**	**MS-ABD**	**MS-cSV**	**MS**
Iris	66	10	93.48	94.37	94.96	93.78
		20	93.94	93.48	94.81	93.62
		30	93.03	94.07	95.26	94.22
		60	90.81	94.52	93.63	94.55
Ionosphere	72	10	94.81	95.14	93.81	94.02
		20	94.78	94.15	90.99	93.9
		30	94.43	93.3	91.7	92.84
		60	92.36	91.83	88.09	91.05
Pima	126	10	74.75	75.54	72.67	75.33
		20	74.99	75.8	72.14	73.94
		30	74.41	73.83	72.29	73.25
		60	73.62	72.87	71.16	72.55
Vehicle	102	10	71.13	72.36	70.31	71.21
		30	70.81	72.15	68.61	69.6
		50	68.56	72.38	68.66	70.81
		90	68.40	70.03	66.27	69.6

3.6 Conclusion

The main objective of this work is to present a fast and efficient batch-mode active learning technique by combining a novel partitioning method with the intrinsic high generalization capabilities of SVM. In this chapter, we have designed a novel partitioning method that partitions the unlabeled samples around the mean sample such that the



Iris$(|L| = 66)$ **Ionosphere**$(|L| = 72)$



Pima$(|L| = 126)$ **Vehicle**$(|L| = 102)$

Figure 3.3: The average overall classification accuracy ($\overline{OA}$) versus the batch size for fixed number of training samples of the different data sets for the existing techniques: MS-RP, MS-ABD, MS-cSV and MS.

informative samples are selected from the whole uncertainty regions of the feature space. As the diversity criterion incorporated is independent of the SVM decision hyperplane, it does not allow the biased shifting of SVM hyperplane when poor initial labeled samples are available. Hence the proposed technique is robust in nature. Moreover, the proposed partitioning method transforms the original n-dimensional feature space into one-dimensional space by dividing the unlabeled pool of samples in equal weighted partitions around the mean of unlabeled samples. As a result, it is highly time efficient. In the literature, there exist no specific criteria for batch-mode active learning techniques to decide about the number of samples to be selected at each iteration of algorithm. It can be observed that the proposed technique identifies the number of samples to be selected in successive iterations without the human intervention. At most one most informative sample is selected from each non-empty partition generated around the mean sample of unlabeled samples. Thus, the number of unlabeled samples selected at each iteration is determined automatically and the learner need not to decide and input the size of batch to train the classifier. The empirical study of the proposed technique with the four existing batch-mode active learning techniques shows that the proposed technique is very fast and always provides better or comparable classification accuracy. This technique will be very effective when the pool size of unlabeled samples is very large.

Chapter 4

A Context Sensitive Thresholding Technique for Automatic Image Segmentation [3]

4.1 Introduction

Image segmentation plays vital role in various applications like medical imaging, object detection, fingerprint identification, text recognition *etc.* [172, 179, 180]. It involves segmenting an image into regions with uniform characteristics. If different objects belong to an image are sufficiently separated from each other, then the histogram of the image may have many peaks to represent different objects. The potential thresholds can be found at the valley regions of the histogram by applying thresholding technique. A survey of various threshold selection techniques and their applications can be found in [149].

Thresholding techniques can be divided into bi-level and multilevel category, depending on the number of thresholds required to be detected. In bi-level thresholding, an image is segmented into two different regions depending on a threshold value selected from the histogram of the image [172, 179, 210]. The pixels with gray values greater than the threshold value are assigned into object region, and the rest are assigned into background. Multilevel thresholding segments a gray level image into several distinct regions by detecting more than one thresholds [211].

Histogram of the image does not consider spatial contextual information of the image. Thus, the existing thresholding techniques based on the histogram are unable to incorporate the spatial contextual information in the threshold selection process. Recently, energy function is defined to generate the energy curve of an image by taking into an account the spatial contextual information [175]. The characteristics of this curve is similar to the histogram of an image. In this work instead of histogram, the energy curve of the image is analyzed to find out the optimal number of thresholds. To select multiple potential thresholds from the energy curve, concavity analysis technique [176]

---

[3] This chapter has been published in *Computational Intelligence in Data Mining, Vol 2, Smart Innovation, Systems and Technologies, Springer*, Volume 32 pages 19-25, 2015.

is employed. In order to select optimal number of thresholds from the list of potential thresholds obtained by applying concavity analysis technique, a threshold elimination technique based on cluster validity measure is proposed. The proposed threshold elimination technique remove some potential thresholds from the list to find out the optimal number of thresholds for automatic segmentation of the image.

The rest of this chapter is organized as follows: Section 4.2 presents the proposed technique. Section 4.3 gives the detailed description of the data set and experimental results. Finally, Section 4.4 draws the conclusion of this work.

4.2 Proposed Method

In order to take into account the spatial contextual information of the image for thresholds selection, the energy curve published in [175] is employed. The detailed steps of the proposed technique are given in the following subsections.

4.2.1 Energy Curve

Recently, we defined an energy function to generate the energy curve of an image [175] which is based on the fact that the spatial relation among the neighborhood of each pixel of the image helps in effective recognition of objects present in the image. This energy function computes the energy of an image at each gray value by taking into an account the spatial contextual information of the image. The characteristics of this energy curve is similar to the histogram of an image i.e., if the energy curve of an image include peaks, we can separate it into a number of modes. Each mode is expected to correspond to a region, and there exists a threshold at the valley between any two adjacent modes. Since the energy curve is generated by taking into an account the spatial contextual information of the image, it is more smooth and may contain better discriminatory capability as compared to that of histogram. In details, Let $I = \{l_{ij}, 1 \leq i \leq m, 1 \leq j \leq n\}$ be an image of size $m \times n$ where l_{ij} is the gray value of the image I at pixel position (i, j) such that $0 \leq l_{ij} \leq L$. L is defined as the maximum gray value of the image I. For the given pixel position (i, j), the spatial correlation with the neighbouring pixels is defined with the help of neighborhood system configuration. This is done by selecting the square or rectangular part of the image centered with the pixel position (i, j). The rectangular or square area part is known as mask. The neighborhood system configuration or the size of mask depends on the value of order [138, 212]. If d is the order value, then for the spatial position (i, j) the neighborhood system N is defined as:

$$N_{ij}^{d} = (i + u, j + v), (u, v) \epsilon N^{d} \tag{4.1}$$

Let $d = 1$, this implies the first-order neighbor system which means $(u, v)\epsilon\{(0, \pm 1), (\pm 1, 0)\}$. To define energy function, second-order neighborhood configuration has been considered which implies $(u, v)\epsilon\{(0, \pm 1), (\pm 1, 0), (1, \pm 1), (-1, \pm 1)\}$. Figure 4.1 represents the first-order and second-order configurations. The energy function is based on second-order neighborhood configuration. Let E_l be the energy value of the image I at gray value l which is computed as:

(a) (b)

Figure 4.1: Neighborhood Configurations: (a) First Order (b) Second order

$$E_l = -\sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{pq \in N_{ij}^{2}} b_{ij}.b_{pq} + C \tag{4.2}$$

where $B_l = \{b_{ij}, 1 \leq i \leq m, 1 \leq j \leq n\}$ such that $b_{ij} = 1$ if $l_{ij} > l$; else $b_{ij} = -1$. N_{ij}^{2} represents the second-order neighbor pixels of the pixel at spatial position (i, j) and C is a constant ensures the energy value $E_l > 0$.

4.2.2 Concavity Analysis

After generating the energy curve of the image, concavity analysis technique presented in [176] is exploited to find out the list of potential thresholds from this energy curve. Let E be the energy curve of an image defined over the set of gray level $[0, L]$. For concavity analysis, we consider the subset of E say C defined over $[k_1, k_2]$ such that $C(k_1)$ and $C(k_2)$ are the first and last non-zero values of the energy curve E respectively. In order to find concavities of C, the smallest closed polygon P is constructed in the range $[k_1, k_2]$ by following the same steps as described in [176]. As energy curve E consist of peaks and valleys, only those peaks lying on the closed polygon P will be considered as boundaries where the potential threshold may exist. Then the vertical distance between P and C for each gray level between the selected consecutive peaks is calculated using an equation 4.3. The gray level with maximum vertical distance V between each selected consecutive peaks is considered as potential threshold.

$$V(i) = P(i) - C(i). \tag{4.3}$$

The number of thresholds obtained by this technique may be larger than the number of objects in the image. In order to select optimal number of thresholds for segmentation the proposed threshold elimination technique is presented next.

4.2.3 Proposed Threshold Elimination Technique

In order to find the optimal number of thresholds for automatic segmentation, a cluster validity measure called Davies Boulding (DB) index is used [213]. It is a function of the ratio of the sum of within-object scatter to between object separations. Let $\omega_1, \omega_2, ..., \omega_k$, be the k objects defined by thresholds $t_1 < t_2 < t_3 < .... < t_k$. Then the DB index is defined as:

$$\begin{aligned} R_{ij} &= \frac{\sigma_i^2 + \sigma_j^2}{d_{ij}^2} \\ R_i &= \max_{j=1...k; i \neq j}\{R_{ij}\} \\ DB &= \frac{1}{k}\sum_{1}^{k} R_i \end{aligned} \tag{4.4}$$

where σ_i^2 and σ_j^2 are the variances of object ω_i, and ω_j , respectively, and d_{ij}^2 is the distance of object centers ω_i and ω_j. Smaller the DB value, better is the segmentation as a low scatter and a high distance between object lead to small values of R_{ij}.

Let $T = \{t_1, t_2, ..., t_k\}$ be the set of k potential thresholds obtained by applying concavity analysis technique as described in Section 4.2.2. To find out the optimal number of objects present in the image, first, DB-index is calculated by taking into account all the k potential thresholds. In the next step, DB-index is calculated: **i)** by dropping the left most trailing potential threshold t_1 from T, **ii)** by dropping the right most trailing potential threshold t_k from T. If the DB-index obtained by i) is smaller than the DB-index obtained by ii), then drop t_1 else drop t_k from T. The process is repeated until T contain single potential threshold. As a result, different combinations of potential thresholds and their corresponding DB-index values are obtained. Finally the combination that is associated with smallest DB-index is selected to segment the image.

Figure 4.2: Original Image Dataset: (a) Man, (b) Cameraman, (c) Fingerprint, (d) Two Swans, (e) Peppers, (f) Lena, (g) House, and (h) Flinstones.

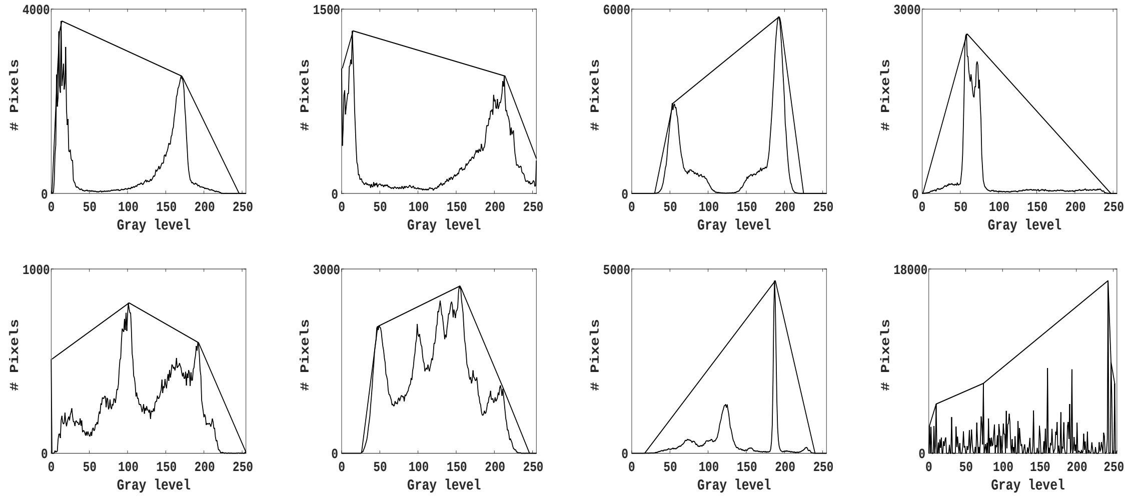


Figure 4.3: Smallest polygon that covers the histogram of the respective images: (a) Man, (b) Cameraman, (c) Fingerprint, (d) Two Swans, (e) Peppers, (f) Lena, (g) House, and (h) Flinstones.

4.3 Experimental Results

In order to assess the effectiveness of the proposed technique eight different images Man, Cameraman, Fingerprint, Two Swans, Peppers, Lena, House and Flinstones are considered. Figures 4.3 and 4.4 show the smallest polygon obtained on the histogram and the

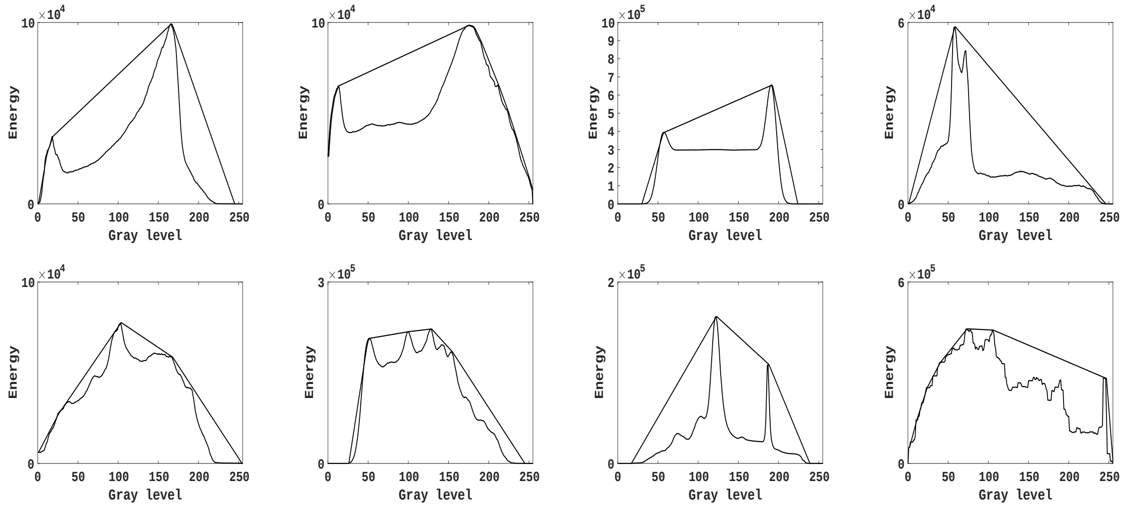


Figure 4.4: Smallest polygon that covers the energy curve of the respective images: (a) Man, (b) Cameraman, (c) Fingerprint, (d) Two Swans, (e) Peppers, (f) Lena, (g) House, and (h) Flinstones.

energy curve of different images respectively, by applying concavity analysis technique. From these figures one can observe that the energy curve of the image is more suitable as compare to the histogram to determine potential thresholds due to its smoothness property. Table 4.1 shows the different combinations of potential thresholds and their associated DB-index values obtained by using the proposed threshold elimination technique considering both the histogram and the energy curve of the images. From the Table 4.1 one can observe that for all the considered images, proposed technique produced better results when applied to energy curve of the images.

Image	Histogram		Energy curve	
	Thresholds	DB-index	Thresholds	DB-index
Man	5 10 13 39 172 183	0.4903	11 81 185	0.5662
	10 13 39 172 183	0.5455	11 81	0.5693
	5 10 13 39 172	0.4537	81 185	0.2046
	10 13 39 172	0.5126	**81**	**0.0291**
	5 10 13 39	0.2401	185	0.5219
	5 10 13	0.3767		
	10 13 39	0.2604		
	10 13	0.3865		
	13 39	0.2138		
	13	0.3816		
	39	**0.0397**		
	2 39 230	0.4410	5 112 183 203 217 229 241 253	0.4262

Image	Histogram		Energy curve	
	Thresholds	DB-index	Thresholds	DB-index
	39 230	0.3874	5 112 183 203 217 229 241	0.4497
	2 39	0.2433	112 183 203 217 229 241 253	0.1910
	2	0.3752	112 183 203 217 229 241	0.1842
	39	**0.0639**	183 203 217 229 241 253	0.2316
			112 183 203 217 229	0.2147
			183 203 217 229 241	0.2298
			183 203 217 229	0.2667
			112 183 203 217	0.2577
			112 183 203	0.2302
			183 203 217	0.3202
			112 183	0.1617
			183 203	0.3008
			112	**0.0444**
			183	0.2500
Fingerprint	42 141 194 195 207	0.5714	43 174 206	0.5259
	42 141 194 195	0.6415	43 174	0.6313
	141 194 195 207	0.5247	174 206	0.1694
	141 194 195	0.6006	**174**	**0.1480**
	194 195 207	0.6611	206	0.7923
	141 194	0.3870		
	194 195	0.7112		
	141	**0.0305**		
	194	0.6595		
Two Swans	51 59 85	0.2451	51 87	0.1818
	51 59	1.6360	51	0.8175
	59 85	0.3978	**87**	**0.1800**
	59	1.5467		
	85	**0.1922**		
Peppers	54 131 204	0.1728	81 121 219	0.2120
	54 131	**0.1266**	81 121	0.1957
	131 204	0.2114	121 219	0.1956
	54	0.1762	**121**	**0.1866**
	131	0.1830	219	0.2766
Lena	37 129 168 225	0.2470	39 69 111 136 185	0.1872
	37 129 168	0.2568	36 69 111 136	0.2456

Image	Histogram		Energy curve	
	Thresholds	DB-index	Thresholds	DB-index
	129 168 225	**0.2160**	**69 111 136 185**	**0.1675**
	129 168	0.2187	69 111 136	0.2355
	168 225	0.2333	111 136 185	0.1950
	129	0.2323	111 136	0.2894
	168	0.2524	136 185	0.2080
			111	0.1981
			136	0.2466
House	182 194	0.1801	111 144 194	0.2444
	182	**0.1318**	111 144	0.1757
	194	0.3128	144 194	0.2091
			144	**0.1231**
			111	0.2525
Flinstones	10 74 243 247 252	0.2500	9 15 31 70 94 205 249	0.3345
	10 74 243 247	0.2584	15 31 70 94 205 249	0.3571
	74 243 247 252	**0.2290**	**9 15 31 70 94 205**	**0.2036**
	74 243 247	0.2342	15 31 70 94 205	0.2081
	243 247 252	0.2489	9 15 31 70 94	0.2151
	74 243	0.2340	15 31 70 94	0.2229
	243 247	0.2542	31 70 94 205	0.2091
	74	0.2397	31 70 94	0.2279
	243	0.2492	70 94 205	0.2313
			70 94	0.2636
			31 70	0.2481
			70	0.2386
			31	0.2416

Table 4.1: Combinations of potential thresholds and corresponding DB-index generated by the proposed threshold elimination technique. The bold one is the optimal number of thresholds selected by the proposed technique.

The optimal results obtained by the proposed technique as shown in Table 4.1 are highlighted in bold. For the images Man, Cameraman, Fingerprint, Two swans images, proposed technique automatically determines single threshold to segment the images into two regions. For the pepper image, it determines two thresholds to segment the image into three homogenous regions by analyzing the histogram where as one threshold with the energy curve of the image. For the Lena image, the proposed technique automatically

Figure 4.5: Segmented images with respect to optimal thresholds obtained by proposed technique using histogram of the images: (a) Man, (b) Cameraman, (c) Fingerprint, (d) Two Swans, (e) Peppers, (f) Lena, (g) House, and (h) Flinstones.

Figure 4.6: Segmented images with respect to optimal thresholds obtained by proposed technique using energy curve of the images: (a) Man, (b) Cameraman, (c) Fingerprint, (d) Two Swans, (e) Peppers, (f) Lena, (g) House, and (h) Flinstones.

find out three and four thresholds by analyzing histogram and energy curve of the image respectively. For qualitative analysis, Figure 4.5 and 4.6 shows the segmented images obtained by the proposed technique considering the histogram and the energy curve of the images. For the images Fingerprint and Pepper, the db-index obtained using energy curve is slightly higher but the segmented images obtained using the energy curve are better than that obtained using the histogram of the image. From these figures one can visualize that the proposed technique based on energy curve produced much better results.

4.4 Discussion and Conclusion

In this work we developed a thresholding technique that works on the context sensitive energy curve of the images to determine the optimal number of thresholds to solve image segmentation problem. In our proposed technique, first concavity analysis technique exists in the literature is applied on the energy curve to detect all the potential thresholds. After detecting a list of potential threshold, a novel threshold elimination technique is used to remove some potential thresholds from the list to determine the optimal number of thresholds for automatic segmentation of the image. To assess the effectiveness of proposed technique, eight different images are considered in the experiment. From the experiment, it is observed that the proposed technique provides better result as compared to that applied on the histogram of the image. This is mainly because the histogram does not consider the spatial contextual information.

Chapter 5

A Fast Automatic Optimal Threshold Selection Technique for Image Segmentation [4]

5.1 Introduction

Image segmentation plays an important role in image analysis and computer vision. It is a process of partitioning an image into several non-overlapping, homogenous regions containing similar objects. In literature, there exist several techniques to solve image segmentation problems, namely histogram thresholding, edge-detection, clustering and so on [185]. If objects present in the image are distinguishable by their gray values, then the histogram of the image may have many peaks to represent different objects. The potential thresholds can be found at the valley regions of the histogram by applying thresholding technique. Automatic detection of these thresholds is one of the major challenge in the thresholding techniques. Due to the advantages of smaller storage space, fast processing speed and ease in manipulation, histogram thresholding have drawn a lot of attention in many applications like medical imaging, object detection, change detection, fingerprint identification, text recognition *etc.* [214, 180, 215]. A survey of various thresholding techniques and their applications can be found in [149].

Histogram thresholding techniques can be divided into bi-level [216, 210, 179, 173, 172, 217] and multilevel [218, 188, 219, 177, 220], depending on the number of thresholds required to be detected. Otsu's method [172] is one of the popular bi-level thresholding technique that selects the optimal threshold by maximizing the between class variance. The minimum error thresholding presented in [173] selects the optimal threshold by optimizing the criterion function based on the assumption that the object and background pixels are normally distributed. In [179] the optimal threshold is determined by maximizing the entropy of the object and background pixels. Bi-level histogram thresholding methods based on fuzzy sets theory are presented in [216, 210, 217]. In [220] a multilevel thresholding method for automatic image segmentation is presented. A fast multilevel thresholding based on low pass and high pass filter is proposed in [188]. Multilevel thresh-

---

[4] This chapter has been published in *Signal Image and Video Processing, Springer*, Vol 10, pages 19-25, 2015.

olding methods based on optimization techniques are found in [218, 219, 177].

The aforementioned thresholding techniques are based on one-dimensional (1D) histogram of the image. Histogram based thresholding techniques are easy to implement and usually have low computational burden. However, they do not take into account the spatial contextual information in thresholds selection process. To mitigate this limitation, two-dimensional histogram [221, 222], two-dimensional direction histogram [223], gray level spatial correlation histogram [224, 225, 226] and gray level gradient magnitude histogram [227] have been proposed. Although these techniques produced improved results but lost the basic advantages of 1D histogram based thresholding techniques. Moreover, all these traditional histogram based thresholding techniques are unable to determine the optimal number of thresholds for automatic segmentation of the input image.

In this chapter, a fast context sensitive multilevel threshold selection technique is presented to solve the image segmentation problems. To incorporate spatial contextual information in threshold selection process, the proposed technique employs recently defined 1D energy curve [175] of the image. The energy curve has the similar characteristics as 1D histogram of the image i.e., different objects in the image are represented by separate modes in the energy curve and there is a valley between two adjacent modes. The proposed technique follows three main steps to accomplish the segmentation task. First, it compute the gray values associated with different peaks of the energy curve and then detects all the possible thresholds passing through the middle of the two consecutive peaks of the energy curve. In the second step, a technique based on cluster validity measure is proposed to identify the optimal number of potential thresholds and their bounds (lower and upper limits) where optimal value of potential threshold may exist. In third step, the genetic algorithm (GA) is employed to compute the optimal value of each potential threshold within their defined bounds to segment the image. The proposed technique has several advantages: **i)** it is context sensitive, **ii)** computationally less demanding, **iii)** preserve the advantages of 1D histogram based thresholding techniques (like smaller storage space, fast processing speed and ease in manipulation etc.), and **iv)** able to determine the optimal number of segments in the image without expert knowledge. To assess the effectiveness of proposed technique the results obtained are compared with four state-of-the-art methods cited in literature. Experimental results on large number of images confirmed the effectiveness of the proposed technique.

The rest of this chapter is organized as follows: The concept of Genetic algorithm (GA) is presented in Section 5.2. The proposed technique is presented in Section 5.3. Section 5.4 provides the detailed description of the experimental settings and the results obtained on the considered images. Finally, Section 5.5 draws the conclusion of this work.

5.2 Genetic Algorithm

Genetic algorithm [228, 229] is a class of iterative procedure that simulate the evolution process of population of structures subject to the competitive forces prescribed in "survival of the fittest" principle. It is based on the concept that high fitness structures have relatively greater chance to survive and produce even higher fitness offspring. As a result, overall fitness of population will increase in successive generations. GA algorithm initialises itself with a random generation of population of individuals called chromosomes. Chromosome is a binary string of 0's and 1's and represents a point in the search space. The chromosomes are evaluated quantitatively for performance as possible solution in search space. Those having high fitness values are given more opportunities to produce better offsprings. In order to optimize, GA employs three operators to propagate it's population from one generation to another. Those operators are: Selection operator, Crossover operator and Mutation operators. The selection operator mimes "Survival of Fittest" principal , crossover operator mimes mating in biological populations where as mutation operator mimes characteristics of diversity in population so that algorithm may not get trapped in local optimum solution. The algorithm (5.1) presents the pseudo-code for GA.

Algorithm 5.1 Standard Genetic algorithm

```
Input:
    Maximum Number of iterations ($Max_{iter}$)
    Population size
    Crossover probability
    Mutation rate
Output:
    Best solution
Begin:
Generate random population of $n$ chromosomes
$itr \leftarrow 1$
while $itr \neq Max_{iter}$ do
    for all Chromosomes in the population do
        Calculate fitness value
    end for
    Select two parent chromosomes with better fitness value
    Crossover the parents to form new offspring.
    Mutate the new offspring
    Replace new offspring in the population
    $itr = itr + 1$
end while
```

5.3 Proposed Technique

The framework of the proposed technique is presented in Figure 5.1. It consists of three steps: 1) Initial thresholds selection, 2) Detection of optimum number of potential thresholds, and 3) Detection of optimal value of each potential threshold. In the first subsection a brief description of the energy curve and its role in initial thresholds selection is presented. The second subsection describes a novel technique based on the cluster validity measure to determine the optimal number of potential thresholds and their bounds. The final subsection presents a GA based technique to determine the optimal value of each potential threshold to segment the input image. Each subsection is elucidated next.

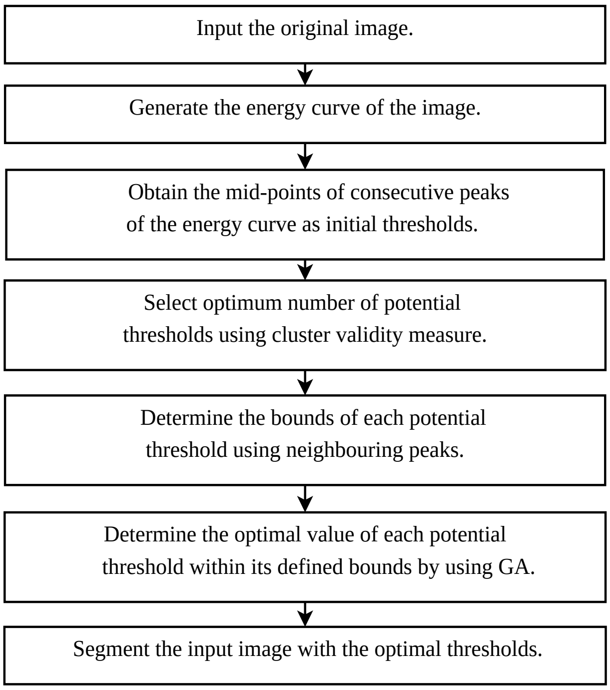


Figure 5.1: Framework of the proposed technique.

5.3.1 Initial thresholds selection

Histogram does not take into account the spatial contextual information of the image. To incorporate spatial contextual information in threshold selection process, the energy curve of the image defined in [175] is used here.

Let $I = \{l_{ij}, 1 \leq i \leq m, 1 \leq j \leq n\}$ be an image of size $m \times n$ where l_{ij} is the gray value of the image I at pixel position (i, j). The energy value E_l of the image I at gray value

l is computed as:

$$E_l = -\sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{pq \in N_{ij}^2} b_{ij}.b_{pq} + C \tag{5.1}$$

where $B_l = \{b_{ij}, 1 \leq i \leq m, 1 \leq j \leq n\}$ such that $b_{ij} = 1$ if $l_{ij} > l$; else $b_{ij} = -1$. N_{ij}^2 represents the second-order neighbor pixels of the pixel at spatial position (i, j) and C is a constant that ensures the energy value $E_l > 0$. For more details, the reader may refer [175].

Like histogram of the image, the energy curve has very much similar characteristics. The energy curve of the image also include peaks, which can be separated into number of modes as that in histogram. Each mode is expected to correspond to a region, and there exists a valley between any two adjacent modes. Since the energy curve is generated by taking into an account the spatial contextual information of the image, it is smoother and has better discriminatory capability as compared to the histogram of the image. Figure 5.2 shows the original House image, its histogram and energy curve. From these figures, one can see that the energy curve of the image is smoother than the histogram of the image hence the number of peaks (as well as valleys) exist in the energy curve is less than that in the histogram of the image. Thus, energy curve may be a better choice for suitable threshold selection than the histogram not only because of the inclusion of spatial contextual information but also for the quick selection of optimal number of potential thresholds.

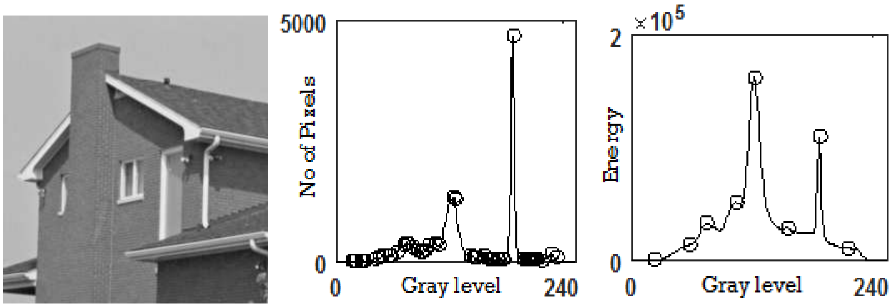


Figure 5.2: (a) Original House image, (b) Histogram and (c) Energy curve. The circles represent the peaks(local maxima) of the curves.

The energy curve based segmentation technique presented in [175] is unable to detect optimal number of segments of the input image. The technique presented in [187] solves this problem by employing concavity analysis technique [176]. Since concavity analysis technique skip some valleys of the energy curve, it may fail to detect the optimal number of potential thresholds. To mitigate these limitations, a novel initial threshold selection technique is proposed as follows:

Let E be the energy curve of an image defined over the set of gray level $[0, L]$. Consider the subset of E say S defined over $[k_1,k_2]$ such that $E(k_1)$ and $E(k_2)$ are the first and last non-zero values of the energy curve E respectively. Let $P = \{p_0, p_1, ..., p_{n-1}\}$ be the set of n peaks exist in S. Then the curve S can be partition into $n + 1$ regions namely $\{(k_1, p_0), (p_0, p_1), ..., (p_{n-1}, k_2)\}$, where each region may contain a threshold to distinguish different objects present in the image. In the proposed technique, the possible $n+1$ initial thresholds are defined by taking the mid-point gray value of each partition. The number of initial thresholds obtained by this method may be larger than the number of objects present in the image. The method to select the optimum number of potential thresholds and their respective bounds where the optimal value of each potential threshold may exist is presented in the next subsection.

5.3.2 Detection of optimum number of potential thresholds

In this subsection, a novel technique based on cluster validity measure is proposed to detect the optimal number of potential thresholds. The optimal number of potential thresholds are selected from the list of initial thresholds obtained from the energy curve. It is agglomerative in nature. The proposed technique exhaustively merge two adjacent modes to select optimal number of potential thresholds with the help of a validity measure called Davies Boulding (DB) index [230]. It is a function of the ratio of the sum of within-object scatter to between object separations. Let $\omega_1, \omega_2, \omega_3, ..., \omega_k$ be the k objects of a segmented image separated from each other by defining thresholds $t_1, t_2, t_3, ..., t_{k-1}$, where $t_1 < t_2 < t_3 < ... < t_{k-1}$. Thus, the pixels of the image whose gray values are in the range $[t_{i-1}t_i]$ construct the object ω_i of the segmented image. The DB index of the segmented image is computed as:

$$\begin{aligned} DB &= \frac{1}{k}\sum_{i=1}^{k} R_i \\ R_i &= \max_{j=1,\dots,k,i\neq j}\{R_{ij}\} \\ R_{ij} &= \frac{\sigma_i^2 + \sigma_j^2}{d_{ij}^2}. \end{aligned} \tag{5.2}$$

where σ_i^2 and σ_j^2 are the variances of object ω_i, and ω_j, respectively, and d_{ij}^2 is the distance between centers of object ω_i and ω_j. The gray values of the pixels belong to an object are used to compute variance and center of that object. Smaller the DB value, better is the segmentation as a low scatter and a high distance between object gives small values of R_{ij}.

Let $T_k = \{t_1, t_2, ..., t_k\}$ be the set of k initial thresholds obtained from the energy curve. To determine the optimal number of objects present in the image following steps are taken into an account: First, DB index is computed by taking into account all the k thresholds. Then at a time by dropping single threshold from T_k, $(k-1)$ subsets each containing $(k-1)$ thresholds are generated along with their respective DB index values. From these $(k-1)$ subsets, the subset denoted as T_{k-1} that produced minimum DB-index is selected for further analysis. The same procedure is repeated to obtain another subset of T_{k-1} denoted as T_{k-2} that contain $(k-2)$ thresholds with lowest DB-index value. The process is repeated until the subset T_1 consisting of single initial threshold is obtained. After the computation of $T_k, T_{k-1}, T_{k-2}, ..., T_2$, and T_1 along with their corresponding DB-index values, the subset $T_i (i = 1, 2, ..., k)$ associated with smallest DB-index is chosen to select the optimal number of potential thresholds. For the House image, different subsets of potential thresholds generated by the proposed technique are shown in Table 5.1. Since the subset T_4 correspond to the lowest DB index value, the optimal number of potential thresholds is determined as 4 with the initial thresholds $t_1 = 20, t_2 = 90, t_3 = 172$ and $t_4 = 227$.

Table 5.1: Different subsets generated by the proposed technique for the house image.The optimal number of potential thresholds selected by the proposed technique are represented in bold.

Subsets	**Thresholds**	**DB index**
T_9	20,40,66,90,114,140,172,202,227	0.1892
T_8	20,40,66,90,114,140,172,227	0.1391
T_7	20,40,90,114,140,172,227	0.1245
T_6	20,40,90,140,172,227	0.1111
T_5	20,40,90,172,227	0.0974
T_4	**20,90,172,227**	**0.0928**
T_3	20,172,227	0.0961
T_2	172,227	0.0971
T_1	172	0.1248

Since the initial thresholds are assumed as the middle of every two adjacent peaks of the energy curve, the value of each potential threshold selected may not be optimal. The optimal value of each potential threshold may lie any where within the range of two adjacent peaks to which it belongs. Thus, to define the bounds of each potential threshold where its optimal value may exist,the nearest left and right peaks associated to a threshold are considered as its lower and upper bound, respectively. Figure 5.3 shows the 4 potential thresholds of the House image determined using the aforementioned technique and their respective regions where the optimal thresholds can be obtained.

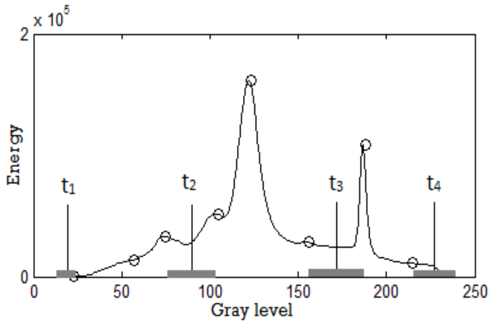


Figure 5.3: Potential thresholds t_1, t_2, t_3, and t_4 of the House image. The Gray region associated with each potential threshold represents the range in which the optimal value can be obtained.

5.3.3 Detection of optimal potential thresholds

To find the optimal (or near optimal) value of each potential threshold within their defined range GA [231] is employed. Let k be the optimal number of potential thresholds obtained from the energy curve of the image. In the proposed technique each chromosome is a sequence of binary numbers representing these k thresholds. If each potential threshold is represented by s bits then the length of the chromosome will be $k \times s$ bits. Initial values of the potential thresholds in a chromosome are taken randomly within their defined ranges. To compute fitness value of each chromosome in the population, DB index criterion presented in Equation 5.2 is used. The fitness value computation, selection, crossover and mutation are executed for a certain number of iterations to find optimal value (or near optimal value) of each potential threshold within its defined range. The GA is terminated when the average fitness value of the population becomes stable. Finally, the chromosome in the population that has maximum fitness value (minimum value of DB-index) represent the optimal thresholds. These optimal thresholds discriminate the different homogenous regions of the image.

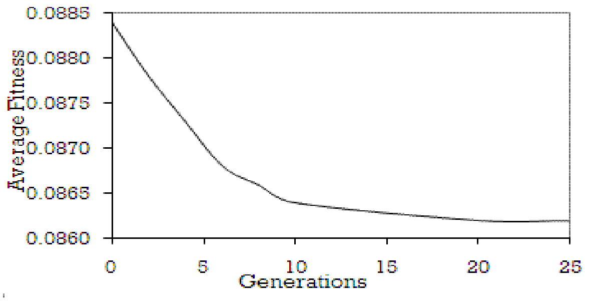


Figure 5.4: Convergence graph of the House image.

The computational complexity of this step is significantly reduce by shrinking the search space of GA. Since the lower and upper bound of each potential threshold are given as input to the GA, the search space becomes narrower. Thus, the termination condition is satisfied in less number of iterations. Figure 5.4 shows the convergence graph of the house image.

5.4 Experimental Results

To evaluate the effectiveness of the proposed technique eight different images are considered for the experiments. Figure 5.5, 5.6, and 5.7 shows the original images, their histograms and energy curves, respectively. From these figures one can observe that the energy curve of an image has similar characteristics as that of histogram of the image i.e., it also has peaks and valleys to discriminate different objects as that of histogram. Since the energy curve is generated by taking into an account the spatial contextual information of the image, it is smoother and has better discriminatory capability as compared to that of histogram of the image. Thus, without loosing the benefits of the histogram for suitable threshold selection, energy curve may be a better choice.

Figure 5.5: Original Image Dataset: (a) Man, (b) Cameraman, (c) Fingerprint, (d) Two Swans, (e) Peppers, (f) Lena, (g) House, and (h) Flinstones.

To assess the effectiveness of the proposed technique, results are compared with four state-of-the-art techniques exist in literature. Since the technique is context-sensitive it has been compared with two GA based context sensitive techniques: an energy curve based (referred as ECCS) [175] and a pattern based (referred as PCS) [178]. The proposed technique is also compared with two context insensitive techniques: a fractional-order Darwinian Particle Swarm Optimization based (referred as FODPSO) [177] and

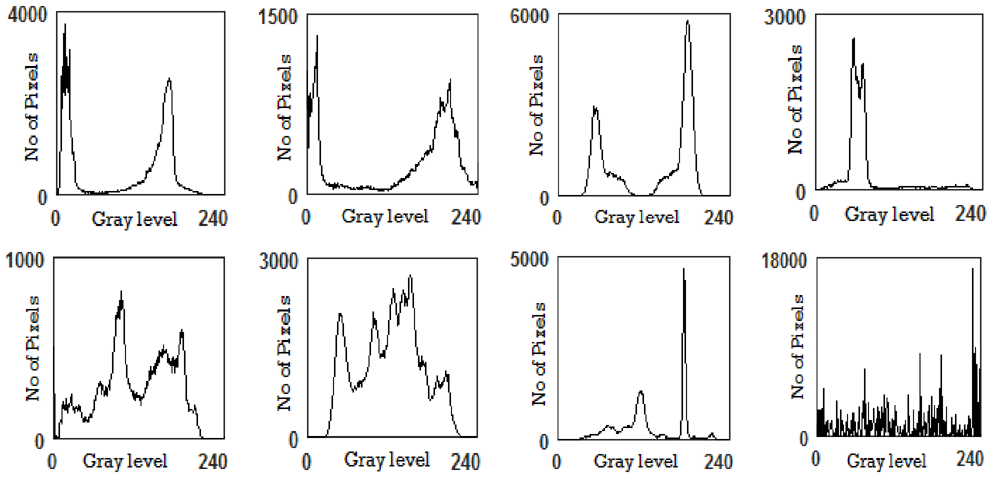


Figure 5.6: Histograms of the experimental image data set: (a) Man, (b) Cameraman, (c) Fingerprint, (d) Two Swans, (e) Peppers, (f) Lena, (g) House, and (h) Flinstones.

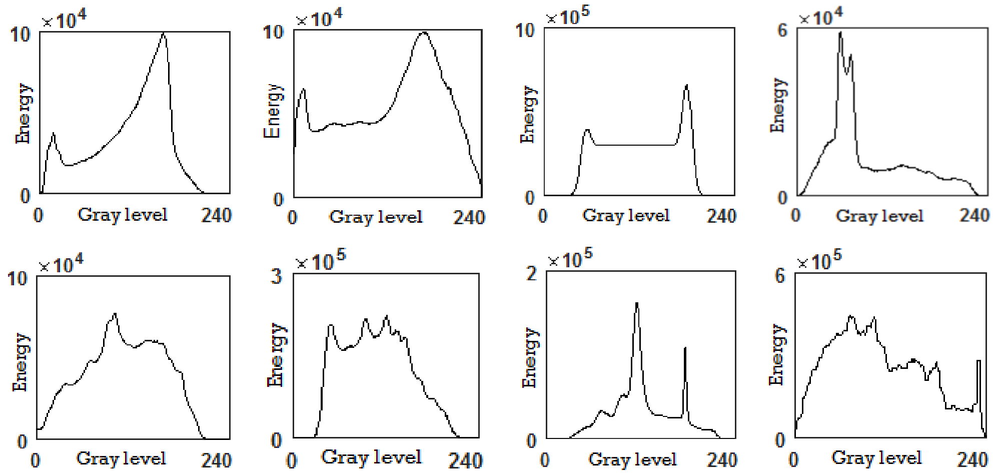


Figure 5.7: Energycurves of the experimental image data set: (a) Man, (b) Cameraman, (c) Fingerprint, (d) Two Swans, (e) Peppers, (f) Lena, (g) House, and (h) Flinstones.

an entropy based histogram thresholding (referred as Kapur method) [179]. The ECCS technique employs GA to detect optimal values for the predefined number of thresholds by maximizing entropy of the energy curve. In PCS, first the input patterns are generated corresponding to each pixel of the image by considering its neighboring pixels information. In experiment each input vector corresponds to a pixel contain two components, the gray value of the pixel and the average gray value of its four neighboring pixels. After generating the input patterns, the PCS technique exploits GA to determine the predefined number of cluster representatives to segment the input image. The FODPSO is based on fractional calculus strategy which controls the convergence of particles thus provides higher level of exploration and ensures global solution. The FODPSO selects the predefined number of optimal thresholds by maximizing the between class variance. The Kapur method [179] is a conventional histogram thresholding technique based on entropy maximization.

To assess the effectiveness of the proposed technique a cluster validity measure S_Dbw index which is not involved in the implementation of the proposed and existing techniques is taken into account [232]. It is based on the cluster compactness in terms of intra-cluster variance and inter-cluster density. The S_Dbw index with C number of clusters denoted as S_Dbw(C) is defined as:

$$S\_Dbw(C) = Scat(C) + Den(C) \tag{5.3}$$

where $Scat(C)$ and $Den(C)$ represent the intra-cluster variance and the inter-cluster density, respectively. The number of clusters that minimizes the S_Dbw index can be considered as an optimal number of the objects present in the image.

The algorithms have been implemented in MATLAB (R2012a). In order to fix different control parameters value of GA, several experiments were performed by varying their values within a wide range. From the experiments, it is observed that the proposed technique produced similar results when the population size, crossover probability and mutation probability of GA is varied in the range [20 80], [0.5 0.9] and [0.05 0.001], respectively. In the present experiments, for all considered images the population size, crossover and mutation probability is set as 20, 0.8 and 0.01 respectively.

Stochastic selection strategy is used to select fittest chromosomes from the mating pool. For FODPSO based thresholding technique, both the individual and social weight of the particles is set as 0.8. The fractional coefficient is set as 0.6.

5.4.1 Results Analysis

The first experiment is devoted to analyze the validity of the thresholds obtained by the proposed technique. Table 5.2 shows the initial thresholds, optimal number of thresholds and their optimal values obtained by the proposed technique for different images. By analyzing the histograms, energy curves shown in the Figures 5.6 and 5.7 and the results reported in Table 5.2, one can visualize that for all the considered images the optimal thresholds obtained by the proposed technique always pass through the valley region of the energy curves as well as histograms of the images. As an example, for the Peppers image the optimal thresholds 51 and 122 obtained by the proposed technique passes through the valley region in both histogram and energy curve as shown in the Figures 5.6(e) and 5.7(e). This confirms the validity of the proposed technique.

Table 5.2: Initial thresholds, potential thresholds and optimal thresholds detected by the proposed technique for different images.

Images	**Initial selected thresholds**	**Potential thresholds**	**Optimal thresholds**
Man	10,22,34,45,107,206	107	81
Cameraman	8,22,31,35,47,61,67 71,74,82,89,92,97,102 104,141,178,180,197,234	89	90
Fingerprint	44,67,76,79,82,85 90,95,97,106,117,139 161,165,179,208	139	120
Two Swans	22,45,53,66,82,94 99,104,108,112,116,120 125,133,140,142,146,149 153,157,160,169,186,198 202,206,212,217,223,237	22,125	1,127
Peppers	21,56,85,101,113,126 132,140,149,154,157,162 172,183,189,223	56,113	51,122
Lena	39,66,81,92,108,122 136,143,150,164,182,218	81,108 182	80,115,179
House	20,40,66,90,114,140 172,202,227	20,90 172,227	17,88,158,219
Flinstones	25,51,56,66,76,82,90 96,103,121,137,148,159 164,169,176,182,188,201 215,220,224,230,239,250	51,148 230,239 250	48,147,230,241,249

Table 5.3: Quantitative results obtained by the proposed, the ECCS, the FODPSO, the PCS and the Kapur methods.

Images	**Proposed**			**ECCS[175]**			**FODPSO[177]**			**PCS[178]**			**Kapur[179]**		
	Thresholds	**DB index**	**SDbw index**	**Thresholds**	**DB index**	**SDbw index**	**Thresholds**	**DB index**	**SDbw index**	**Cluster representative**	**DB index**	**SDbw index**	**Thresholds**	**DB index**	**SDbw index**
Man	81	0.029	0.077	101	0.032	0.089	90	0.030	0.080	15.23 34.01 155.91 141.87	0.068	0.129	181	0.549	4.036
Cameraman	90	0.042	0.120	125	0.048	0.133	110	0.044	0.122	22.04 25.69 196.79 194.30	0.048	0.118	166	0.145	0.314
Fingerprint	120	0.029	0.060	124	0.029	0.060	126	0.029	0.060	63.99 93.31 186.76 168.18	0.617	0.116	153	0.050	0.105
Two Swans	1 127	0.075	0.299	83 162	0.186	0.433	103 176	0.133	0.325	52.52 55.26 74.70 78.21 201.81 149.20	0.345	0.460	83 163	0.187	0.430
Peppers	51 122	0.129	0.251	71 142	0.161	0.297	71 138	0.159	0.283	75.10 76.10 145.30 144.71 188.52 185.66	0.257	0.540	79 149	0.189	0.373
Lena	80 115 179	0.149	0.229	82 131 179	0.158	0.232	81 127 171	0.160	0.241	48.61 49.30 107.03 106.87 155.58 154.48 201.55 200.64	0.156	0.243	81 125 179	0.156	0.231
House	17 88 158 219	0.086	0.068	64 102 148 211	0.138	0.106	81 111 156 205	0.163	0.105	67.11 67.71 112.88 113.73 133.75 131.62 186.69 185.69 224.44 219.88	0.257	0.222	58 87 114 193	0.271	1.089
Flinstones	48 144 231 241 250	0.129	0.177	56 107 144 196 240	0.229	0.168	43 88 133 177 220	0.150	0.164	16.63 24.51 47.36 60.75 78.46 78.51 113.44 108.40 167.18 164.17 228.98 226.68	0.262	0.183	0 54 93 134 188	0.144	0.159

The second experiment compares the performance of the proposed technique with the ECCS, the FODPSO, the PCS and Kapur methods by using different images. Table 5.3 reports the optimal threshold and corresponding DB index and S_Dbw index obtained by the proposed, the ECCS, the FODPSO and the Kapur methods considering different images. It also reports the cluster representatives, associated DB-index and S_Dbw index obtained by the PCS techniques. From this table one can observe that the proposed technique always produced better DB-index as compared to the ECCS, the FODPSO, the PCS and the Kapur methods. These results are expected as proposed technique minimizes the DB index to obtain optimal thresholds. For fair comparisons, another cluster validity measure called S_Dbw index that has no involvement for the implementation of these techniques is computed. From Table 5.3 one can observe for the images Cameraman, Fingerprint and Flinstones the proposed technique produced similar S_Dbw index as obtained by the most effective technique. For the other images the proposed technique produced much better S_Dbw index. For qualitative analysis, Figures 5.8, 5.9, 5.10, 5.11 and 5.12 show the segmented results of the proposed technique as well as the existing ECCS, FODPSO, PCS and Kapur's technique repectively.

Figure 5.8: Segmented images of the experimental image data set using proposed technique: (a) Man, (b) Cameraman, (c) Fingerprint, (d) Two Swans, (e) Peppers, (f) Lena, (g) House, and (h) Flinstones.

The third experiment validates the consistency of the proposed technique. Figure 5.13 shows box-plot of all the images for 20 runs. Form this figure one can see that the widths of the boxes are thin. This indicate the standard deviation obtained for each considered image is very small which ensures the consistency of the proposed technique.

The fourth experiment deals with the computational time required by the different techniques for the experimental images. All the experiments were carried out on a PC [IN-

Figure 5.9: Segmented images of the experimental image data set using ECCS technique: (a) Man, (b) Cameraman, (c) Fingerprint, (d) Two Swans, (e) Peppers, (f) Lena, (g) House, and (h) Flinstones.

Figure 5.10: Segmented images of the experimental image data set using FODPSO technique: (a) Man, (b) Cameraman, (c) Fingerprint, (d) Two Swans, (e) Peppers, (f) Lena, (g) House, and (h) Flinstones.

Table 5.4: Computational time (in seconds) taken by the different techniques for all the input images.

Images	**Segments**	**Proposed**	**ECCS**	**FODPSO**	**PCS**	**Kapur**
Man	2	3.59	5.33	4.36	20.14	0.6250
Cameraman	2	1.87	4.25	5.02	9.66	0.4375
Fingerprint	2	5.88	7.42	6.48	24.15	0.3750
Two Swans	3	2.36	4.19	7.27	13.78	1.1250
Peppers	3	1.78	4.07	5.33	22.05	0.9844
Lena	4	8.02	10.12	9.92	50.81	45.5938
House	5	1.81	4.01	12.39	16.86	2.2652e+03
Flinstones	6	8.55	10.75	12.61	64.02	11.4534e+05

Figure 5.11: Segmented images of the experimental image data set using PCS technique: (a) Man, (b) Cameraman, (c) Fingerprint, (d) Two Swans, (e) Peppers, (f) Lena, (g) House, and (h) Flinstones.

Figure 5.12: Segmented images of the experimental image data set using Kapur's technique: (a) Man, (b) Cameraman, (c) Fingerprint, (d) Two Swans, (e) Peppers, (f) Lena, (g) House, and (h) Flinstones.

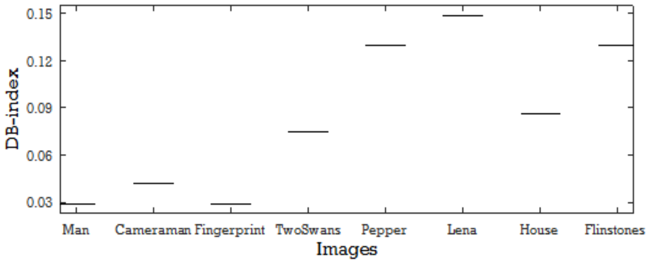


Figure 5.13: Box-plot of 20 runs for all the input images.

TEL(R) Core(TM)2 Duo 2.0 GHz with 2.0 GB of RAM]. As explained earlier, the proposed technique narrow down (shrinks) the search space of GA by defining lower and upper bound of each detected potential threshold. Thus, the termination criteria is satisfied in less number of iterations. This resulted faster convergence of GA as compared to the conventional optimization techniques. Table 5.4 shows the computational time taken by the different techniques for all eight images. From this table, it can be observed that the computational time required by the proposed technique is significantly lower than the ECCS, the FODPSO and the PCS techniques. The Kapur method is computationally efficient for detecting small number of thresholds. When number of thresholds increase its computational complexity also increase exponentially. Thus, for the House and the Flinstones images its computational time is very high. Moreover, the existing techniques are unable to determine optimal number of segments present in the image. On the other hand, the proposed technique automatically determines the optimal number of segments present in the image.

5.5 Conclusion

In this chapter, a context sensitive fast threshold selection technique is proposed for solving image segmentation problems. To incorporate spatial contextual information in threshold selection process, the technique analyzed energy curve of the image recently proposed in the literature [175]. The proposed technique first select all possible thresholds by detecting valley regions of the energy curve. Then based on a cluster validity measure it selects optimal number of the potential thresholds from the initial selected thresholds. Finally, GA is employed to detect the optimal value of each potential threshold. The proposed technique has several advantages: **i)** it is context sensitive, **ii)** it is computationally less demanding, **iii)** it preserve the advantages of 1D histogram based thresholding techniques, and **iv)** it is able to determine optimal number of segments present in the image.

To assess the effectiveness of proposed technique the results obtained by it are compared with the four state-of-the-art methods. Experimental results on large number of images confirmed the effectiveness of the proposed technique.

Although the proposed technique produced better results, still there is a scope to improve the results by using more sophisticated cluster validity criterion. As future developments of this work, we plan to explore such different criteria to improve the results. Variants of evolutionary approaches can also be explored to achieve the improved results. Moreover, the researchers may focus to develop a model where GA will detect the optimal number of thresholds in addition to threshold values.

Chapter 6

Conclusions and Scope for Further Research

6.1 Conclusions

In all the chapters (except Chapter 1) of this dissertation, conclusions drawn from the respective methodologies developed and the experimental results have been presented in respective chapters. Here we consolidate them to provide an overall summary on the contribution of the thesis.

In chapter 2, we have proposed a novel semi-supervised learning technique for pattern classification based on PTSVM which overcomes the limitations of the PTSVM based techniques cited in literature. The existing PTSVM based techniques select the transductive samples by employing only the properties of SVM classifier and do not consider the low-density region of the feature space. As a result, the classification performance may be degraded when the poor initial training set is available. In order to improve the classifier accuracy and to deal with such situation of low-density region, the proposed technique has exploited the properties of the SVM classifier, a k-nn technique and the cluster assumption property of patterns to select accurate transductive samples.

In chapter 3, we have proposed a fast and efficient batch mode active learning technique by designing a novel partitioning method combined with the SVM to incorporate both the diversity and uncertainty criteria so that most informative samples are selected in successive iterations. The proposed technique first partitions the unlabeled pool U so that the informative samples are selected from all the uncertainty regions of the feature space. Thus the proposed technique is robust in nature as it does not depend on the initial SVM (which is poor) and the partitioning does not allow the biased shifting of SVM hyperplane. The proposed technique is highly time efficient as the partitions are obtained by transforming the original feature space into 1-D feature space. In batch-mode active learning techniques cited in literature, the size of the batch i.e. the number of samples selected in each iteration is fixed and given by expert. We have designed an active learning technique in which the number of most informative samples selected at each iteration is not fixed and determined automatically depending on the number of non-empty partitions generated around the mean of unlabeled samples. This do not require the expert knowledge that what should be the size of batch.

Recently we have designed a novel energy function which includes the spatial contextual information of the gray image and helps to design 1-D energy curve of the image which has similar characteristics as histogram of the image. It has been concluded that the energy curve is smoother than the histogram of the image and hence has better discriminatory capabilities. Based on the energy curve, we have developed a context sensitive thresholding technique in chapter 4 which determines the optimal number of thresholds using concavity analysis. This work illustrates the effectiveness of energy curve when used to determine the thresholds and the number of objects present in the image. Experimental results have proved that the energy curve based thresholding technique provides better results as compared to that applied on histogram of the image.

The thresholding technique developed in chapter 4, may skip few potential threshold/s in some cases depending on the nature of energy curve of the image hence the thresholds obtained may not be the optimal. To mitigate such problem, in chapter 5, a context sensitive fast threshold selection technique is proposed. In this work also, the energy curve has been employed to incorporate spatial contextual information in threshold selection process. The proposed technique has several advantages: **i)** it is context sensitive, **ii)** it is computationally less demanding, **iii)** it preserve the advantages of 1D histogram based thresholding techniques, and **iv)** it is able to determine optimal number of segments present in the image. The effectiveness of proposed technique has been assessed by comparing the results obtained with the four state-of-the-art techniques.

6.2 Scope for Further Research

Although the presented advanced techniques for pattern classification provide number of significant findings, facing vast amount of combination or hybridization of models, designs and applications, it remains a tip of iceberg. As future developments of this work, one can identify the applications or domains where these techniques can be applied to achieve better results. Although the semi-supervised and active learning techniques in Chapters 2 and 3 have been designed in such way that they can be applied to any data set which meets the said conditions. Still, depending on the domains, constraints, parameters, and other environmental conditions, the presented algorithms may require certain customization. Both the semi-supervised and active learning techniques presented in Chapters 2 and 3 respectively, can also be applicable for the analysis of different images to accomplish certain classification tasks. For the analysis of images, generation of input patterns considering spatial information of the image may provide better classification results. As future developments, we will try to exploit mathematical morphology, wavelet transform or Markov random field etc. to generate spatial information based patterns that can be used as input to our developed semi-supervised and active learning techniques.

In Chapters 4 and 5, the thresholding techniques presented for image segmentation produced better results, still there is a scope to improve the results by using more sophisticated cluster validity criterion. As future developments of this work, different cluster validity criteria can be explored to improve the thresholding results. Also these segmentation techniques are generalized ones and can be applied in different image based pattern recognition domains as preprocessing tool such as document binarization, tumors detection, land covers systems etc. Depending on the domain and types of images these segmentation techniques also may require certain customization.

The technique presented in Chapter 5 employs GA to determine optimal thresholds. Variants of evolutionary approaches may be explored to achieve the improved thresholding results. The researchers may also focus to develop a model where the optimal number of thresholds will also be determined by the evolutionary algorithm like GA, PSO etc. in addition to threshold values of the input image.

Bibliography

[1] S. Theodoridis and K. Koutroumbas. *Pattern Recognition.* Amsterdam: Elsevier Academic Press, 2003.

[2] JP Marques De Sa. *Pattern Recognition: Concepts, Methods and Applications.* Springer Science & Business Media, 2012.

[3] David J Galas, Mark Eggert, and Michael S Waterman. Rigorous pattern-recognition methods for dna sequences: Analysis of promoter sequences from escherichia coli. *Journal of Molecular Biology*, 186(1):117–128, 1985.

[4] Dariu M Gavrila. The visual analysis of human movement: A survey. *Computer Vision and Image Understanding*, 73(1):82–98, 1999.

[5] Larry M Manevitz and Malik Yousef. One-class svms for document classification. *Journal of Machine Learning Research*, 2:139–154, 2001.

[6] Michelangelo Diligenti, Paolo Frasconi, and Marco Gori. Hidden tree markov models for document image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(4):519–523, 2003.

[7] Moataz El Ayadi, Mohamed S Kamel, and Fakhri Karray. Survey on speech emotion recognition: Features, classification schemes, and databases. *Pattern Recognition*, 44(3):572–587, 2011.

[8] Subhadip Basu, Chitrita Chaudhuri, Mahantapas Kundu, Mita Nasipuri, and Dipak Kumar Basu. Text line extraction from multi-skewed handwritten documents. *Pattern Recognition*, 40(6):1825–1839, 2007.

[9] Subhadip Basu, Nibaran Das, Ram Sarkar, Mahantapas Kundu, Mita Nasipuri, and Dipak Kumar Basu. A hierarchical approach to recognition of handwritten bangla characters. *Pattern Recognition*, 42(7):1467–1484, 2009.

[10] Anil Kumar Sao and B Yegnanarayana. Face verification using template matching. *IEEE Transactions on Information Forensics and Security*, 2(3):636–641, 2007.

[11] Anil Kumar Sao, B Yegnanarayana, and BVK Vijaya Kumar. Significance of image representation for face verification. *Signal, Image and Video Processing*, 1(3):225–237, 2007.

[12] RM Haralick and K Shanmugam. Computer classification of reservoir sandstones. *IEEE Transactions on Geoscience Electronics*, 11(4):171–177, 1973.

[13] Saroj K Meher and Sankar K Pal. Rough-wavelet granular space and classification of multispectral remote sensing image. *Applied Soft Computing*, 11(8):5662–5673, 2011.

[14] B Uma Shankar, Saroj K Meher, and Ashish Ghosh. Wavelet-fuzzy hybridization: Feature-extraction and land-cover classification of remote sensing images. *Applied Soft Computing*, 11(3):2999–3011, 2011.

[15] Carlo Marin, Francesca Bovolo, and Lorenzo Bruzzone. Building change detection in multitemporal very high resolution sar images. *IEEE Transactions on Geoscience and Remote Sensing*, 53(5):2664–2682, 2015.

[16] Sicong Liu, Lorenzo Bruzzone, Francesca Bovolo, and Peijun Du. Hierarchical unsupervised change detection in multitemporal hyperspectral images. *IEEE Transactions on Geoscience and Remote Sensing*, 53(1):244–260, 2015.

[17] Sicong Liu, Lorenzo Bruzzone, Francesca Bovolo, and Peijun Du. Unsupervised multitemporal spectral unmixing for detecting multiple changes in hyperspectral images. *IEEE Transactions on Geoscience and Remote Sensing*, 54(5):2733–2748, 2016.

[18] Andréa P Fernandes, Mirian C Santos, Sherlan G Lemos, Márcia MC Ferreira, Ana Rita A Nogueira, and Joaquim A Nóbrega. Pattern recognition applied to mineral characterization of brazilian coffees and sugar-cane spirits. *Spectrochimica Acta Part B: Atomic Spectroscopy*, 60(5):717–724, 2005.

[19] AJ Perez, F Lopez, JV Benlloch, and Svend Christensen. Colour and shape analysis techniques for weed detection in cereal fields. *Computers and Electronics in Agriculture*, 25(3):197–212, 2000.

[20] Anil Jain, Ruud Bolle, and Sharath Pankanti. *Biometrics: Personal identification in Networked society*, volume 479. Springer Science & Business Media, 2006.

[21] Keinosuke Fukunaga. *Introduction to Statistical Pattern Recognition.* Academic press, 2013.

[22] R Bajesy, R Lieberson, and M Reivich. A computerized system for the elastic matching of deformed radiographic images to idealized atlas imaging. *Journal of Computer Assisted Tomography*, 7:618–625, 1983.

[23] Tim McInerney and Demetri Terzopoulos. Deformable models in medical image analysis: a survey. *Medical Image Analysis*, 1(2):91–108, 1996.

[24] Anil K. Jain, Yu Zhong, and Sridhar Lakshmanan. Object matching using deformable templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(3):267–278, 1996.

[25] Asker M Bazen and Sabih H Gerez. Fingerprint matching by thin-plate spline modelling of elastic deformations. *Pattern Recognition*, 36(8):1859–1867, 2003.

[26] Ankit Kumar, Ashish Joshi, Anil Kumar, Ankush Mittal, and D Gangodkar. Template matching application in geo-referencing of remote sensing temporal image. *International Journal of Signal Processing, Image Processing and Pattern Recogni-*

tion, 7(2):201–210, 2014.

[27] John Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (6):679–698, 1986.

[28] Barbara Zitova and Jan Flusser. Image registration methods: a survey. *Image and Vision Computing*, 21(11):977–1000, 2003.

[29] Asker M Bazen, Gerben TB Verwaaijen, Sabih H Gerez, Leo PJ Veelenturf, and Berend Jan Van Der Zwaag. A correlation-based fingerprint verification system. 2000.

[30] Luigi Di Stefano, Stefano Mattoccia, and Federico Tombari. Zncc-based template matching using bounded partial correlation. *Pattern Recognition Letters*, 26(14):2129–2134, 2005.

[31] Shou-Der Wei and Shang-Hong Lai. Fast template matching based on normalized cross correlation with adaptive multilevel winner update. *IEEE Transactions on Image Processing*, 17(11):2227–2235, 2008.

[32] E De Castro and C Morandi. Registration of translated and rotated images using finite fourier transforms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(5):700, 1987.

[33] Qin-sheng Chen, Michel Defrise, and Frank Deconinck. Symmetric phase-only matched filtering of fourier-mellin transforms for image registration and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(12):1156–1168, 1994.

[34] Yong Seok Heo, Kyoung Mu Lee, and Sang Uk Lee. Mutual information-based stereo matching combined with sift descriptor in log-chromaticity color space. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2009*, pages 445–452. IEEE, 2009.

[35] Roger M Dufour, Eric L Miller, and Nikolas P Galatsanos. Template matching based object recognition with unknown geometric parameters. *IEEE Transactions on Image Processing*, 11(12):1385–1396, 2002.

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

[37] Andrew R Webb. *Statistical Pattern Recognition*. John Wiley & Sons, 2003.

[38] Alan H Strahler. The use of prior probabilities in maximum likelihood classification of remotely sensed data. *Remote Sensing of Environment*, 10(2):135–163, 1980.

[39] Xin Tong. A plug-in approach to neyman-pearson classification. *Journal of Machine Learning Research*, 14(1):3011–3040, 2013.

[40] Gregory Ditzler, Robi Polikar, and Gail Rosen. A bootstrap based neyman-pearson

test for identifying variable importance. *IEEE Transactions on Neural Networks and Learning Systems*, 26(4):880–886, 2015.

[41] CA Trujillo-Pulgarín and M Orozco-Alzate. Parzen classification in generalised dissimilarity spaces. *Electronics Letters*, 49(3):1, 2013.

[42] Asgari Nezhad Yousef, Tokhmechi Behzad, Kamkar Rouhani Abolghasem, Sherkati Shahram, Kavousi Kaveh, and Jamali Amin. A combined parzen-wavelet approach for detection of vuggy zones in fractured carbonate reservoirs using petrophysical logs. *Journal of Petroleum Science and Engineering*, 119:1–7, 2014.

[43] V. N. Vapnik. *Statistical Learning Theory*. Wiley, New York, 1998.

[44] Simon Haykin and Neural Network. A comprehensive foundation. *Neural Networks*, 2(2004), 2004.

[45] James A Freeman and David M Skapura. *Algorithms, Applications, and Programming Techniques*. Addison-Wesley Publishing Company, USA, 1991.

[46] Christopher M Bishop. *Neural networks for Pattern Recognition*. Oxford university press, 1995.

[47] Mohamed Ibnkahla. Applications of neural networks to digital communications–a survey. *Signal Processing*, 80(7):1185–1215, 2000.

[48] Saroj K Meher, B Uma Shankar, and Ashish Ghosh. Wavelet-feature-based classifiers for multispectral remote-sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 45(6):1881–1886, 2007.

[49] Ashish Ghosh, B Uma Shankar, and Saroj K Meher. A novel approach to neuro-fuzzy classification. *Neural Networks*, 22(1):100–109, 2009.

[50] Pritpal Singh and Bhogeswar Borah. High-order fuzzy-neuro expert system for time series forecasting. *Knowledge-Based Systems*, 46:12–21, 2013.

[51] Pritpal Singh and Bhogeswar Borah. An effective neural network and fuzzy time series-based hybridized model to handle forecasting problems of two factors. *Knowledge and Information Systems*, 38(3):669–690, 2014.

[52] Šarūnas Raudys. Evolution and generalization of a single neurone: I. single-layer perceptron as seven statistical classifiers. *Neural Networks*, 11(2):283–296, 1998.

[53] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1989.

[54] Elias B Kosmatopoulos, Marios M Polycarpou, Manolis A Christodoulou, and Petros A Ioannou. High-order neural network structures for identification of dynamical systems. *IEEE Transactions on Neural Networks*, 6(2):422–431, 1995.

[55] Sheng Chen, Colin FN Cowan, and Peter M Grant. Orthogonal least squares learning algorithm for radial basis function networks. *IEEE Transactions on Neural Networks*, 2(2):302–309, 1991.

[56] Jamuna Kanta Sing, Dipak Kumar Basu, Mita Nasipuri, and Mahantapas Kundu. Face recognition using point symmetry distance-based rbf network. *Applied Soft Computing*, 7(1):58–70, 2007.

[57] Teuvo Kohonen. The self-organizing map. *Proceedings of the IEEE*, 78(9):1464–1480, 1990.

[58] Teuvo Kohonen. Essentials of the self-organizing map. *Neural Networks*, 37:52–65, 2013.

[59] Melody Y Kiang, Michael Y Hu, and Dorothy M Fisher. An extended self-organizing map network for market segmentationa telecommunication example. *Decision Support Systems*, 42(1):36–47, 2006.

[60] King Sun Fu. *Syntactic methods in Pattern Recognition*, volume 112. Elsevier, 1974.

[61] Theodoridis Pavlidis. *Structural Pattern Recognition*, volume 1. Springer, 2013.

[62] KS Fu. Step towards unification of syntactic and statistical pattern recognition. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 5(2):200–205, 1983.

[63] GR Pandey and V-T-V Nguyen. A comparative study of regression based methods in regional flood frequency analysis. *Journal of Hydrology*, 225(1):92–101, 1999.

[64] Christopher M Bishop and Michael E Tipping. Bayesian regression and classification. *Nato Science Series sub Series III Computer And Systems Sciences*, 190:267–288, 2003.

[65] Chris Tofallis. Least squares percentage regression. *Journal of Modern Applied Statistical Methods*, 2009.

[66] Tae-Hwan Kim and Christophe Muller. Two-stage quantile regression when the first stage is based on quantile regression. *The Econometrics Journal*, 7(1):218–231, 2004.

[67] Curtis B Storlie, Laura P Swiler, Jon C Helton, and Cedric J Sallaberry. Implementation and evaluation of nonparametric regression procedures for sensitivity analysis of computationally demanding models. *Reliability Engineering & System Safety*, 94(11):1735–1763, 2009.

[68] Leonard Kaufman and Peter J Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis*, volume 344. John Wiley & Sons, 2009.

[69] Anil K Jain and Richard C Dubes. *Algorithms for Clustering Data*. Prentice-Hall, Inc., 1988.

[70] Eduardo Raul Hruschka, Ricardo JGB Campello, Alex A Freitas, et al. A survey of evolutionary algorithms for clustering. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 39(2):133–155, 2009.

[71] Frank Höppner. *Fuzzy Cluster Analysis: Methods for Classification, Data analysis and Image Recognition.* John Wiley & Sons, 1999.

[72] Dorin Comaniciu and Peter Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002.

[73] Girish Punj and David W Stewart. Cluster analysis in marketing research: Review and suggestions for application. *Journal of Marketing Research*, pages 134–148, 1983.

[74] Thomas A Runkler and James C Bezdek. Web mining with relational clustering. *International Journal of Approximate Reasoning*, 32(2):217–236, 2003.

[75] Alexander Sturn, John Quackenbush, and Zlatko Trajanoski. Genesis: cluster analysis of microarray data. *Bioinformatics*, 18(1):207–208, 2002.

[76] David J Ketchen and Christopher L Shook. The application of cluster analysis in strategic management research: an analysis and critique. *Strategic Management Journal*, 17(6):441–458, 1996.

[77] Sotiris B Kotsiantis, I Zaharakis, and P Pintelas. Supervised machine learning: A review of classification techniques, 2007.

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

[79] John Richard Otukei and Thomas Blaschke. Land cover change assessment using decision trees, support vector machines and maximum likelihood classification algorithms. *International Journal of Applied Earth Observation and Geoinformation*, 12:S27–S31, 2010.

[80] V Sugumaran, V Muralidharan, and KI Ramachandran. Feature selection using decision tree and classification through proximal support vector machine for fault diagnostics of roller bearing. *Mechanical Systems and Signal Processing*, 21(2):930–942, 2007.

[81] Ron Kohavi and George H John. Wrappers for feature subset selection. *Artificial Intelligence*, 97(1):273–324, 1997.

[82] Helmut Schmid. Probabilistic part-ofispeech tagging using decision trees. In *New methods in language processing*, page 154. Routledge, 2013.

[83] Geoffrey McLachlan. *Discriminant Analysis and Statistical Pattern Recognition*, volume 544. John Wiley & Sons, 2004.

[84] Bernhard Scholkopft and Klaus-Robert Mullert. Fisher discriminant analysis with kernels. *Neural Networks for Signal Processing*, 1(1):1, 1999.

[85] Haitao Zhao and Pong Chi Yuen. Incremental linear discriminant analysis for face recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 38(1):210–221, 2008.

[86] Xuming He and Wing K Fung. High breakdown estimation for multiple populations with applications to discriminant analysis. *Journal of Multivariate Analysis*, 72(2):151–162, 2000.

[87] Harry Zhang. The optimality of naive bayes. *AA*, 1(2):3, 2004.

[88] Stuart J Russell and Peter Norvig. Artificial intelligence: a modern approach (international edition). 2002.

[89] Wei Wei, Shyam Visweswaran, and Gregory F Cooper. The application of naive bayes model averaging to predict alzheimer's disease from genome-wide data. *Journal of the American Medical Informatics Association*, 18(4):370–375, 2011.

[90] Mrutyunjaya Panda and Manas Ranjan Patra. Network intrusion detection using naive bayes. *International Journal of Computer Science and Network Security*, 7(12):258–263, 2007.

[91] Liangxiao Jiang, Harry Zhang, and Zhihua Cai. A novel bayes model: Hidden naive bayes. *IEEE Transactions on Knowledge and Data Engineering*, 21(10):1361–1371, 2009.

[92] Ziheng Yang. Maximum likelihood phylogenetic estimation from dna sequences with variable rates over sites: approximate methods. *Journal of Molecular Evolution*, 39(3):306–314, 1994.

[93] Wen Wei and Jerry M Mendel. Maximum-likelihood classification for digital amplitude-phase modulations. *IEEE Transactions on Communications*, 48(2):189–193, 2000.

[94] Karen C Seto and Weiguo Liu. Comparing artmap neural network with the maximum-likelihood classifier for detecting urban change. *Photogrammetric Engineering & Remote Sensing*, 69(9):981–990, 2003.

[95] Kevin Beyer, Jonathan Goldstein, Raghu Ramakrishnan, and Uri Shaft. When is nearest neighbor meaningful? In *International conference on database theory*, pages 217–235. Springer, 1999.

[96] Nikunj C Oza and Kagan Tumer. Classifier ensembles: Select real-world applications. *Information Fusion*, 9(1):4–20, 2008.

[97] Slavko Vasilic and Mladen Kezunovic. Fuzzy art neural network algorithm for classifying the power system faults. *IEEE Transactions on Power Delivery*, 20(2):1306–1314, 2005.

[98] Jens Keuchel, Simone Naumann, Matthias Heiler, and Alexander Siegmund. Automatic land cover analysis for tenerife by supervised classification using remotely sensed data. *Remote sensing of Environment*, 86(4):530–541, 2003.

[99] Alec Go, Richa Bhayani, and Lei Huang. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, 1:12, 2009.

[100] Yusuf Artan, Aytekin Oto, and Imam Samil Yetik. Cross-device automated prostate cancer localization with multiparametric mri. *IEEE Transactions on Image Processing*, 22(12):5385–5394, 2013.

[101] Vanessa Lucieer, Nicole A Hill, Neville S Barrett, and Scott Nichol. Do marine substrates lookand soundthe same? supervised classification of multibeam acoustic data using autonomous underwater vehicle images. *Estuarine, Coastal and Shelf Science*, 117:94–106, 2013.

[102] Chenghai Yang, Gary N Odvody, Carlos J Fernandez, Juan A Landivar, Richard R Minzenmayer, and Robert L Nichols. Evaluating unsupervised and supervised image classification methods for mapping cotton root rot. *Precision Agriculture*, 16(2):201–215, 2015.

[103] AV Egorov, MC Hansen, DP Roy, A Kommareddy, and PV Potapov. Image interpretation-guided supervised classification using nested segmentation. *Remote Sensing of Environment*, 165:135–147, 2015.

[104] Jordi Muñoz-Marí, Lorenzo Bruzzone, and Gustavo Camps-Valls. A support vector domain description approach to supervised classification of remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 45(8):2683–2692, 2007.

[105] Suju Rajan, Joydeep Ghosh, and Melba M Crawford. An active learning approach to hyperspectral data classification. *IEEE Transactions on Geoscience and Remote Sensing*, 46(4):1231–1242, 2008.

[106] Xiaojin Zhu and Andrew B Goldberg. Introduction to semi-supervised learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 3(1):1–130, 2009.

[107] T. Joachims. Transductive inference for text classification using support vector machines. In *Proc. 16th ICML*, pages 200–209, 1999.

[108] Masashi Inoue and Naonori Ueda. Exploitation of unlabeled sequences in hidden markov models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(12):1570–1581, 2003.

[109] O. Chapelle and A. Zien. Semi-supervised classification by low density separation. In *Proc. 10th International Workshop on Artificial Intelligence and Statistics*, 2005.

[110] Lorenzo Bruzzone and Claudio Persello. A novel context-sensitive semisupervised svm classifier robust to mislabeled training samples. *IEEE Transactions on Geoscience and Remote Sensing*, 47(7):2142–2154, 2009.

[111] Shiming Xiang, Feiping Nie, and Changshui Zhang. Semi-supervised classification via local spline regression. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(11):2039–2053, 2010.

[112] Lei Shi, Xinming Ma, Lei Xi, Qiguo Duan, and Jingying Zhao. Rough set and

ensemble learning based semi-supervised algorithm for text classification. *Expert Systems with Applications*, 38(5):6300–6306, 2011.

[113] Hang Su, Zhaozheng Yin, Seungil Huh, and Takeo Kanade. Cell segmentation in phase contrast microscopy images via semi-supervised classification over optics-related features. *Medical image analysis*, 17(7):746–765, 2013.

[114] Haitao Gan, Nong Sang, Rui Huang, Xiaojun Tong, and Zhiping Dan. Using clustering analysis to improve semi-supervised classification. *Neurocomputing*, 101:290–298, 2013.

[115] Gustavo Camps-Valls, Devis Tuia, Lorenzo Bruzzone, and Jon Atli Benediktsson. Advances in hyperspectral image classification: Earth monitoring with statistical learning methods. *IEEE Signal Processing Magazine*, 31(1):45–54, 2014.

[116] Ishtiaq Ahmed, Rahman Ali, Donghai Guan, Young-Koo Lee, Sungyoung Lee, and TaeChoong Chung. Semi-supervised learning using frequent itemset and ensemble learning for sms classification. *Expert Systems with Applications*, 42(3):1065–1073, 2015.

[117] Burr Settles. Active learning literature survey. *University of Wisconsin, Madison*, 52(55-66):11, 2010.

[118] David Cohn, Les Atlas, and Richard Ladner. Improving generalization with active learning. *Machine Learning*, 15(2):201–221, 1994.

[119] Simon Tong and Daphne Koller. Active learning for parameter estimation in bayesian networks. In *NIPS*, volume 13, pages 647–653, 2000.

[120] Simon Tong and Daphne Koller. Active learning for structure in bayesian networks. In *International joint conference on artificial intelligence*, volume 17, pages 863–869. LAWRENCE ERLBAUM ASSOCIATES LTD, 2001.

[121] David JC MacKay. Information-based objective functions for active data selection. *Neural Computation*, 4(4):590–604, 1992.

[122] David A Cohn. Neural network exploration using optimal experiment design. *Neural Networks*, 9(6):1071–1083, 1996.

[123] David A Cohn, Zoubin Ghahramani, and Michael I Jordan. Active learning with statistical models. *Journal of Artificial Intelligence Research*, 1996.

[124] Greg Schohn and David Cohn. Less is more: Active learning with support vector machines. In *ICML*, pages 839–846. Citeseer, 2000.

[125] Begüm Demir, Francesca Bovolo, and Lorenzo Bruzzone. Detection of land-cover transitions in multitemporal remote sensing images with active-learning-based compound classification. *IEEE Transactions on Geoscience and Remote Sensing*, 50(5):1930–1941, 2012.

[126] Begüm Demir, Francesca Bovolo, and Lorenzo Bruzzone. Updating land-cover

maps by classification of image time series: A novel change-detection-driven transfer learning approach. *IEEE Transactions on Geoscience and Remote Sensing*, 51(1):300–312, 2013.

[127] Shusen Zhou, Qingcai Chen, and Xiaolong Wang. Active deep learning method for semi-supervised sentiment classification. *Neurocomputing*, 120:536–546, 2013.

[128] Richard O Duda, Peter E Hart, and David G Stork. *Pattern Classification*. John Wiley & Sons, 2012.

[129] Anil K Jain. Data clustering: 50 years beyond k-means. *Pattern Recognition Letters*, 31(8):651–666, 2010.

[130] Matthew C Clark, Lawrence O Hall, Dmitry B Goldgof, Laurence P Clarke, Robert P Velthuizen, and Martin S Silbiger. Mri segmentation using fuzzy clustering techniques. *IEEE Engineering in Medicine and Biology Magazine*, 13(5):730–742, 1994.

[131] Lorenzo Bruzzone and Diego F Prieto. Automatic analysis of the difference image for unsupervised change detection. *IEEE Transactions on Geoscience and Remote sensing*, 38(3):1171–1182, 2000.

[132] Sanghoon Lee and Melba M Crawford. Unsupervised multistage image classification using hierarchical clustering with a bayesian similarity measure. *IEEE Transactions on Image Processing*, 14(3):312–320, 2005.

[133] Dali Wei, Yichao Jin, Serdar Vural, Klaus Moessner, and Rahim Tafazolli. An energy-efficient clustering solution for wireless sensor networks. *IEEE Transactions on Wireless Communications*, 10(11):3973–3983, 2011.

[134] Ehsan Elhamifar and Rene Vidal. Sparse subspace clustering: Algorithm, theory, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11):2765–2781, 2013.

[135] Grigorios Tzortzis and Aristidis Likas. The minmax k-means clustering algorithm. *Pattern Recognition*, 47(7):2505–2516, 2014.

[136] Biplab Banerjee, Francesca Bovolo, Avik Bhattacharya, Lorenzo Bruzzone, Subhasis Chaudhuri, and B Krishna Mohan. A new self-training-based unsupervised satellite image classification technique using cluster ensemble strategy. *IEEE Geoscience and Remote Sensing Letters*, 12(4):741–745, 2015.

[137] D Dutta Majumder and B Chanda. Digital image processing and analysis. *Prentince Hall of India Private Limited (September 2007)*, 2006.

[138] Rafael C Gonzalez and Richard E Woods. Digital image processing. *Nueva Jersey*, 2008.

[139] N. R. Pal and S. K. Pal. A review on image segmentation techniques. *Pattern Recognition*, 26(9):1277–1294, 1993.

[140] Yu Jin Zhang. A survey on evaluation methods for image segmentation. *Pattern Recognition*, 29(8):1335–1346, 1996.

[141] King-Sun Fu and JK Mui. A survey on image segmentation. *Pattern Recognition*, 13(1):3–16, 1981.

[142] James C Bezdek, LO Hall, and L_P Clarke. Review of mr image segmentation techniques using pattern recognition. *Medical Physics*, 20(4):1033–1048, 1992.

[143] Zhenyu Wu and Richard Leahy. An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(11):1101–1113, 1993.

[144] W ladys law Skarbek and Andreas Koschan. Colour image segmentation a survey. *IEEE Transactions on Circuits and Systems for Video Technology*, 14, 1994.

[145] OLEG Okun, MATTI Pietikäinen, O Okun, and M Pietikäinen. A survey of texture-based methods for document layout analysis. In *Proc. Workshop on Texture Analysis in Machine Vision.* in: Proc. Workshop on Texture Analysis in Machine Vision, June 14-15, Oulu, Finland, 137-148., 1999.

[146] Bulent Sankur and Mehmet Sezgin. Image thresholding techniques: A survey over categories. *Pattern Recognition*, 34(2):1573–1583, 2001.

[147] Warren S Sarle. Algorithms for clustering data. *Technometrics*, 32(2):227–229, 1990.

[148] Jitendra Malik, Serge Belongie, Thomas Leung, and Jianbo Shi. Contour and texture analysis for image segmentation. In *Perceptual Organization for artificial vision systems*, pages 139–172. Springer, 2000.

[149] M. Sezgin and B. Sankur. Survey over image thresholding techniques and quantitative performance evaluation. *J. Electronic Imaging*, 13(1):146–165, 2004.

[150] P. K. Sahoo, S. Soltani, and A. Wong. A survey of thresholding techniques. *Computer Vision Graphics and Image Process*, 41(2):233–260, 1988.

[151] Sudip Kumar Adhikari, Jamuna Kanta Sing, Dipak Kumar Basu, and Mita Nasipuri. Conditional spatial fuzzy c-means clustering algorithm for segmentation of mri images. *Applied Soft Computing*, 34:758–769, 2015.

[152] Sang Uk Lee, Seok Yoon Chung, and Rae Hong Park. A comparative performance study of several global thresholding techniques for segmentation. *Computer Vision, Graphics, and Image Processing*, 52(2):171–190, 1990.

[153] Kamal Hammouche, Moussa Diaf, and Patrick Siarry. A comparative study of various meta-heuristic techniques applied to the multilevel thresholding problem. *Engineering Applications of Artificial Intelligence*, 23(5):676–688, 2010.

[154] Ming-Huwi Horng. A multilevel image thresholding using the honey bee mating optimization. *Applied Mathematics and Computation*, 215(9):3302–3310, 2010.

[155] Hao Gao, Wenbo Xu, Jun Sun, and Yulan Tang. Multilevel thresholding for image segmentation through an improved quantum-behaved particle swarm algorithm. *IEEE Transactions on Instrumentation and Measurement*, 59(4):934–946, 2010.

[156] Khang Siang Tan and Nor Ashidi Mat Isa. Color image segmentation using histogram thresholding–fuzzy c-means hybrid approach. *Pattern Recognition*, 44(1):1–15, 2011.

[157] Ming-Huwi Horng. Multilevel thresholding selection based on the artificial bee colony algorithm for image segmentation. *Expert Systems with Applications*, 38(11):13785–13791, 2011.

[158] PD Sathya and R Kayalvizhi. Modified bacterial foraging algorithm based multilevel thresholding for image segmentation. *Engineering Applications of Artificial Intelligence*, 24(4):595–615, 2011.

[159] Qingyong Li, Weitao Lu, and Jun Yang. A hybrid thresholding algorithm for cloud detection on ground-based color images. *Journal of Atmospheric and Oceanic Technology*, 28(10):1286–1296, 2011.

[160] V. Sindhwani and S. S. Keerthi. Large scale semisupervised linear SVMs. In *Proc. SIGIR*, 2006.

[161] M. M. Adankon and M Cheriet. Learning semi-supervised SVM with genetic algorithm. In *Proc. Int. conf. on Neural Networks*, 2007.

[162] Y. Chen, G. Wang, and S. Dong. Learning with progressive transductive support vector machine. *Pattern Recognition Letters*, 24:1845–1855, 2003.

[163] Ujjwal Maulik and Debasis Chakraborty. Learning with transductive svm for semisupervised pixel classification of remote sensing imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 77:66–78, 2013.

[164] L. Bruzzone, M. Chi, and M. Marconcini. A novel transductive SVM for semisupervised classification of remote-sensing images. *IEEE Transaction of Geoscience and Remote Sensing*, 44(11):3363–3373, 2006.

[165] Keinosuke Fukunaga and L Hostetler. Optimization of k nearest neighbor density estimates. *IEEE Transactions on Information Theory*, 19(3):320–326, 1973.

[166] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[167] David Maxwell Chickering, David Heckerman, and Christopher Meek. A bayesian approach to learning bayesian networks with local structure. In *Proceedings of the Thirteenth conference on Uncertainty in artificial intelligence*, pages 80–89. Morgan Kaufmann Publishers Inc., 1997.

[168] Lian-Zhi Huo and Ping Tang. A batch-mode active learning algorithm using region-partitioning diversity for svm classifier. *IEEE Journal of Selected Topics in Applied*

Earth Observations and Remote Sensing, 7(4):1036–1046, 2014.

[169] Devis Tuia, Frédéric Ratle, Fabio Pacifici, Mikhail F Kanevski, and William J Emery. Active learning methods for remote sensing image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 47(7):2218–2232, 2009.

[170] Klaus Brinker. Incorporating diversity in active learning with support vector machines. In *ICML*, volume 3, pages 59–66, 2003.

[171] Pabitra Mitra, B Uma Shankar, and Sankar K Pal. Segmentation of multispectral remote sensing images using active support vector machines. *Pattern Recognition Letters*, 25(9):1067–1074, 2004.

[172] N. Otsu. A threshold selection method from gray level histograms. *IEEE Transactions on Systems, Man and Cybernetics*, 9:62–66, 1979.

[173] J. Kittler and J. Illingworth. Minimum error thresholding. *Pattern Recognition*, 19(1):41–47, 1986.

[174] Y. Qiaoa, Q. Hua, G Qiana, S. Luob, and W. L. Nowinskia. Thresholding based on variance and intensity contrast. *Pattern Recognition*, 40:596–608, 2007.

[175] S. Patra, R. Gautam, and A. Singla. A novel context sensitive multilevel thresholding for image segmentations. *Applied Soft Computing*, 23:122–127, 2014.

[176] A. Rosenfeld and P. D. Torre. Histogram concavity as an aid in threshold selection. *IEEE Transactions on Systems, Man and Cybernetics*, 13(3):231–235, 1983.

[177] P. Ghamisi, M. S. Couceiro, J. A. Benediktsson, and M. F. N. Ferreira. An efficient method for segmentation of images based on fractional calculus and natural selection. *Expert Systems with Applications*, 39:12407–12417, 2012.

[178] U. Maulik and S. Bandyopadhyay. Genetic algorithm-based clustering technique. *Pattern Recognition*, 33:1455–1965, 2000.

[179] J. N. Kapur, P. K. Sahoo, and A. K. C. Wong. A new method for gray-level picture thresholding using the entropy of the histogram. *Computer Vision, Graphics, Image Processing*, 29(3):273–285, 1985.

[180] S. Patra, S. Ghosh, and A. Ghosh. Histogram thresholding for unsupervised change detection of remote sensing images. *Int. J. Remote Sensing*, 32(21):6071–6089, 2011.

[181] K. Nigam, A. McCallum, and T. Mitchell. Learning to classify text from labeled and unlabeled documents. In *Proc. AAAI*, pages 792–799, 1998.

[182] X. Zhu. Semi-supervised learning literature survey. Technical Report 1530, Computer Sciences, University of Wisconsin-Madison, 2005.

[183] C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.

[184] Behzad M Shahshahani and David A Landgrebe. The effect of unlabeled samples in reducing the small sample size problem and mitigating the hughes phenomenon.

IEEE Transactions on Geoscience and Remote Sensing, 32(5):1087–1095, 1994.

[185] S. K. Pal and S. Mitra. Multilayer perceptron, fuzzy sets, and classification. *IEEE Transaction on Neural Networks*, 3(5):683–697, 1992.

[186] A. Frank and A. Asuncion. *UCI Machine Learning Repository*, 2010. University of California, School of Information and Computer Sciences, `http://archive.ics.uci.edu/ml`.

[187] S. Patra and L. Bruzzone. A fast cluster-assumption based active learning technique for classification of remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 49(5):1617–1626, 2011.

[188] Chin-Chung Chang and Chih-Jen Lin. *LIBSVM: a library for support vector machine*, 2001. Software available at `http://csie.ntu.edu.tw/~cjlin/libsvm`.

[189] H Sebastian Seung, Manfred Opper, and Haim Sompolinsky. Query by committee. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 287–294. ACM, 1992.

[190] Yoav Freund, H Sebastian Seung, Eli Shamir, and Naftali Tishby. Selective sampling using the query by committee algorithm. *Machine Learning*, 28(2-3):133–168, 1997.

[191] Naoki Abe Hiroshi Mamitsuka. Query learning strategies using boosting and bagging. In *Machine Learning: Proceedings of the Fifteenth International Conference (ICML'98)*, page 1. Morgan Kaufmann Pub, 1998.

[192] Kenji Fukumizu. Statistical active learning in multilayer perceptrons. *IEEE Transactions on Neural Networks*, 11(1):17–26, 2000.

[193] Colin Campbell, Nello Cristianini, Alex Smola, et al. Query learning with large margin classifiers. In *ICML*, pages 111–118, 2000.

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

[195] Shouxian Cheng and Frank Y Shih. An improved incremental training algorithm for support vector machines using active query. *Pattern Recognition*, 40(3):964–971, 2007.

[196] Begüm Demir, Claudio Persello, and Lorenzo Bruzzone. Batch-mode active-learning methods for the interactive classification of remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 49(3):1014–1031, 2011.

[197] Swarnajyoti Patra and Lorenzo Bruzzone. A fast cluster-assumption based active-learning technique for classification of remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 49(5):1617–1626, 2011.

[198] Swarnajyoti Patra and Lorenzo Bruzzone. A cluster-assumption based batch mode active learning technique. *Pattern Recognition Letters*, 33(9):1042–1048, 2012.

[199] Swarnajyoti Patra and Lorenzo Bruzzone. A novel som-svm-based active learning technique for remote sensing image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 52(11):6899–6910, 2014.

[200] Sheng jun Huang, Rong Jin, and Zhi hua Zhou. Active learning by querying informative and representative examples. In J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 892–900. Curran Associates, Inc., 2010.

[201] Mingkun Li and Ishwar K Sethi. Confidence-based active learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(8):1251–1261, 2006.

[202] Vladimir Vapnik. *The Nature of Statistical Learning Theory*. Springer Verlag, New York, NY, USA, 2 edition, 2001.

[203] Olga Brezhneva and Alexey A Tret'yakov. An elementary proof of the karush–kuhn–tucker theorem in normed linear spaces for problems with a finite number of inequality constraints. *Optimization*, 60(5):613–618, 2011.

[204] Han Jiawei and Micheline Kamber. Data mining: concepts and techniques. *San Francisco, CA, itd: Morgan Kaufmann*, 5, 2001.

[205] Christopher JC Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.

[206] N Cristianini and J Shawe-Taylor. An introduction to support vector machines (cambridge, 2000.

[207] Abe Shigeo. Support vector machines for pattern classification. *Advances in Pattern Recognition, Springer, Heidelberg*, 2005.

[208] Jinchang Ren. Ann vs. svm: Which one performs better in classification of mccs in mammogram imaging. *Knowledge-Based Systems*, 26:144–153, 2012.

[209] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

[210] L. K. Huang and M. J. J. Wang. Image thresholding by minimizing the measures of fuzziness. *Pattern Recognition*, 28:41–51, 1995.

[211] L. K. Huang and M. J. J. Wang. Thresholding technique with adaptive window selection for uneven lighting image. *Pattern Recognition Letters*, 26:801–808, 2005.

[212] S. Ghosh, L. Bruzzone, S. Patra, F. Bovolo, and A. Ghosh. A context-sensitive technique for unsupervised change detection based on hopfield-type neural networks. *IEEE Transaction of Geosciences and Remote Sensing*, 45(3):778–789, 2007.

[213] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Singapore: John Wiley, 2001.

[214] Y. Liu and S. N. Srihari. Document image binarization based on texture features.

IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(5):540–544, 1997.

[215] Y. Q. Song, Z. Liu, J. M. Chen, F. Zhu, and C. H. Xie. Medical image segmentation based on non-parametric mixture models with spatial information. *Signal, Image and Video Processing*, 6(4):569–578, 2012.

[216] V. P. Ananthi, P. Balasubramaniam, and C. P. Lim. Segmentation of gray scale image based on intuitionistic fuzzy sets constructed from several membership functions. *Pattern Recognition*, 47:3870–3880, 2014.

[217] O. J. Tobias and R. Seara. Image segmentation by histogram thresholding using fuzzy sets. *IEEE Transactions on Image Processing*, 11(12):1457–1465, 2012.

[218] B. Akay. A study on particle swarm optimization and artificial bee colony algorithms for multilevel thresholding. *Applied Soft Computing*, 13:3066–3091, 2013.

[219] E. Cuevas, F. Sencin-Echauri, D. Zaldivar, and M. Prez-Cisneros. A multi-threshold segmentation approach based on artificial bee colony optimization. *Applied Intelligence*, 37(3):321–336, 2012.

[220] J. C. Yen, F. J. Chang, and S. Chang. A new criterion for automatic multilevel thresholding. *IEEE Transactions on Image Processing*, 4(3):370–378, 1995.

[221] A. S. Abutaleb. Automatic thresholding of gray-level pictures using two dimensional entropy. *Computer Vision Graphics and Image Processing*, 47(1):22–32, 1989.

[222] S. Sarkar and S. Das. Multilevel image thresholding based on 2D histogram and maximum tsallis entropy a differential evolution approach. *IEEE Transactions on Image Processing*, 22(12):4788–4797, 2013.

[223] A. Yimit, Y. Hagihara, T. Miyoshi, and Y. Hagihara. 2-D direction histogram based entropic thresholding. *Neurocomputing*, 120(23):287–297, 2013.

[224] M. Siahi, J. Razjouyan, O. Khayat, A. A. Mansouri, and Z. Azimi. A multi-class bi-level thresholding method for accurate anthropometric measurements of scanned plantar images. *Signal, Image and Video Processing*, 9(2):295–304, 2015.

[225] Y. Y. Xiao, Z. Cao, and S. Zhong. New entropic thresholding approach using gray-level spatial correlation histogram. *Optical Engineering*, 49(12):127007, 2010.

[226] Y. Y. Xiao, Z. Cao, and W. Zhuo. Type-2 fuzzy thresholding using GLSC histogram of human visual nonlinearity characteristics. *Optical Express*, 19(11):10656–10672, 2011.

[227] Y. Y. Xiao, Z. Cao, and J. Yuan. Entropic image thresholding based on GLGM histogram. *Pattern Recognition Letters*, 40:47–55, 2014.

[228] D Goldberg. Genetic algorithms in search, optimization, and machine learning, 1989.

[229] Lawrence Davis. *Handbook of Genetic Algorithms*. 1991.

[230] D. L. Davis and D. W. Bouldin. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-1(2):224–227, 1979.

[231] D. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning.* New York: Addison-Wesley, 1989.

[232] M. Halkidi and M. Vazirgiannis. Clustering validity assessment: Finding the optimal partitioning of a data set. In *In: Proc. ICDM, California, USA*. 2001.

List of Publications

International Journal

1. Anshu Singla, Swarnajyoti Patra, Lorenzo Bruzzone. *"A Novel Classification Technique Based on Progressive Transductive SVM Learning"*, Pattern Recognition Letters, Elsevier, Vol 42, pages 101-106, 2014 (SCI: Impact Factor: 1.551).
2. Swarnajyoti Patra, Rahul Gautam, Anshu Singla. *"A Novel Context Sensitive Multilevel Thresholding Technique for Image Segmentation"*. Applied Soft Computing, Vol 23, pages 122-127, 2014 (SCI: Impact Factor: 2.856).
3. Anshu Singla, Swarnajyoti Patra. *"A Fast Automatic Threshold Selection Technique for Image Segmentation"*. Signal Image and Video Processing, Vol 11(2), pages 243-250, 2017 (SCI: Impact Factor: 0.872).
4. Anshu Singla, Swarnajyoti Patra. *"A Fast Partition-based Batch-Mode Active Learning Technique using SVM Classifier"*. Soft Computing, doi:10.1007/s00500-017-2645-0, 2017 (SCI: Impact Factor: 2.472).

Book Chapter

1. Anshu Singla, Swarnajyoti Patra. *"A Context Sensitive Thresholding Technique for Automatic Image Segmentation"*. Computational Intelligence in Data Mining, Vol 2, Smart Innovation, Systems and Technologies, Vol 32, pages 19-25, 2015.

International Conference

1. Anshu Singla, Swarnajyoti Patra. *" PSO Based Context Sensitive Thresholding Technique for Automatic Image Segmentation"*. Proceedings of 6th international conference on Soft Computing for problem solving. SocProS 2016 - Volume 2, Advances in Intelligent Systems and Computing, Volume 547, pages 151-162, 2017.