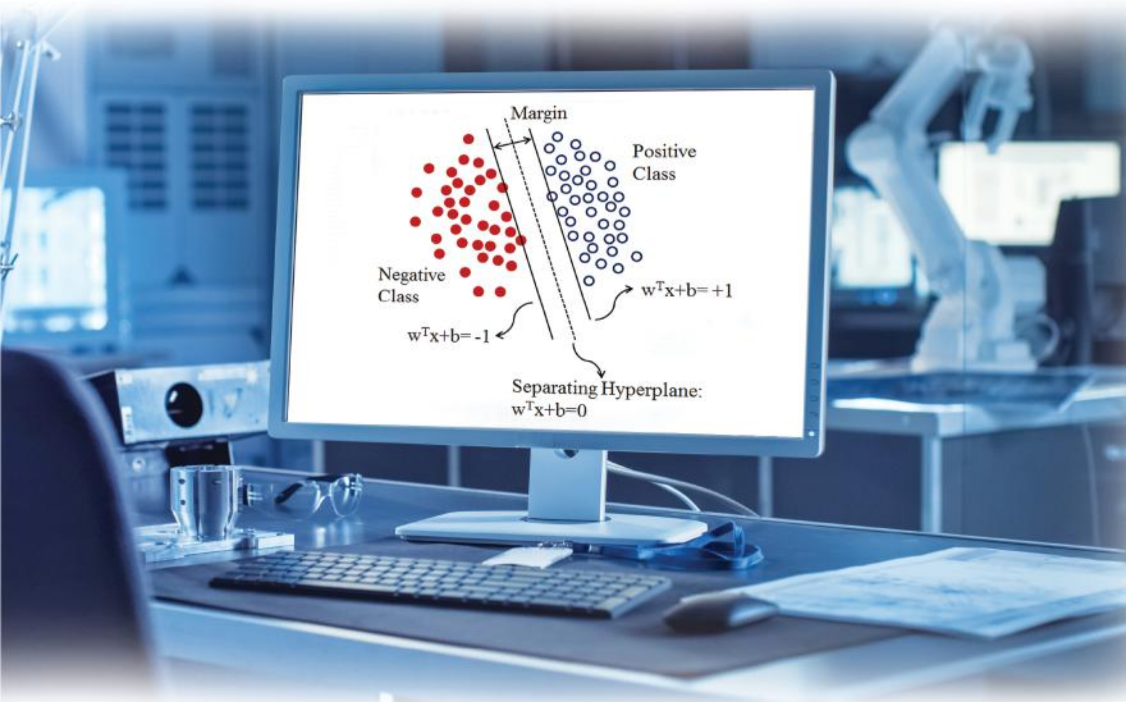
Pooja Saigal, PhD
Editor



Support Vector Machines

Evolution and Applications

COMPUTER SCIENCE, TECHNOLOGY AND APPLICATIONS

NOVA

COMPUTER SCIENCE, TECHNOLOGY AND APPLICATIONS

SUPPORT-VECTOR MACHINES

EVOLUTION AND APPLICATIONS

COMPUTER SCIENCE, TECHNOLOGY AND APPLICATIONS

Additional books and e-books in this series can be found on Nova's website under the Series tab.

COMPUTER SCIENCE, TECHNOLOGY AND APPLICATIONS

SUPPORT-VECTOR MACHINES

EVOLUTION AND APPLICATIONS

POOJA SAIGAL
EDITOR

nova
science publishers
New York

Library of Congress Cataloging-in-Publication Data

ISBN: 978-1-53618-865-3(eBook)

Published by Nova Science Publishers, Inc. † New York

CONTENTS

PREFACE

This book “Support Vector Machines: Evolution and Applications” includes the basics of Support Vector Machines (SVM), its evolution and applications in diverse fields. SVM is an efficient supervised learning approach which is popularly used for pattern recognition, medical image classification, face recognition and various other applications. In the last 25 years, a lot of research has been done to extend the use of SVM to a variety of domains. This book is an attempt to present the description of a conventional SVM, along with discussion of its different versions and recent application areas.

The first chapter of this book introduces the Support Vector Machines and presents the optimization problems for a conventional SVM. Another chapter discusses the journey of SVM over a period of more than two decades. SVM is proposed as a separating hyperplane classifier that partitions the data belonging to two classes. Later on, various versions of SVM are proposed that obtain two hyperplanes instead of one. A few of these variants of SVM are discussed in this book.

The major part of this book discusses some interesting applications of SVM in areas like Quantitative Diagnosis of Rotor Vibration Process Faults through Power Spectrum Entropy-based SVM, Hardware Architectures of SVM Applied in Pattern Recognition Systems, Speaker Recognition using SVM, Classification of iron ore in mines and

Simultaneous prediction of the density and viscosity for the ternary system water– ethanol–ethylene glycol ionic liquids.

The latter part of the book is dedicated to various approaches for extension of SVM and similar classifiers to multi-category framework, so that they can be used for classification of data with more than two classes.

ACKNOWLEDGMENTS

The journey of compiling and editing this book has been a very enriching and fulfilling experience for me. It took an immense amount of effort and it would not have been possible without the invaluable contributions of thoughtful and creative people. I would like to take this opportunity to thank all of them.

I would first like to thank the publishing house "Nova Science Publishers" for giving me the responsibility of editing this book. The entire team has been very supportive. My queries were promptly answered and I was guided throughout the editing process.

I would like to express my deep and sincere gratitude to "Vivekananda Institute of Professional Studies, New Delhi, India", my workplace, which provided me a professional environment to carry out the research work. I am extremely grateful to Dr. S.C Vats, Chairman, VIPS and Mr. Suneet Vats, Vice-Chairman, VIPS, for appreciating my work. They have been the constant source of motivation for me. I would also like to thank Prof. Sidharth Mishra, Chairperson, VSIT and Prof. (Dr.) Supriya Madan, Dean, VSIT, for their support and motivation.

I am overwhelmed in all humbleness to acknowledge the contributions of all the authors, who have helped me to organize this book in the best possible way. Their work is the heart and core of this book. I would like to thank all of them for submitting the work on time and as per the standards required for publishing.

My sincere thanks go to Dr. Reshma Khemchandani Rastogi and Dr. Suresh Chandra, for guiding me throughout.

And finally, I would like to express my deepest gratitude for my loving, caring and supporting family. This book would not have been possible without their constant support and encouragement. My husband Amit Saigal has supported me unconditionally in every sphere of life and has motivated me throughout. My loving daughter Akaisha inspires me to explore new territories. I am grateful to my parents-in-law who supported me with great patience and took upon my responsibilities at home, in my absence. I could not thank my parents Mr. Shiv Kumar Khanna and Mrs. Neelam Khanna enough, for supporting me emotionally and believing in me more than anyone else could. I am blessed to have Nidhi, my sister and Vaibhav, my brother, whose love and support helped me to get out of hard times.

List of Acronyms

ADALINE	ADAptive LInear NEuron
ANFIS	Adaptive neuro-fuzzy inference system
ASIC	Application Specific Integrated Circuit
ASR	Automatic Speech Recognition
ASSP	Application Specific Standard Product
BHQ	Banded hematite quartz
BT	Binary tree
DAG	Directed acyclic graph
EEG	Electroencephalography
ERM	Empirical Risk Minimisation
FFBDO	Flaky friable blue dust ore
FN	False Negative
FP	False Positive
FPGA	Field Programmable Gate Array
FSM	Finite State Machine
GEPSVM	Generalized Eigenvalue Proximal Support Vector Machine
GMM	Gaussian Mixture Model
HLS	High-Level Synthesis
KKT	Karush-Kuhn-Tucker
LIO	Lateritic iron ore

LS-SVM	Least-squares SVM
MAC	Multiplier-Accumulator
MFCC	Mel-frequency Cepstral Coefficients
MIO	Massive iron ore
ML	Machine Learning
MLP	Multilayer Perceptron
NPHC	Non-parallel hyperplane classifiers
OAA	One-against-all
OAO	One-against-one
PCA	Principal component analysis
PPSE	Process power spectrum entropy
PSO	Particle Swarm Optimization
PSVM	Proximal Support Vector Machine
QPP	Quadratic programming problems
SC	Speech Corpus
SMO	Sequential Minimal Optimization
SoC	System on Chip
SR	Speaker recognition
SRM	Structural Risk Minimisation
SVM	Support Vector Machine
TDS	Ternary Decision Structure
TM	True Negative
TP	True Positive
TWSVM	Twin support vector machine
VLSI	Very Large-Scale Integration

In: Support-Vector Machines
Editor: Pooja Saigal
ISBN: 978-1-53618-757-1

Chapter 1

INTRODUCTION TO SUPPORT VECTOR MACHINES

Pooja Saigal, PhD*
Vivekananda School of Information Technology,
Vivekananda Institute of Professional Studies, New Delhi, India

Abstract

Support Vector Machines (SVMs) have been used as an efficient classification tool in the last two decades. SVMs are based on statistical learning theory. The formulation of SVM embodies the Structural Risk Minimisation (SRM) principle. Before SVM, the conventional neural networks used Empirical Risk Minimisation (ERM). But SRM proved to give superior results than its predecessors. SVM gives global optimum solution by solving a convex quadratic programming problem. It has good generalization ability and can efficiently develop the classifier model in lesser time than neural networks or other probability based models. This chapter explains the formulation of a classical SVM, and introduces the terminology associated with SVM.

Keywords: classifiers, support vector machines, margin, efficiency, generalization ability

*Corresponding Author's Email: pooja.saigal@vips.edu, saigal.pooja.in@gmail.com.

1. INTRODUCTION

The advent of Support Vector Machines (SVMs) brought a new revolution in the machine learning community. SVMs are based on sound theoretical framework and proved to be an efficient classifier. SVM offers a meticulous approach to machine learning problems. SVMs were originally proposed for regression problems [1] by Cortes and Vapnik in 2005 and further studied by Smola and Vapnik [2]. Later on, SVM was modified to be used as a binary classifier [3]. SVM constructs its solution in terms of a subset of the training input. SVM has been extensively used for classification, regression, pattern recognition, and feature reduction tasks. The idea of SVM is to reduce the generalization error by maximizing the margin of a hyperplane. SVM is based on the strong foundation of statistical learning theory. SVMs have been successfully applied to a wide range of areas like handwritten digit recognition, object recognition, speaker identification, pattern recognition, text categorization, biomedicine, financial time series forecasting and many more. This chapter focuses on SVM for supervised classification tasks only and provides SVM optimization problem formulations for linearly separable as well as linearly non-separable input spaces. SVM can efficiently manage unbalanced data.

2. GEOMETRIC INTERPRETATION OF SVM

The classification tasks can be perform by machine learning approaches that can be broadly classified as discriminant or generative. Discriminant technique uses training datasets and aims to find a function that can correctly predict labels for unseen patterns [3]. For any new test pattern x, discriminant classification function assigns it to one of the different classes that are a part of the classification problem. Whereas, the generative machine learning approaches require computations of conditional probability distributions and are mostly used when prediction involves outlier detection. Discriminant approaches are better than generative ones as they require fewer computational resources and less training data, especially for a multidimensional feature space. Therefore, from a geometric perspective, learning a classifier is equivalent to finding the equation for a multidimensional hyperplane or hyper-surface that best separates the classes in the feature space [4, 5].

SVM is a discriminant technique that solves a convex optimization problem. It obtains the same optimal solution in the form of hyperplane parameters, for a given training set. For perceptrons and genetic algorithms, which are widely used for classification in machine learning, solutions are highly dependent on the initial state and terminating criteria. By using the kernel trick, the data can be transformed from the input space to a higher dimensional feature space. For a given training set and kernel, SVM returns unique hyperplane parameters, whereas the perceptron and GA classifiers could obtain different models with different initial state. Since, their aim is to minimize error during training, which results into several candidate hyperplanes that meet this requirement. Out of these many hyperplanes, only the optimal one is retained. The training set represents a sample of population and if it is not the representative of overall population, then the optimal hyperplane might give poor prediction results. This happens because the test data may not exhibit same distribution as the training set and the model leads to poor generalization.

Figure 1 illustrates the optimal hyperplanes obtained with SVM on two-dimensional, two-class dataset. Points labelled with $+$ (red) and $*$ (blue) represent patterns of positive and negative classes respectively. The black dotted line represents the separating hyperplane. SVM does not suffer from the multi-local minima problem which is generally observed in neural networks and is characterised as the curse of dimensionality and overfitting. Several machine learning models strive to achieve a zero error on the training data which leads to a condition called as overfitting. This happens when machine learning approaches involve loss functions that are variants of the sum of squares error, also known as empirical risk minimization. SVM avoids overfitting by minimizing the structural risk instead of empirical risk. In contrast to neural networks, SVM does not explicitly control model complexity by limiting the feature set, rather it is automatically done by selecting the number of support vectors.

3. THE CLASSICAL SVM

For discussion of SVM, let us first start with the introduction of symbols and basic terminology. In a binary classification problem, let the training set has m data patterns, given as

$$X = \{(x_i, y_i), \ \ (i = 1, 2, ..., m)\}. \tag{1}$$

Here, $x_i \ \ (i = 1, 2, ..., m)$ is a row vector in n-dimensional real space $\mathbb{R}^n$. Let matrix X of dimensions $m \times n$, represents all the training patterns. The

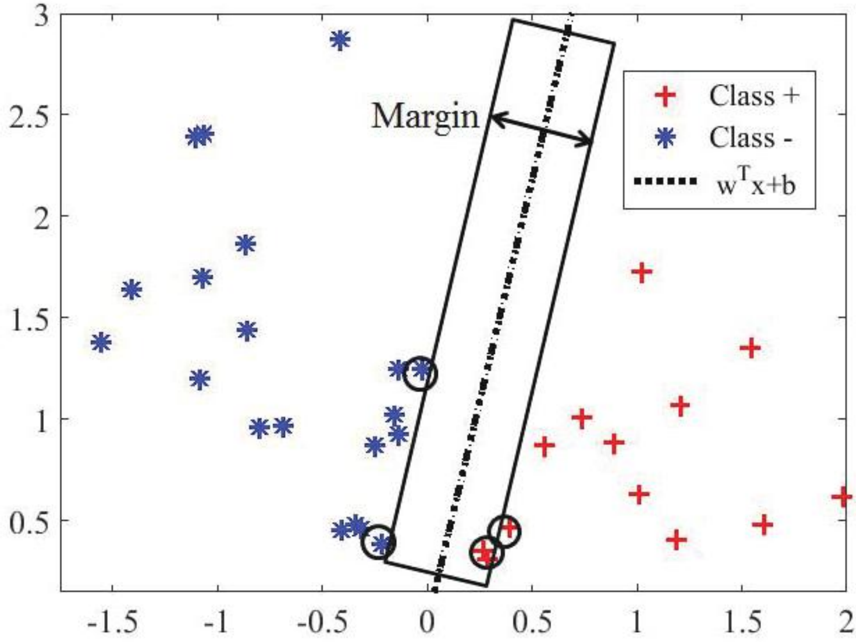


Figure 1. The SVM classifier: Separating hyperplane with two classes. Red $+$ data points are taken as the positive patterns and blue $*$ are the negative patterns. The figure shows margin around the separating hyperplane and the support vectors are encircled.

labels $y_i \in \{+1, -1\}$ for positive and negative classes are given by $+1$ and -1 respectively.

3.1. The Hard-Margin SVM Classifier

Let us first consider the case when the patterns of two classes are linearly separable. The separating hyperplane for SVM is

$$w^T x + b = 0, \tag{2}$$

where $w \in \mathbb{R}^n$ and $b \in \mathbb{R}$. Here, $w,\ b$ represent the parameters of the optimal hyperplane that separates the pattern of the positive $(+1)$ and the negative (-1)

classes. The supporting hyperplanes are given as

$$w^T x + b = 1 \text{ and } w^T x + b = -1. \tag{3}$$

The separating hyperplane (2) lies midway between the two supporting hyperplanes (3). SVM attempts to reduce the generalization error by maximizing the margin between two disjoint bounding planes. The margin is given by

$$\text{margin: } \frac{2}{\|w\|},$$

where w is the normal vector to the optimal hyperplane and $\|.\|$ represents L_2-norm. It is assumed that the two classes lie on either side of the separating hyperplane [6]. The maximum margin SVM classifier can be obtained by following optimization problem

$$\begin{aligned} \max_{w,b} \quad & \frac{2}{\|w\|} \\ \text{subject to} \quad & Aw + eb \geq 1, \\ & Bw + eb \leq -1, \end{aligned} \tag{4}$$

where A and B are the matrices for patterns of positive and negative classes respectively. If there are m_1 patterns in positive class and m_2 patterns in negative class, then dimensions of A and B are $m_1 \times n$ and $m_2 \times n$ respectively, with $m = m_1 + m_2$. Here, e is a vector of ones of appropriate dimensions.

The optimization problem of classical hard margin SVM is written as minimization of a convex quadratic function subject to linear inequality constraints.

$$\begin{aligned} \min_{w,b} \quad & \frac{\|w\|}{2} \\ \text{subject to} \quad & Aw + eb \geq 1, \\ & Bw + eb \leq -1, \end{aligned} \tag{5}$$

Data samples which lie on corresponding hyperplanes, given by (3), are known as support vectors.

3.2. The Soft-Margin SVM Classifier

Working with hard-margin SVM classifier is not possible for practical problems. So, a relaxed version of SVM i.e., Soft-margin classifier is proposed. This allows few samples to be present in the margin or beyond it. This leads to wrong classification of few patterns. To minimize the error due to wrong classifications, an error term or slack variable is introduced in the optimization problem. The soft-margin SVM classification problem is written as

$$\begin{aligned}
\min_{w,b,\xi} \quad & \frac{\|w\|}{2} + ce^T\xi \\
\text{subject to} \quad & Aw + eb \geq 1 - \xi, \\
& Bw + eb \leq -1 + \xi, \\
& \xi \geq 0
\end{aligned} \tag{6}$$

In classical soft-margin SVM formulation, the objective function involves minimization of two terms: inverse of margin between two classes and L_1 norm of slack variable that measures the amount of error due to misclassification. In Eq. (6), ξ measures the amount of violation of constraints due to misclassified points. Here, c is a scalar which maintains a trade-off between the classification error and the classification margin. A higher value of c reduces the classification error, while a smaller one gives emphasis to the classification margin. The above equation, which is written in vector form, can also be written as

$$\begin{aligned}
\min_{w,b,\xi} \quad & \frac{\|w\|}{2} + c\sum_{i=1}^{m}\xi_i \\
\text{subject to} \quad & y_i(w^T x_i + b) \geq 1 - \xi_i, \\
& \xi_i \geq 0,
\end{aligned} \tag{7}$$

where i refers to the i^{th} data point.

The SVM formulation of Eq.(6) uses hinge loss function and can also be written as

$$\min_{w} \quad \frac{\|w\|}{2} + c\sum_{i=1}^{m} max(0, 1 - y_i f(x_i)), \tag{8}$$

where x_i and y_i are defined in the beginning of the section. These optimization problems are known as the primal problems of SVM. In practice, the dual problem of SVM primal is solved to obtain the classifier.

3.3. Dual Formulation of Soft-Margin SVM

The Wolfe dual of primal problem is obtained by solving its Lagrangian function. For this work, we obtain dual of Eq.(7) as

$$\begin{aligned} & \max L \\ \text{subject to} \quad & \partial L = 0, \end{aligned} \tag{9}$$

where the Lagrangian function L is given as

$$\begin{aligned} L(w, b, \xi) \; = \; & \frac{\|w\|}{2} + c\sum_{i=1}^{m} \xi_i \\ & + \sum_{i=1}^{m} \alpha_i[1 - \xi_i - y_i(w^T x_i + b)] - \sum_{j=1}^{m} \beta_j \xi_j, \end{aligned} \tag{10}$$

where $\alpha = (\alpha_1, \alpha_2, ..., \alpha_m)^T$ and $\beta = (\beta_1, \beta_2, ..., \beta_m)^T$ are Lagrange multipliers of dimensions $(m \times 1)$. By writing the Karush-Kuhn-Tucker (KKT) necessary and sufficient optimality conditions [7] for the above equation and solving them, we get the dual optimization problem as

$$\begin{aligned} \max \quad & -\frac{1}{2}\sum_i \sum_j \alpha_i \alpha_j y_i y_j (x_i^T x_j) + \sum \alpha_i \\ \text{subject to} \quad & \alpha_i y_i = 0 \\ & 0 \leq \alpha_i \leq c. \end{aligned} \tag{11}$$

The above problem is solved for α and the classifying hyperplane for dual problem is given as

$$f(x) = \sum_{i=1}^{m} \alpha_i y_i (x_i^T x_i) + b. \tag{12}$$

Using Representer's Theorem [8], the solution for w can be obtained as a linear combination of the training data

$$w = \sum_{i=1}^{m} \alpha_i y_i x_i. \tag{13}$$

Here, many of α_is are zero. The ones that are non-zero define the support vectors x_i.

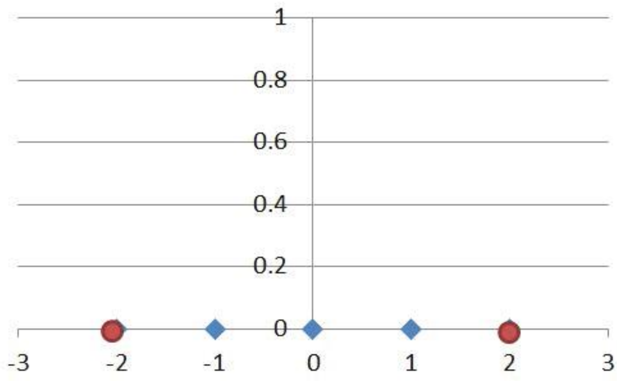


Figure 2. Data points from two classes (Blue diamond and red circles): Linearly inseparable data.

4. Nonlinear Kernel SVM Classifier

For linearly inseparable data, SVM will not be able to efficiently classify the data points. Let us take an example where data belonging to positive and negative classes is given as $\{-1,\ 0,\ +1\}$ and $\{-2,\ +2\}$ respectively, as shown in Figure 2. For such data, there cannot be a linear classifier that can separate the data points belonging to two classes. This data can be made linearly separable by mapping it to a high-dimensional space. Let us use a mapping function, $f(x) = x^2$, or the mapping function is given as $\phi : \mathbb{R} \rightarrow \mathbb{R}^2$, where $\mathbb{R}$ is the real space. Data points for the two classes after mapping are $\{(-1,1),\ (0,0),\ (+1,1)\}$ and $\{(-2,4),\ (+2,4)\}$. Due to mapping of one-dimensional data points in higher (two) dimension, as shown in Figure 3, the points become linearly separable. Now, a linear classifier can separate the two classes, shown in Figure 4. In other words, we can say that by mapping data into a higher dimension, the problem can be solved by a linear classifier.

To generate an SVM classifier in a transformed feature space, a mapping function $\phi : x \rightarrow \phi(x)$ is required, which can transform data from $\mathbb{R}^d \rightarrow \mathbb{R}^D$. Here, $d << D$. However, the nonlinear classifier can be learnt without explicitly computing $\phi(x)$. We use kernel trick and find kernel $K(x, z)$ which could give transformed data like $(x^T z)$. Here, K is the kernel. This trick transforms the data to a higher dimension.

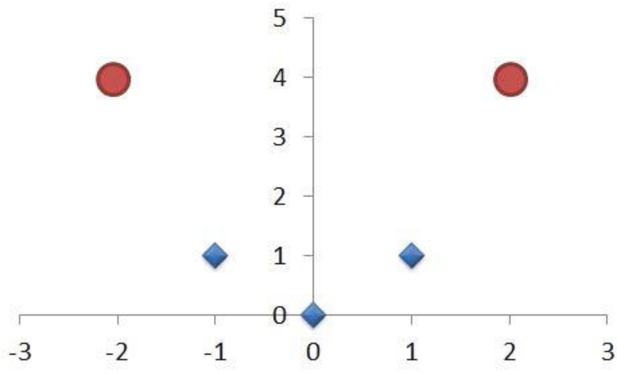


Figure 3. Data in two-dimensional space.

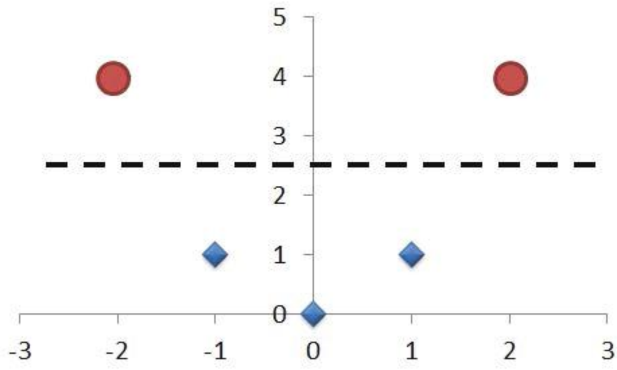


Figure 4. Data in two-dimensional space with linear classifier.

Kernels $K(x_1, x_2) =< \phi(x_1), \phi(x_2) >$ are functions that return inner products between he images of data points in some space. By replacing inner products with kernels in linear algorithms, we obtain very flexible representations. Kernel functions can often be computed efficiently even for very high dimensional spaces. Few popular kernels are listed below:

- Linear Kernels

$$K(x, x') = x^T x'$$

- Polynomial Kernels

$$K(x, x') = (1 + x^T x')^d, \text{ for any } d > 0$$

- Gaussian (Radial basis function) Kernels

$$K(x, x') = exp(-\|x - x'\|^2/2\sigma^2), \text{ for } \sigma > 0.$$

CONCLUSION

This chapter presents the geometric interpretation and optimization problem for a conventional Support Vector Machine (SVM). Two implementations of SVM are discussed as hard and soft margin classifiers, along with the introduction of kernel SVM.

REFERENCES

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

[2] Smola Alex and Vapnik Vladimir, Support vector regression machines. *Advances in Neural Information Processing Systems*, 9:155–161, 1997.

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

[4] Burges D. and Crisp C., A geometric interpretation of nu-svm classifiers. *Advances in Neural Information Processing Systems*, 12(12):244, 2000.

[5] Gunn Steve R. et al. Support vector machines for classification and regression. *ISIS technical report*, 14, 1998.

[6] Vapnik Vladimir N., An overview of statistical learning theory. *IEEE Transactions on Neural Networks,*, 10(5):988–999, 1999.

[7] Mangasarian Olvi L., *Nonlinear programming*, volume 10. SIAM, 1993.

[8] Vapnik Vladimir, *The nature of statistical learning theory*. Springer science & business media, 2013.

In: Support-Vector Machines
Editor: Pooja Saigal
ISBN: 978-1-53618-757-1

Chapter 2

JOURNEY OF SUPPORT VECTOR MACHINES: FROM MAXIMUM-MARGIN HYPERPLANE TO A PAIR OF NON-PARALLEL HYPERPLANES

Pooja Saigal, PhD*
Vivekananda School of Information Technology,
Vivekananda Institute of Professional Studies, New Delhi, India

Abstract

Support Vector Machine (SVM) has proved to be a very powerful machine learning tool for binary classification. SVM is based on the theory of statistical learning and its optimization problem follows Structural Risk Minimization (SRM) principle. The idea of SVM is to obtain maximum margin hyperplane, which can separate the data points of two classes. For this, SVM formulates a convex quadratic minimization function, subject to linear inequality constraints. The hyperplane is obtained by solving its dual problem. For a binary classification problem with m data points, the learning time complexity of SVM is $O(m^3)$. SVM has evolved in the last two decades and many variants of SVM have been proposed, which are more efficient than SVM in terms of generalization ability and training-testing time. In order to improve the training time of the classifier, least-squares version of SVM (LS-SVM) was proposed. A breakthrough development in this area is Twin support vector machine (TWSVM) which proved to be more robust and almost four times faster

*Corresponding Author's Email: pooja.saigal@vips.edu, saigal.pooja.in@gmail.com.

than SVM. TWSVM gave a new dimension to SVM by introducing the concept of pair of non-parallel proximal hyperplanes. In this chapter, few variations of SVM have been discussed which are recently proposed. They are compared based on their strengths and time complexities.

Keywords: classifiers, twin support vector machines, non-parallel hyperplanes, efficiency, generalization ability, training-testing time

1. INTRODUCTION

Support Vector Machines (SVMs) [1] are one of the most distinguished works in supervised learning. Its popularity has grown in the last two decades. SVM [2, 3] has its foundation in statistical learning theory and is based on Structural Risk Minimization (SRM) principle. SVM attempts to reduce the generalization error by maximizing the margin between two disjoint half planes. The separating hyperplane is obtained as mean of these parallel half planes. The optimization problem involves minimization of a convex quadratic function subject to linear inequality constraints. In classical SVM, the formulation involves minimization of inverse of margin between two classes and L_1 norm of slack variable that measures the amount of error. For more details on SVM optimization problem, kindly refer to Chapter 1. SVM has been proved to be an effective classifier. By solving a convex optimization problem, SVM is able to give global optimal solution. Figure 1 shows the classical SVM classifier. In the last few years, many variations of SVM have been proposed.

Inspired by the success of SVM, many researchers have been trying to bring variations to the classical SVM model. Few classifiers are proposed as parallel hyperplane classifiers, where a pair of hyperplanes are obtained which are parallel to each other. Later another category of classifiers were developed that obtain a pair of non-parallel hyperplanes. In this chapter, we are going to discuss some remarkable variants of SVM which have been proposed in the last two decades.

Symbols and Notations

For a binary classification problem, let the training set has m data patterns, given as $\{(x_i, y_i),\ \ (i = 1, 2, ..., m)\}$. Here, $x_i\ \ (i = 1, 2, ..., m)$ is a row vector in n-dimensional real space $\mathbb{R}^n$. Let matrix X of dimensions $m \times n$, represents all the training patterns. The labels $y_i \in \{+1, -1\}$ for positive and

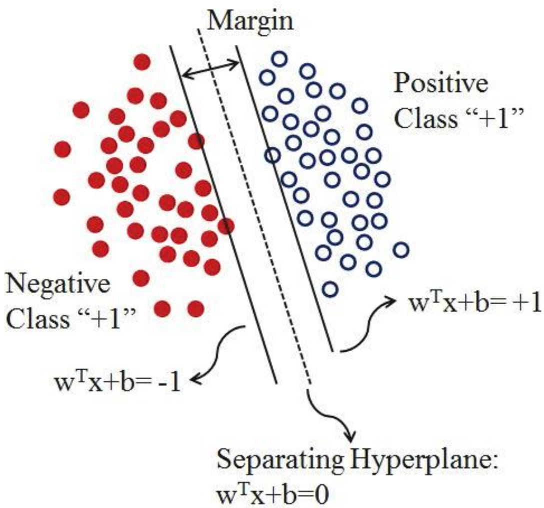


Figure 1. Support Vector Machine Classifier: Hollow circles represent the positive class and solid circles represent negative class. The separating hyperplane is shown between two disjoint half planes.

negative classes are given by $+1$ and -1 respectively. When working with a pair of optimization problems, the patterns belonging to positive and negative classes will be represented by matrices A and B respectively. The number of patterns in these classes be given by m_1 and m_2 $(m = m_1 + m_2)$; therefore, the order of matrices A and B are $(m_1 \times n)$ and $(m_2 \times n)$ respectively. Here, n is the dimension of feature space and A_i $(i = 1, 2, ..., m_1)$ is a row vector in n-dimensional real space $\mathbb{R}^n$, that represents feature vector of a data sample. In this chapter, 'positive class' and 'Class $+1$' are used interchangeably; similarly 'negative class' and 'Class -1' would refer to set of negative patterns.

2. Least-Squares Support Vector Machine

To improve the time complexity of SVM classifier, Sukyens and Vandewalle [4] proposed Least-squares Support Vector Machine (LS-SVM) classifier in 1999. The major difference between SVM and LS-SVM is the way in which the empirical error is measured. LS-SVM proposed the use of L_2-norm, i.e. the squared or Euclidean norm, to measure the classification error. This brought a significant change in the formulation of the optimization problem and greatly improved the learning time of the classifier. The inequality constraints of classical SVM are replaced by the equality constraints for least-squares formulation. So, the solution of LS-SVM optimization problem can be directly obtained by solving the primal problem. Another advantage of least-squares formulation is that it avoids solving the expensive quadratic programming problems (QPPs) as solved by SVM. Instead of this, LS-SVM solves a system of linear equations to generate the separating hyperplane.

The LS-SVM obtains the maximum-margin hyperplane of the form

$$w^T x + b = 0. \tag{1}$$

Here, Eq.(1) represents the separating hyperplane, with $w \in \mathbb{R}^n$ and $b \in \mathbb{R}$. The supporting hyperplanes are given as $w^T x + b = 1$ and $w^T x + b = -1$. The hyperplane is obtained by solving the following optimization problem

$$\begin{aligned} \min_{w,b,\xi} \quad & \frac{1}{2} w^T w + \frac{C}{2} \xi^2, \\ \text{subject to} \quad & Y[w^T X + b] = e - \xi, \end{aligned} \tag{2}$$

where, X represents the data matrix that contains all the data points. The dimensions of X are $(m \times n)$, here m is the number of training points and n is the feature dimension in real space $\mathbb{R}^n$. The first term of the objective function is the reciprocal of the distance between the two proximal hyperplanes, and minimizing it attempts to find two hyperplanes that are as far as possible. At the same time, these hyperplanes pass through the samples of their respective classes. The slack vector ξ captures the amount of violations of constraints and C is the penalty parameter. $Y \in \{+1, \ -1\}$ is the vector of class labels and e is a vector of ones of appropriate dimension.

Instead of solving the dual problem, as done in SVM, the solution of LS-SVM is obtained by solving the following system of linear equations:

$$\begin{bmatrix} I & 0 & 0 & -Z^T \\ 0 & 0 & 0 & -Y^T \\ 0 & 0 & c_1 I & -I \\ Z & Y & I & 0 \end{bmatrix} \begin{bmatrix} w \\ b \\ \xi \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ e \end{bmatrix} \tag{3}$$

where, Z=$[X^TY_1; ...; X^TY_N]$, $Y = [Y_1; ...Y_N]$, $e = [1; ...1]$, $\xi = [\xi_1; ...; \xi_N]$, $\alpha = [\alpha_1; ...; \alpha_N]$, and the equation (3) can be further solved as:

$$\begin{bmatrix} 0 & -Y^T \\ Y & ZZ^T + c_1^{-1}I \end{bmatrix} \begin{bmatrix} b \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ e. \end{bmatrix} \tag{4}$$

LS-SVM is proved to have better generalization ability and lower computational cost than the traditional SVM. For more details about LS-SVM, kindly refer to the original text [4]. To classify linearly inseparable datasets, we can make use of kernel trick, as discussed for SVM.

3. Proximal Support Vector Machine

In 2001, Mangasarian and Wild proposed Proximal Support Vector Machine (PSVM) [5], which is another variation of SVM. Similar to LS-SVM, PSVM uses L_2-norm of error or slack variable. The two classifiers i.e. PSVM and LS-SVM, differ slightly in their problem formulation. The optimization problem for PSVM is given as:

$$\begin{aligned} \min_{w,b,\xi} \quad & \frac{1}{2}(w^T w + b^2) + \frac{C}{2}\xi^2, \\ \text{subject to} \quad & Y[w^T X + b] = e - \xi. \end{aligned} \tag{5}$$

PSVM has the optimization problem similar to LS-SVM, with an additional b^2 term in the objective function. This makes the problem strictly convex and simplifies its algebraic solution.

The solutions of both LS-SVM and PSVM aim at finding $w^T x + b = \pm 1$, which represent a pair of parallel hyperplanes. Here, $w^T x + b = +1$ is the hyperplane which is proximal to class A (positive class), while $w^T x + b = -1$ is the hyperplane for class B. Each hyperplane is the representative of its own class. The hyperplanes are called as ‘proximal’ as they are close to patterns of their own class and are at maximum distance from the patterns of other class. Figure.2 shows the pair of proximal hyperplanes generated by PSVM.

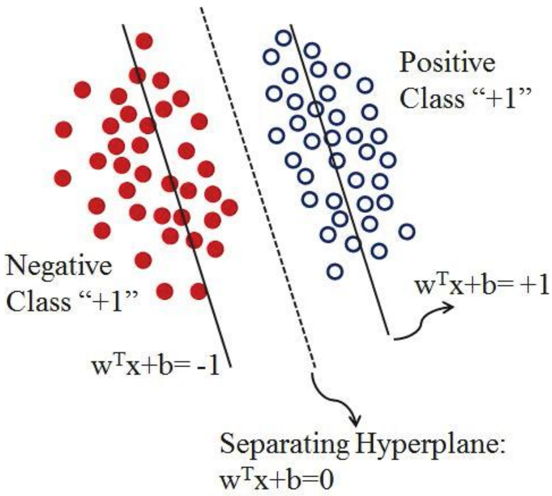


Figure 2. Proximal Support Vector Machine Classifier: Hollow circles represent the positive class and solid circles represent negative class. The pair of proximal hyperplanes are shown for two classes.

4. Generalized Eigenvalue Proximal Support Vector Machine

The first non-parallel hyperplane classifier is proposed by Mangasarian and Wild [6] in 2006 and is termed as Generalized Eigenvalue Proximal Support Vector Machine (GEPSVM). This classifier generates a pair of non-parallel hyperplanes which are proximal to their own class. The two hyperplanes are obtained by the eigenvectors for smallest eigenvalues of two generalized eigenvalue problems (GEPs) [7]. The formulation for GEPSVM differs from that of SVM; since GEPSVM generates two hyperplanes that approximately pass through their classes, while SVM generates one separating hyperplane. Also, SVM solves a quadratic programming problem (QPP), whereas GEPSVM solves two GEPs. Therefore, GEPSVM is faster than SVM.

GEPSVM [8] generates two non-parallel hyperplanes by solving two GEPs of the form $Pz = \mu Qz$, where P and Q are symmetric positive semi-definite matrices. The eigenvector corresponding to the smallest eigenvalue of each GEP determines the hyperplane. The data points of two classes (referred as positive and negative classes) are given by matrices A and B respectively, with m_1 and m_2 data points. Therefore, matrices A and B have dimensions $(m_1 \times n)$ and $(m_2 \times n)$. The GEPSVM formulation determines two non-parallel planes

$$x^T w_1 + b_1 = 0 \text{ and } x^T w_2 + b_2 = 0, \tag{6}$$

so as to minimize the distance of these hyperplanes from the patterns of positive and negative classes respectively. The optimization problem is given as:

$$\underset{w,b\neq 0}{Min} \ \frac{\|Aw + eb\|^2 / \|[w, b]^T\|^2}{\|Bw + eb\|^2 / \|[w, b]^T\|^2}, \tag{7}$$

where e is a vector of ones of appropriate dimension and $\|\cdot\|$ represents the L_2 norm. Here, it is assumed that $(w, b) \neq 0 \Rightarrow Bw + eb \neq 0$ [6]. The objective function (7) is simplified and regularized by adding a term as proposed by Tikhonov [9]:

$$\underset{w,b\neq 0}{Min} \ \frac{(\|Aw + eb\|^2 + \delta \|[w, b]^T\|^2)}{\|Bw + eb\|^2}, \tag{8}$$

where $\delta > 0$ is the regularization parameter. This, in turn, takes the form of Rayleigh Quotient [10]

$$\underset{w,b\neq 0}{Min} \ \frac{z^T P z}{z^T Q z}, \tag{9}$$

where P and Q are symmetric matrices in $R^{(n+1)\times(n+1)}$ and are given as

$$\begin{aligned} P &= [A \ \ e]^T \times [A \ \ e] + \delta \times I \text{ for some } \delta > 0, \\ Q &= [B \ \ e]^T \times [B \ \ e], \text{ and } z = [w, b]^T. \end{aligned} \tag{10}$$

I is an identity matrix. Using the properties of the Rayleigh Quotient [6],[10], we can get the solution of (9) by solving the following GEP

$$Pz = \mu Q z, \ \ z \neq 0, \tag{11}$$

where the solution of (9) is attained at an eigenvector corresponding to the smallest eigenvalue μ_{min} of (11). Therefore, if z_1 is the eigenvector for μ_{min}, then $[w_1, b_1]^T = z_1$ is the plane $x^T w_1 + b_1 = 0$ that is passing through the positive class. The other minimization problem can be similarly defined by switching the roles of A and B. The eigenvector z_2 for the smallest eigenvalue of second GEP yields the hyperplane $x^T w_2 + b_2 = 0$, which is closer to negative class.

Comparison of Time Complexity

The solution of GEPSVM is generated by a system of linear equations of order n^3 [6], where n is the feature dimension and SVM solves a QPP of order m^3 ($m >>> n$). Therefore, GEPSVM is computationally more efficient than SVM, although the classification accuracy results are comparable to those of SVM.

In the past few years, various modifications for GEPSVM have been proposed like Regularized GEPSVM (RegGEPSVM) [11] and Improved GEPSVM (IGEPSVM) [12].

4.1. Regularized GEPSVM

Later in 2007, Guarracino et al. [11] modified the formulation of GEPSVM, so that a single GEP can be used to generate both the hyperplanes and called it as Regularized GEPSVM (RegGEPSVM). The GEP $Pz = \mu Qz$ is transformed as $P^*z = \mu Q^*z$ where

$$P^* = \tau_1 P - \delta_1 Q, \;\; Q^* = \tau_2 Q - \delta_2 P. \tag{12}$$

The parameters τ_1, τ_2, δ_1 and δ_2 are selected as a singular matrix Ω, given by

$$\Omega = \begin{bmatrix} \tau_2 & \delta_1 \\ \delta_2 & \tau_1 \end{bmatrix}. \tag{13}$$

As discussed in [11], the problem $P^*z = \mu Q^*z$ would generate same eigenvectors as that of $Pz = \mu Qz$. The eigenvalue λ^* of new problem is related to an eigenvalue λ of initial problem by

$$\lambda = \frac{\tau_2 \lambda^* + \delta_1}{\tau_1 + \delta_2 \lambda^*}. \tag{14}$$

By setting $\tau_1 = \tau_2 = 1$ and $\nu_1 = -\delta_1, \nu_2 = -\delta_2$, the problem is stated as

$$\underset{w,b \neq 0}{Min} \;\; \frac{||Aw + eb||^2 + \nu_1 ||Bw + eb||^2}{||Bw + eb||^2 + \nu_2 ||Aw + eb||^2}. \tag{15}$$

When Ω is singular and $\nu_1, \; \nu_2$ are non-negative, then the eigenvectors corresponding to the minimum and maximum eigenvalues of (15) would be same as obtained by solving the two GEPSVM problems [11].

In terms of learning time, RegGEPSVM outperforms GEPSVM and SVM as RegGEPSVM [11] solves one GEP instead of two. In [11], authors have shown the out-performance of RegGEPSVM over SVM for linear kernel, but with gaussian kernel, SVM achieves better performance than RegGEPSVM and GEPSVM.

4.2. Improved GEPSVM

Improved GEPSVM (IGEPSVM) [12], proposed by Shao et al. (2013), replaces the generalized eigenvalue decomposition by standard eigenvalue problems, which resulted in solving two optimization problems that are simpler than GEP. A parameter is introduced to the objective function to improve the generalization ability. IGEPSVM formulated the two problems as

$$\underset{w,b\neq 0}{Min}\ \frac{\|Aw+eb\|^2}{\|w\|^2+b^2} - \nu\frac{\|Bw+eb\|^2}{\|w\|^2+b^2}, \tag{16}$$

where $\nu > 0$ arbitrates the terms in the objective functions. Thus, IGEPSVM has a bias factor that can be adjusted by the user and is particularly useful when working with imbalanced data. By introducing a Tikhonov regularization term [9] and solving its Lagrange function [13], we get

$$((M^T + \delta I) - \nu N^T)z = \lambda z, \tag{17}$$

where $M = [A\ \ e]^T[A\ \ e],\ N = [B\ \ e]^T[B\ \ e]$, $z = [w, b]^T$ and λ is Lagrange multiplier. I is an identity matrix of proper size. The second problem can be defined similar to (16) by switching the roles of A and B, as discussed for GEPSVM.

IGEPSVM replaces GEP with a standard eigenvalue problem and hence, results in a lighter optimization problem [12]. It also avoids the possible singularity in GEPSVM by adding a regularization term.

5. Twin Support Vector Machine

In 2007, a major breakthrough was achieved in the field of supervised learning with the development of Twin Support Vector Machine (TWSVM) [7]. It has better generalization ability than SVM and is almost four times faster than the classical SVM. On the lines of GEPSVM, Jayadeva et al. proposed TWSVM

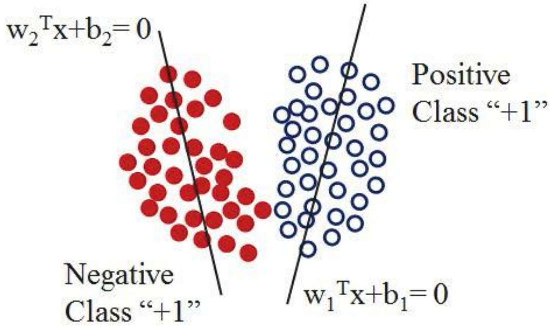


Figure 3. Twin Support Vector Machine Classifier: Hollow circles represent the positive class and solid circles represent negative class. The non-parallel proximal hyperplanes are shown for two classes.

which classifies data by generating two non-parallel hyperplanes. TWSVM solves two QPPs of smaller size and generates two non-parallel hyperplanes such that each hyperplane is proximal to its own class and unit distance away from the other class.

TWSVM [14] is a supervised learning tool that classifies data by generating two non-parallel hyperplanes which are proximal to their respective classes and at least unit distance away from the patterns of other class. TWSVM solves a pair of quadratic programming problems (QPPs) and is based on empirical risk minimization (ERM) principle [15]. The binary classifier TWSVM [7, 16] determines two non-parallel hyperplanes by solving two related SVM-type problems, each of which has fewer constraints than those in a conventional SVM. The hyperplanes are given by Eq.(6), where $w_1,\ b_1,\ w_2,\ b_2$ are the parameters of normals to the two hyperplanes, referred as positive and negative hyperplanes. The proximal hyperplanes are obtained by solving the following pair of QPPs.

$$\begin{aligned} \min_{w_1,b_1,\xi_2} \quad & \frac{1}{2}\|Aw_1+e_1b_1\|_2^2+c_1e_2^T\xi_2 \\ \text{subject to} \quad & -(Bw_1+e_2b_1)+\xi_2 \geq e_2, \quad \xi_2 \geq 0, \end{aligned} \tag{18}$$

and

$$\begin{array}{ll}\min\limits_{w_2,b_2,\xi_1} & \frac{1}{2}\|Bw_2+e_2b_2\|_2^2+c_2e_1^T\xi_1 \\ \text{subject to} & (Aw_2+e_1b_2)+\xi_1 \geq e_1, \quad \xi_1 \geq 0.\end{array} \tag{19}$$

Here, c_1 $(or\ c_2) > 0$ is a trade-off factor between error vector ξ_2 (or ξ_1) due to misclassified negative (or positive) class patterns and distance of hyperplane from positive (or negative) class; e_1, e_2 are vectors of ones of appropriate dimensions and $\|.\|$ represents L_2 norm. The first term in the objective function of (18) or (19) is the sum of squared distances of the hyperplane to the data patterns of its own class. Thus, minimizing this term tends to keep the hyperplane closer to the patterns of one class and the constraints require the hyperplane to be at least unit distance away from the patterns of other class. Since this constraint of unit distance separability cannot be always satisfied; so, TWSVM is formulated as a soft-margin classifier and a certain amount of error is allowed. If the hyperplane is less than unit distance away from data patterns of other class, then the error variables ξ_1 and ξ_2 measure the amount of violation. The objective function minimizes L_1-norm of error variables to reduce misclassification. The solution of the problems (18) and (19) can be obtained indirectly by solving their Lagrangian functions and using Karush-Kuhn-Tucker (KKT) conditions [13]. The Wolfe dual of TWSVM primal problems are as follows:

$$\begin{array}{ll}\max\limits_{\alpha} & e_2^T\alpha - \frac{1}{2}\alpha^T G(H^TH)^{-1}G^T\alpha \\ \text{subject to} & 0 \leq \alpha \leq c_1,\end{array} \tag{20}$$

and

$$\begin{array}{ll}\max\limits_{\beta} & e_1^T\beta - \frac{1}{2}\gamma^T P(Q^TQ)^{-1}P^T\beta \\ \text{subject to} & 0 \leq \beta \leq c_2.\end{array} \tag{21}$$

Here, $H = [A\ \ e_1]$, $G = [B\ \ e_2]$, $P = [A\ \ e_1]$, $Q = [B\ \ e_2]$ are augmented matrices of respective classes. The augmented vectors $u_1 = [w_1, b_1]^T$ and $u_2 = [w_2, b_2]^T$ are given by

$$u_1 = -(H^TH)^{-1}G^T\alpha, \tag{22}$$

$$u_2 = (Q^TQ)^{-1}P^T\gamma, \tag{23}$$

where $\alpha = (\alpha_1, \alpha_2, ..., \alpha_{m2})^T$ and $\gamma = (\gamma_1, \gamma_2, ..., \gamma_{m_1})^T$ are Lagrange multipliers.

Testing a New Pattern

As we obtain the solutions (w_1, b_1) and (w_2, b_2) of the problems (18) and (19) respectively, a new data sample $x \in \mathbb{R}^n$ is assigned to class r $(r = 1, 2)$, depending on which of the two planes given by (6), it lies closer to i.e.

$$r = arg\,(\min_{l=1,2} \frac{|x^T w_l + b_l|}{\|w_l\|_2}), \tag{24}$$

where $|.|$ is the perpendicular distance of point x from the plane $x^T w_l + b_l = 0,\ l = 1, 2$. The label assigned to the test data is given as $y = \begin{cases} +1 & (r = 1) \\ -1 & (r = 2) \end{cases}$.

Comparison of Time Complexity

The complexity of SVM problem is of the order m^3, where m is the total number of patterns appearing in the constraints and TWSVM solves two problems (18) and (19), each of which has approximately $(m/2)$ constraints. Therefore, the ratio of learning-time of SVM and TWSVM is approximately $[(m^3)/(2 \times (m/2)^3)] = 4$; this makes TWSVM almost four times faster than SVM [7].

Kernel TWSVM

TWSVM has been extended to handle linearly inseparable data by considering two kernel generated surfaces, given as:

$$Ker(x^T, C^T)z_1 + b_1 = 0, \tag{25}$$

$$Ker(x^T, C^T)z_2 + b_2 = 0, \tag{26}$$

where $C^T = [A\ ;\ B]^T$ is the augmented data matrix and Ker is an appropriately chosen kernel. The primal QPP of non-linear TWSVM corresponding to the surface (25) is given by

$$\begin{aligned} \min_{z_1, b_1, \xi_2} \quad & \frac{1}{2}\|Ker(A, C^T)z_1 + e_1 b_1\|_2^2 + c_1 e_2^T \xi_2 \\ \text{subject to} \quad & -(Ker(B, C^T)z_1 + e_2 b_1) + \xi_2 \geq e_2, \quad \xi_2 \geq 0. \end{aligned} \tag{27}$$

The second problem of non-linear TWSVM can be defined in similar manner as (27) and their solution is obtained from the dual problems, as done for linear case [7].

In the last decade, TWSVM has attracted many researchers and a lot of work has been done based on TWSVM. It is beyond the scope of this chapter to discuss all of them. But few variants of TWSVM are briefly discussed in the following section.

5.1. Least Square Twin Support Vector Machine

Least Square Twin Support Vector Machine (LS-TWSVM) [17] is motivated by TWSVM and solves a pair of QPPs on the lines of LS-SVM [4]. Proposed by Kumar et al. in 2009, LS-TWSVM modifies the primal problems of TWSVM and solves them directly instead of finding the dual problems. Further, the solution of primal problems is reduced to solving two systems of linear equations instead of solving two expensive QPPs, along with two systems of linear equations, as required in TWSVM. The primal problems of LS-TWSVM deal with equality constraints and are given as follows:

$$\begin{aligned} \min_{w_1, b_1, \xi_2} \quad & \frac{1}{2}\|Aw_1 + e_1 b_1\|^2 + \frac{c_1}{2}\xi_2^T \xi_2 \\ \text{subject to} \quad & -(Bw_1 + e_2 b_1) + \xi_2 = e_2. \end{aligned} \tag{28}$$

and

$$\begin{aligned} \min_{w_2, b_2, \xi_1} \quad & \frac{1}{2}\|Bw_2 + e_2 b_2\|^2 + \frac{c_2}{2}\xi_1^T \xi_1 \\ \text{subject to} \quad & (Aw_2 + e_1 b_2) + \xi_1 = e_1. \end{aligned} \tag{29}$$

The QPPs (28), (29) use L_2-norm of error variables ξ_1, ξ_2 with weights $c1$, $c2$; whereas TWSVM uses L_1 norm of error variables. This makes the constraint $\xi_2 \geq 0$ and $\xi_1 \geq 0$ of (18) and (19) respectively, redundant.

Time Complexity

Linear LS-TWSVM obtains the classifier with two matrix inverse operations, each of order $(n+1) \times (n+1)$, where $n << m$. LS-TWSVM has been extended to non-linear kernel by considering the kernel generated surfaces [17].

5.2. Twin Bounded Support Vector Machine

Similar to TWSVM, Twin Bounded Support Vector Machine (TBSVM) [18] also constructs two non-parallel hyperplanes, as given in (6), by solving two QPPs. However, TBSVM (proposed in 2011) distinguishes itself from TWSVM by adding a regularization term, in the primal problems of TWSVM, with the idea of maximizing the margin [18]. TWSVM takes care of the empirical risk whereas TBSVM minimizes both the empirical as well as structural risk. TBSVM considers the following primal problems:

$$\begin{aligned} \min_{w_1,b_1,\xi_2} \quad & \frac{1}{2}\|Aw_1 + e_1b_1\|_2^2 + c_1e_2^T\xi_2 + \frac{1}{2}c_3(\|w_1\|_2^2 + b_1^2) \\ \text{subject to} \quad & -(Bw_1 + e_2b_1) + \xi_2 \geq e_2, \quad \xi_2 \geq 0, \end{aligned} \tag{30}$$

and

$$\begin{aligned} \min_{w_2,b_2,\xi_1} \quad & \frac{1}{2}\|Bw_2 + e_2b_2\|_2^2 + c_2e_1^T\xi_1 + \frac{1}{2}c_4(\|w_2\|_2^2 + b_2^2) \\ \text{subject to} \quad & (Aw_2 + e_1b_2) + \xi_1 \geq e_1, \quad \xi_1 \geq 0. \end{aligned} \tag{31}$$

The constants $c_1,\ c_2,\ c_3$ and c_4 are positive parameters which associate weights with the corresponding terms. The TBSVM QPPs are solved in similar manner as TWSVM and can be extended to non-linear kernel version.

5.3. Twin Parametric-Margin Support Vector Machine

In 2011, Peng et al. proposed Twin Parametric-Margin Support Vector Machine (TPMSVM) as a binary classifier that determines two non-parallel parametric-margin hyperplanes by solving two related SVM-type problems [19], each of which is smaller than a conventional SVM [3] or Parametric ν-Support Vector Machine (par-ν-SVM) [20] problem. TPMSVM separates the data of the two classes if and only if

$$\begin{aligned} A_iw_1 + b_1 \geq 0, \quad & \text{for} \quad A_i \in A, \\ B_iw_2 + b_2 \leq 0, \quad & \text{for} \quad B_i \in B, \end{aligned} \tag{32}$$

where A_i and B_i represent i^{th} data sample of their respective classes. The primal formulation for the pair of QPPs in TPMSVM is given as follows:

$$\begin{aligned}
\min_{w_1,b_1,\xi_1} \quad & \frac{1}{2}\|w_1\|^2 + \frac{c_1}{m_2}e_2^T(Bw_1 + e_2b_1) + \frac{c_2}{m_1}e_1^T\xi_1 \\
\text{subject to} \quad & Aw_1 + e_1b_1 \geq 0 - \xi_1, \quad \xi_1 \geq 0,
\end{aligned} \tag{33}$$

and

$$\begin{aligned}
\min_{w_2,b_2,\xi_2} \quad & \frac{1}{2}\|w_2\|^2 - \frac{c_3}{m_1}e_1^T(Aw_2 + e_1b_2) + \frac{c_4}{m_2}e_2^T\xi_2 \\
\text{subject to} \quad & Bw_2 + e_2b_2 \leq 0 + \xi_2, \quad \xi_2 \geq 0.
\end{aligned} \tag{34}$$

The constants $c_1, c_2, c_3, c_4 > 0$ are trade off factors; e_1, e_2 are vectors of ones, in real space, of appropriate dimensions and $\|.\|_2$ represents L_2-norm. The first term of the objective function of (33) and (34) controls the complexity of the model. The second term of (33) minimizes the sum of projection values of negative class training patterns on the hyperplane of positive class, with parameter c_1. The objective function also minimizes the sum of error, which occurs due to the data patterns lying on wrong sides of the hyperplanes. The constraints of (33) require that the projection values of positive training patterns on the positive hyperplane should be at least zero. A slack vector ξ_1 measures the amount of error due to positive training points. The optimization problem of (34) can be defined analogously.

5.4. ν-Twin Support Vector Machine

X. Peng [21] proposed a modification to TWSVM, termed as ν-Twin Support Vector Machine (ν-TWSVM) and introduced two parameters ν_1 and ν_2 instead of the trade-off parameters c_1 and c_2 of TWSVM. The parameters ν_1, ν_2 in the ν-TWSVM control the bounds on number of support vectors and the margin errors. The primal optimization problems of ν-TWSVM are as follows:

$$\begin{aligned}
\min_{w_1,b_1,\rho_1,\xi_2} \quad & \frac{1}{2}\|Aw_1 + e_1b_1\|_2^2 - \nu_1\rho_1 + \frac{1}{m_2}e_2^T\xi_2 \\
\text{subject to} \quad & -(Bw_1 + e_2b_1) + \xi_2 \geq e_2\rho_1, \\
& \xi_2 \geq 0, \quad \rho_1 \geq 0,
\end{aligned} \tag{35}$$

and

$$\begin{aligned} \min_{w_2,b_2,\rho_2,\xi_1} \quad & \frac{1}{2}\|Bw_12+e_2b_2\|_2^2-\nu_2\rho_2+\frac{1}{m_1}e_1^T\xi_1 \\ \text{subject to} \quad & (Aw_2+e_1b_2)+\xi_1 \geq e_1\rho_2, \\ & \xi_1 \geq 0, \quad \rho_2 \geq 0. \end{aligned} \tag{36}$$

Here, ρ_i $(i = 1, 2)$ measure the minimum separating distance between the patterns of one class and hyperplane of other class and are optimized in (35) and (36). Both the optimization problems try to maximize this distance. The role of ρ_i is to separate the data patterns of one class from the hyperplane of other class by a margin of $\rho_i/(w_i^T w_i)$ where $(i = 1, 2)$ [21]. The parameter ν_2 (or ν_1) determines an upper bound on the fraction of positive class (or negative class) margin errors and a lower bound on the fraction of positive class (or negative class) support vectors [21].

5.5. Non-Parallel Support Vector Machine with One Optimization Problem

In 2015, Tian and Ju [22] proposed a binary classifier Non-parallel Support Vector Machine with One Optimization Problem (NSVMOOP), that determines the two non-parallel proximal hyperplanes by solving a single optimization problem. NSVMOOP aims at maximizing the angle between the normal vectors of the two hyperplanes. NSVMOOP combines the two QPPs of TWSVM together and formulates a single QPP which is given as

$$\begin{aligned} \min_{w_\pm,b_\pm,\eta_\pm,\xi_\pm} \quad & \frac{1}{2}(\|w_1\|_2^2+\|w_2\|_2^2) \\ & +c_1(\eta_1^T\eta_1+\eta_2^T\eta_2+e_1^T\xi_1+e_2^T\xi_2)+c_2(w_1.w_2), \\ \text{subject to} \quad & Aw_1+e_1b_1=\eta_1, \\ & Bw_2+e_2b_2=\eta_2, \\ & -(Bw_1+e_2b_1)+\xi_2 \geq e_2, \quad \xi_2 \geq 0 \\ & (Aw_2+e_1b_2)+\xi_1 \geq e_1, \quad \xi_1 \geq 0, \end{aligned} \tag{37}$$

where c_1 and c_2 are positive trade-off parameters. The first set of terms in the objective function of (37) are the regularization terms. The second set of terms consist of two types of errors. The error terms $\eta_1^T\eta_1$ and $\eta_2^T\eta_2$ are the sum of the squared distances of data patterns from their own hyperplane, and hence their minimization keeps the respective hyperplanes proximal to the patterns of their own class. The other error terms $e_1^T\xi_2$ and $e_1^T\xi_2$ are the sum of errors contributed due to violation of corresponding constraints. The term $w_1.w_2$ in the objective function is the inner product of normal vectors to the hyperplanes and its minimization essentially maximizes the separation between the two classes.

6. Recently Proposed Classifiers

Researchers are actively working to develop new classifiers which could deliver better results than well-established methodologies. This section further briefly discusses some recently proposed non-parallel hyperplane classification algorithms.

Improvements on ν-Twin Support Vector Machine (Iν-TWSVM) [23], proposed in 2016, is a classification algorithm which solves a smaller-sized quadratic programming problem (QPP) and an unconstrained minimization problem (UMP), instead of solving a pair of QPPs as done for TWSVM, to generate two non-parallel proximal hyperplanes. The faster version of Iν-TWSVM, termed as Iν-TWSVM (Fast), modifies the first problem of Iν-TWSVM as minimization of a unimodal function for which line search methods can be used; this further avoids solving the QPP in the first problem. Both these classifiers have good generalization ability.

Two recently proposed classifiers i.e., Angle-based Twin Parametric-Margin Support Vector Machine (ATP-SVM) [24] and Angle-based Twin Support Vector Machine (ATWSVM) [25], aim to maximize the angle between the normal vectors to the two non-parallel hyperplanes so as to generate larger separation between the two classes. ATP-SVM solves only one modified QPP with fewer number of representative patterns. Further, it avoids the explicit computation of inverse of matrices in the dual problem and has efficient learning time. Although only one QPP is being solved in ATP-SVM, it still manages to attain the speed comparable to that of TWSVM, due to efficient selection of representative training points. ATWSVM finds the two hyperplanes by solving a QPP and a UMP.

CONCLUSION

SVM is proved to be a versatile classifier. In the last two decades, a lot of SVM variants are proposed which provide better generalization results. Few such SVM-based classifiers are discussed in this chapter- Twin support Vector Machine, Least-squares SVM, Generalised Eigenvalue Proximal SVM etc.

REFERENCES

[1] Vapnik Vladimir N., An overview of statistical learning theory. *IEEE Transactions on Neural Networks,*, 10(5):988–999, 1999.

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

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

[4] Suykens Johan A.K. and Vandewalle Joos, Least squares support vector machine classifiers. *Neural Processing Letters*, 9(3):293–300, 1999.

[5] Mangasarian Olvi L. and Wild Edward W., Proximal support vector machine classifiers. In *Proceedings KDD-2001: Knowledge Discovery and Data Mining*. Citeseer, 2001.

[6] Mangasarian Olvi L. and Wild Edward W., Multisurface proximal support vector machine classification via generalized eigenvalues. *IEEE Transactions on Pattern Analysis and Machine Intelligence,*, 28(1):69–74, 2006.

[7] Jayadeva, Khemchandani R., and Chandra Suresh, Twin support vector machines for pattern classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(5):905–910, 2007.

[8] Saigal Pooja and Khemchandani Reshma, Nonparallel hyperplane classifiers for multi-category classification. In *Computational Intelligence: Theories, Applications and Future Directions (WCI), 2015 IEEE Workshop on*, pages 1–6. IEEE, 2015.

[9] Nikolaevich Tikhonov Andre and IAkovlevich Arsenin Vasili, *Solutions of ill-posed problems*. Vh Winston, 1977.

[10] Parlett Beresford N., *The symmetric eigenvalue problem*, volume 7. SIAM, 1980.

[11] Guarracino Mario R., Cifarelli Claudio, Seref Onur, and Pardalos Panos M., A classification method based on generalized eigenvalue problems. *Optimisation Methods and Software*, 22(1):73–81, 2007.

[12] Shao Yuan-Hai, Deng Nai-Yang, Chen Wei-Jie, and Wang Zhen, Improved generalized eigenvalue proximal support vector machine. *Signal Processing Letters, IEEE*, 20(3):213–216, 2013.

[13] Mangasarian Olvi L., *Nonlinear programming*, volume 10. SIAM, 1993.

[14] Jayadeva, Khemchandani Reshma, and Chandra Suresh, *Twin Support Vector Machines: Models, Extensions and Applications*, volume 659. Springer, 2016.

[15] Saigal Pooja, *Time efficient variants of twin support vector machine with applications in image processing*. PhD thesis, South Asian University, New Delhi-110021, India, 2017.

[16] Khemchandani Reshma, *Mathematical programming applications in machine learning*. PhD thesis, Indian Institute of Technology Delhi New Delhi-110016, India, 2008.

[17] Kumar M. Arun and Gopal Madan, Least squares twin support vector machines for pattern classification. *Expert Systems with Applications*, 36(4):7535–7543, 2009.

[18] Shao Yuan-Hai, Zhang Chun-Hua, Wang Xiao-Bo, and Deng Nai-Yang, Improvements on twin support vector machines. *IEEE Transactions on Neural Networks,*, 22(6):962–968, 2011.

[19] Peng Xinjun, Tpmsvm: a novel twin parametric-margin support vector machine for pattern recognition. *Pattern Recognition*, 44(10):2678–2692, 2011.

[20] Hao Pei-Yi, New support vector algorithms with parametric insensitive/margin model. *Neural Networks*, 23(1):60–73, 2010.

[21] Peng Xinjun, A ν-twin support vector machine (ν-tsvm) classifier and its geometric algorithms. *Information Sciences*, 180(20):3863–3875, 2010.

[22] Tian Ying-Jie and Ju Xu-Chan, Nonparallel support vector machine based on one optimization problem for pattern recognition. *Journal of the Operations Research Society of China*, 3(4):499–519, 2015.

[23] Khemchandani Reshma, Saigal Pooja, and Chandra Suresh, Improvements on ν-twin support vector machine. *Neural Networks*, 79:97–107, 2016.

[24] Rastogi Reshma, Saigal Pooja, and Chandra Suresh, Angle-based twin parametric-margin support vector machine for pattern classification. *Knowledge-Based Systems*, 139:64–77, 2018.

[25] Khemchandani Reshma, Saigal Pooja, and Chandra Suresh, Angle-based twin support vector machine. *Annals of Operations Research*, 269(1-2):387–417, 2018.

In: Support-Vector Machines
Editor: Pooja Saigal
ISBN: 978-1-53618-757-1

Chapter 3

POWER SPECTRUM ENTROPY-BASED SUPPORT VECTOR MACHINE FOR QUANTITATIVE DIAGNOSIS OF ROTOR VIBRATION PROCESS FAULTS

Cheng-Wei Fei*
Department of Aeronautics and Astronautics,
Fudan University, Shanghai, China

ABSTRACT

Stochastics is one of key characteristics in rotor fault diagnosis, and severely influences the diagnostic accuracy, so that stochastic process is urgently considered in fault diagnosis. To improve the performance of rotor vibration fault diagnosis in stochastic operation process, this chapter discusses an effective process fault diagnosis method which is called process power spectrum entropy (PPSE) and support vector machine (SVM) (PPSE-SVM, short for) method. The fault diagnosis of PPSE-SVM is modeled by adopting PPSE method to precisely extract the process feature of rotor transient faults and applying SVM approach to

* Corresponding Author's Email: cwfei@fudan.edu.cn.

recognize the extracted PPSE characteristics. In respect of the simulation experiment of rotor vibration faults, the process signal data of four typical vibration faults (rotor imbalance, shaft misalignment, rotor-stator rubbing and pedestal looseness) under speed up and speed down were collected under multi-point (multiple channels) and multi-speed. By using the PPSE method, the PPSE values of these data for four faults were computed, and then were constructed as the fault feature vectors for the SVM modeling of rotor vibration fault diagnosis. As illustrated the diagnosis of rotor vibration faults, the proposed method possesses high precision, good learning ability, good generalization ability and strong fault-tolerant ability (robustness) in four aspects which are the diagnosis of fault types, the evaluation of fault severity, the recognition of fault location and the estimation of noise immunity for rotor stochastic vibration. The efforts of this study present a promising method, PPSE-SVM, for the diagnosis of rotor vibration faults with transients and the real-time monitoring of rotor vibration. The presented works are significant to accurately real-time detect and diagnose the process faults of rotating machinery like gas turbine under operation.

Keywords: process power spectrum entropy, support vector machine, rotor vibration, feature extraction, fault diagnosis

INTRODUCTION

Rotor system is one core system and is also one main fault source of rotating machinery like an aeroengine [1]. Rotor vibration seriously influences the security and reliability of the machine system operation [2]. With the development of the performance and reliability of rotating machinery such as gas turbine, aeroengine, and so forth, it is urgent to detect and predict vibration fault of rotor system in advances, to prevent the disorder of machinery system [3]. Therefore, how to predict and control the vibration faults of rotor is one of hot issues in preventing the failures of mechanical system. In this background, numerous feasible and effective approaches on vibration fault analysis and diagnosis emerge, such as wavelet analysis approach [3], support vector machine [4,5], eigenvectors algorithm [6], expert system [7], transfer entropy [8], information entropy method [9], and so on. However, most of present vibration analytical techniques need a mass of vibration samples to establish fault diagnosis model for vibration fault diagnosis from a qualitative perspective. The qualitative analytical methods, in fact, possess some blindness in fault diagnosis, so that they seriously influence diagnosis accuracy, because these approaches did not focus on the description of fault varying process with time and the quantitative assessment of vibration faults. Meanwhile, it is always difficult to collect a large number of vibration fault data. To improve the effectiveness and feasibility of fault diagnosis, it is required to regard the process and quantitative factors to determine fault types, failure severity, fault location and even development tendency. The development of information entropy theory provides a heuristic thought for quantitative analysis of rotor vibration conditions, because the information entropy approach can accurately and quantitatively reflect the uncertainty degree of vibration signal distribution or change [8, 10-12]. Information entropy technique as a process fault diagnosis method has been animato researched and applied to machinery fault diagnosis [5, 13–17]. For the quantitative diagnosis of rotor vibration faults, specially, one process information entropy method, i.e., process power spectrum entropy (PPSE) method, was developed and demonstrated to be effective from a quantitative perspective [13]. These techniques are proved to be feasible in some simple fault diagnoses. However, for the complex vibration signals of large rotating machinery with multi-fault classes and coupling fault, it is necessary to employ information fusion technology to comprehensively process multi-sensor information, and this technology can breezily improve the precision of fault analysis with regard to many kinds of extracted information for faults [15, 18-20]. As a new intelligent pattern recognition method, support vector machine (SVM) is likely to solve the problems of small-sample, non-linear and high dimension, by adopting structural risk minimization instead of empirical risk minimization, maximum margin principle, mapping from low dimensional space to high dimensional space. The SVM is validated to have numerous strengths such as complete theory, good adaptability, global optimization, short training time, good generalization ability in fault diagnosis, and so forth [1, 4, 5, 21–26].

In this chapter, we propose an effective approach, which is named as process power spectrum entropy-SVM (PPSE-SVM) method, for the process diagnosis of rotor vibration faults by fusing PPSE method and SVM theory. In respect of this method, the randomization of vibration fault and the difficulty of extracting vibration fault samples are addressed. Four faults are simulated to acquire vibration data in both speed up and speed down for rotor, based on rotor vibration simulation rig. In line with the PPSE method, the PPSE values (one information feature) of all the fault data are calculated. Then the PPSE values for each fault to construct eigenvectors as the samples of rotor process fault diagnosis. In light of the PPSE vectors, the SVM fault diagnosis model was established for all faults. The feasibility and validity of the proposed PPSE-SVM method is verified by diagnosing rotor vibration fault diagnoses from four aspects such as fault types, failure severity, fault point and anti-noise ability.

PROCESS POWER SPECTRUM ENTROPY METHOD

Shannon first proposed the concept of information entropy and applied it to evaluate information capacity [4, 10–17]. When $\boldsymbol{M}$ indicates a Lebesgue space with an algebra δ that is generated by a measurable set H and a measure μ ($\mu(\boldsymbol{M})=1$), and the space $\boldsymbol{M}$ can be decomposed into a limited incompatible partitioning $\boldsymbol{A}=(A_i)$ and meet $\boldsymbol{M}=\bigcup_{i=1}^{n} A_i$ and $A_i \cap A_j = 0, \forall i \neq j$, based on the information entropy theory [17], the information entropy $E(\boldsymbol{A})$ of $\boldsymbol{A}$ is denoted by

$$E(\boldsymbol{A}) = -\sum_{i=1}^{n} \mu(A_i) \log \mu(A_i) \tag{1}$$

in which $\mu(A_i)$ is the measurement of the sample A_i, in which i=1, 2, …, n.

As seen in Eq. (1), information entropy E actually reflects the chaotic degree of uncertain factors in a system. In respect of the calculation of information entropy in Eq. (1), more disorder and randomness of the system have larger information entropy values; and vice versa [9].

Based on the energy method, the power spectrum entropy (PSE) values of rotor vibration signal can be extracted from vibration signals as frequency domain features of the signal. When the discrete Fourier transform of a single-channel signal $\{x_t\}$ is denoted by $X(\omega)$, i.e.,

$$X(\omega) = \frac{1}{2n\pi} \sum_{t=1}^{N-1} x_t e^{-j\omega t} \tag{2}$$

The power spectrum of $X(\omega)$ [1] is expressed as

$$S(\omega) = \frac{1}{2\pi N} |X(\omega)|^2 \tag{3}$$

According to the energy conserve law in the signal transformation process from time domain to frequency domain [11], the Eq.(3) can be rewrote as

$$\sum x^2(t)\Delta t = \sum \left|X(\omega)\right|^2 \Delta\omega \tag{4}$$

The above equation illustrates that the total energy of signal and the sum of the sub-energy of each frequency component are same. Therefore, the power spectrum entropy $S = \{S_1, S_2, ..., S_N\}$ on each natural frequency is termed as one original signal partition, the corresponding information entropy (usually also called as PPSE) is defined as

$$E_f = -\sum_{i=1}^{N} q_i \log q_i \tag{5}$$

in which the subscript f indicates frequency domain; q_i is the ratio of the ith power spectrum to the whole spectrum, i.e.,

$$q_i = \frac{S_i}{\sum_{i=1}^{N} S_i} \tag{6}$$

We call the Eq. (5) as the power spectrum entropy (PSE) of the signal *S*. When the PSE is employed to fault diagnosis, here the method is called as the PSE method. The PSE stands for the spectrum structure of a single-channel vibration signal. The uniformity of energy distribution for vibration signals indicates the complexity and uncertainty of the signals. Similarly, the condition that the vibration energy distribution is more uniform in the whole frequency composition indicates that the signal is more complex and uncertain. When rotor speed up/down process is acquired from multiple rotation speeds, the PSE values under different rotation speeds describe the vibration status of the whole speed up/down process. Obviously, we also call the PSE method as process PSE (PPSE) method.

In most cases, information entropy approach lacks quantitative index so that information entropy values cannot be directly applied to diagnose rotor fault. The PPSE presenting the conditions of rotor vibration is promising to be applied to rotor fault diagnosis. However, the different moments and measuring points of vibration waveform for each fault signal always make their PPSE values distribute a certain range, and have some overlap among the PPSE distributions of different vibration faults [9,10,12,17]. For example, in respect of multiple simulation experiments, we acquire the PPSE values of four typical faults (i.e., rotor imbalance, shaft misalignment, rotor rubbing and pedestal looseness) as listed in Table 1.

Table 1. PPSE distributions of four rotor vibration faults

Fault type	**Distribution range**
Rotor imbalance	3.661~5.254
Shaft misalignment	3.956~5.421
Rotor rubbing	4.345~5.245
Pedestal looseness	3.432~5.689

As indicated in Table 1, there is large overlap among the distribution ranges of the four faults. For instance, when the PPSE value of an unknown fault is considered as 5.0, it is difficult to judge to which class (rotor imbalance, shaft misalignment, rotor rubbing or pedestal looseness) the unknown fault belongs. Therefore, it is urgent to develop an effective identification technique to address the above issue. The SVM method has been demonstrated to be an effective tool to address the overlap problem of the PPSE values of different fault modes by intelligent pattern recognition. Therefore, the SVM will be utilized to identify the PPSE features of four vibration faults under operation.

SUPPORT VECTOR MACHINE METHOD

Fundamental Theory

As a new machine learning method, the support vector machine (SVM) [21–26] can address the realistic issues in fault diagnosis such as small sample, nonlinear, high-dimensional pattern recognition, and so forth, since it has complete theory, good adaptability, global optimization, short training time and good generalization ability. In the SVM algorithm, only small support vectors in the training samples are considered to establish an SVM classifier for fault diagnosis rather than the whole vector set. Assuming that a fault training sample set is $\{(x_i, y_i), i = 1, 2, \cdots, L\}$ in which $x \in R^n$ and $y \in | \text{-}1, +1 |$, and $g(x) = (\omega \cdot x) + b$ stands for the general form of linear discriminated function in an *N*-dimensional space, the optimal hyper plane $(\omega \cdot x) + b = 0$ can correctly classify the data in the sample set. The distance between support vector and hyperplane is demonstrated by $1/\|\omega\|$. In this case, the problem of searching for hyperplane can be translated into the question of solving quadratic programming by adopting

$$\begin{cases} \min \dfrac{1}{2}(\omega \cdot \omega) + C\sum_{i=1}^{n} \zeta_i \\ s.t.\ y_i\left[(\omega \cdot x_i) + b\right] + \zeta_i \geq 1 \end{cases} \tag{7}$$

in which ω and b are the unknown coefficients of linear discriminated function; ζ_i is the relaxation factor of quadratic programming; C is the penalty factor of quadratic programming which stands for the compromise of classification interval and error rate.

In line with the optimization method in Eq. (7), both objective function and constraint conditions are convex functions so that the objective function only has a unique global optimal solution. For the linear programming problem, with Lagrange algorithm and $y_i\left[(w \cdot x_i) + b\right] = 1$, the decision function of optimal hyperplane is defined as

$$\begin{aligned} f(x) &= \operatorname{sgn}\left[\omega \cdot x + b^*\right] \\ &= \operatorname{sgn}\left(\sum_i a_i y_i x_i \cdot x + b^*\right) \end{aligned} \tag{8}$$

For the nonlinear programming problem, introducing a kernel function is to find the optimal hyperplane of feature vectors. The corresponding decision function is rewritten as

$$f(x) = \operatorname{sgn}\left(\sum_i a_i^* y_i k(x_i, x) + b^*\right) \tag{9}$$

The optimal hyperplane in the SVM is to map the input vector ***x*** into a high dimensional feature space ***Z*** by the preselected nonlinear mapping function to construct the optimal classification hyperplane in high dimensional feature space ***Z***. The output results of SVM classification function is a linear combination of intermediate nodes. Each node corresponds to one support vector. The principle of SVM is shown in Figure 1.

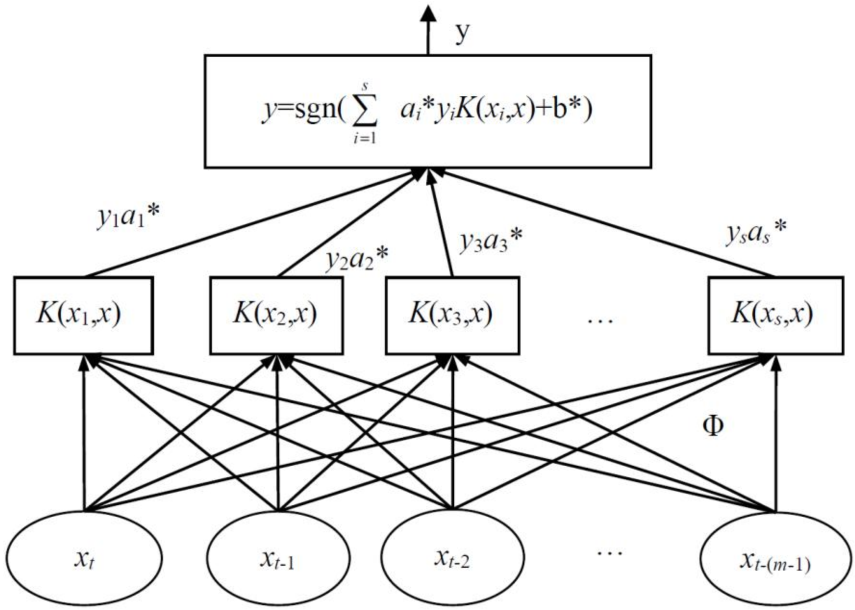


Figure 1. SVM schematic diagram.

In Figure 1, the aim of using kernel function is to avoid complex calculation in low dimensional feature space. The Kernel function generally comprises linear kernel function, polynomials kernel function, RBF kernel function, Sigmoid kernel function, Gauss function, and so on [4, 23–29]. The RBF kernel function is demonstrated to be effective in SVM modeling, and are widely used. In this study, the RBF kernel function is selected which is expressed by

$$k\left(x_i, x\right) = \exp\left[-\frac{\left\|x - x_i\right\|^2}{\sigma^2}\right] \tag{10}$$

in which σ is the width of kernel.

SVM Multiclass Classification Method

The faults of rotor involve many types of faults, so that rotor vibration fault diagnosis is a multiclass signal process problem in practice and requires establishing a multiclass SVM classifier. The construction methods of multi-class classifier include one to one (1–a–1) method, one against all (1–a–a) method, global optimization classification method, directed a cyclic graph SVM (DAGSVM) method, and so forth [15, 21, 25]. In this study, the 1–a–1 method was used to design SVM multi-classifier. According the algorithm of SVM multi-classifier with 1-a-1 method, a set of training samples is assumed as

$$T=\{(x_i,y_i),\cdots,(x_l,y_l)\}\in(X\times Y)^l$$

where $x_i \in X \in \mathbf{R}^n$; $y_i \in Y=(1,\cdots,M), i=1,2,\cdots,l$.

By searching for a discrimination function $f(x)$ in $\mathbf{R}^n$, it is assured that each input value x has one corresponding output value y. In fact, the essence of multi-class classification is how to find one reasonable rule to divide all points into M portions in $\mathbf{R}^n$. Based on the 1-a-1 method, the analytical procedure of SVM multi-class classification problem is summarized as follows:

1) Regard the jth class as the positive class and the rest M–1 classes as the negative class, in light of the SVM theory, the decision functions of the jth class is expressed by

$$f^j(x)=\mathrm{sgn}\left(g^j(x)\right) \tag{11}$$

where $g^j(x)=\sum_{i=1}^{l} y_i a_i^j k\left(x,x_j\right)+b^j$, j=1, 2, …, M.

2) Judge the input x belonging to the jth class in which j is the maximum label in $\left\{g^1(x),g^2(x),\cdots,g^M(x)\right\}$.

In line with the above steps, we can build a multi-class classifier of samples based on the SVM as the surplus training samples are correctly classified.

ROTOR SIMULATION EXPERIMENT

Selection of Typical Faults

In this study, we selected four typical rotor vibration fault modes, i.e., rotor imbalance, shaft misalignment, rubbing and pedestal looseness, as the object of study to study on rotor vibration fault diagnosis based on PPSE-SVM method. The rotor imbalance is often induced by unreasonable design, manufacturing and fixing error, attrition, and so forth. The rotor imbalance involves rotor mass imbalance, rotor initial bend and unbalanced coupling primarily. The shaft misalignment comprises both coupling misalignment and bearing misalignment. Contact-rubbing constantly occurs when dynamic-static gap, imbalance, misalignment and hot bend reduce. Pedestal looseness is often caused by bad fixing and long-term vibration.

Experiment of Simulation

Test Rig

To acquire enough fault data, rotor test rig as shown in Figure 2 is adopted to simulate four typical faults under different speeds and different channels. As indicated in Figure 2, the simulation system divided into test bench and measurement system. On the test bench, we selected a double rotor (birotor) system. The two rotors in this system are linked by a flexible coupling. One disk with equally distributed holes is installed on each rotor. On the locations A, B, C and D of pedestal, four acceleration sensors are installed, respectively, as shown in Figure 2. One speed sensor is used to measure rotor speed. The double rotor is driven by a motor. On the point

D, a bolt is utilized to simulate the rub-impact fault of rotor. The measurement system is constituted of signal acquisition instrument, signal-amplifier, speed controller, velocity indicator and computer. The signal acquisition instrument is used to collect the vibration signals from acceleration and speed sensors. Under the effect of the poor quality of sensor and measuring instrument as well as the influence of imbalance inherently, resonance and other signals maybe exist in rotor vibration signals gained from rotor experiment. However, it hardly influences the procedure used in this study. The reason is that the propose PPSE-SVM method is better in pure fault diagnosis of rotor vibration if it holds acceptable and high precision in the coupling fault diagnosis of rotor vibration. Meanwhile, the faults coupling case seems to be more reasonably simulate the real rotor system.

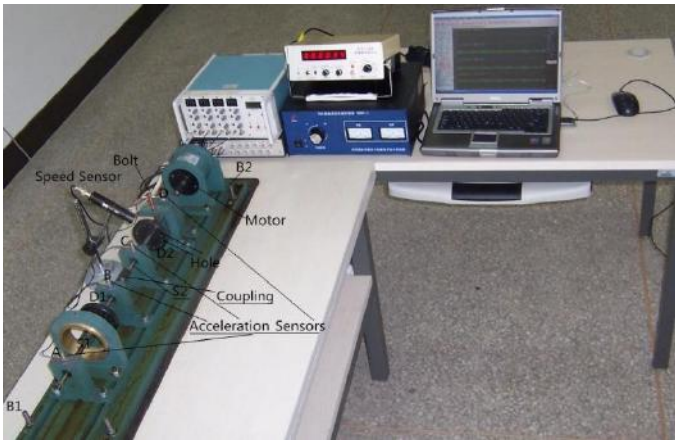


Figure 2. Test bench and measurement systems of rotor vibration simulation.

Experimental Process

To investigate the process features of rotor vibration, the four typical faults such as rotor imbalance, shaft misalignment, pedestal looseness and rubbing, are simulated on rotor vibration test rig when the rotational speed are controlled by adjusting from 0 rpm to 3 000 rpm. Each fault is simulated by multiple accelerated experiments. The acquired fault dada is collected by the interval of 100rpm sampling speed. In the measurement system, four acceleration sensors (four vibration signal channels) are fixed on rotor test rig, to measure four-point vibration accelerations and these signals are gathered as the original data of rotor vibration fault diagnosis. In this process, mass block is added in the holes of disk to simulate rotor imbalance fault. The shaft axes of two rotor (S1 and S2) are not consistent and keep some angle. In other words, the shaft axes of two rotor (S1 and S2) are not located on the same line. In this case, we simulate the shaft misalignment of rotor system. The looseness of one (B1) or two bolts (B1 and B2) on pedestal is employed to simulation the pedestal looseness fault of rotor system. The condition of bolt contacted rotor shaft is regarded to simulate rotor rubbing fault.

Original Data

According to the simulation experiment, a mass of original vibration signals for each typical fault is gathered when rotor speed is from 0 to 3 000 rpm with the 100 rpm sampling interval. One group of vibration signals includes 30 groups of vibration waveform under one measuring point and different speeds. Therefore, in one process of speed up or speed down experiment, we can collect 120 groups of vibration signal waveforms for each fault modes, which reflect the process characteristics of rotor vibration faults with rotational speed. For the point B, three-dimensional (3-D) spectrographs of normal state and four faults are shown in Figures 3 ~ 7.

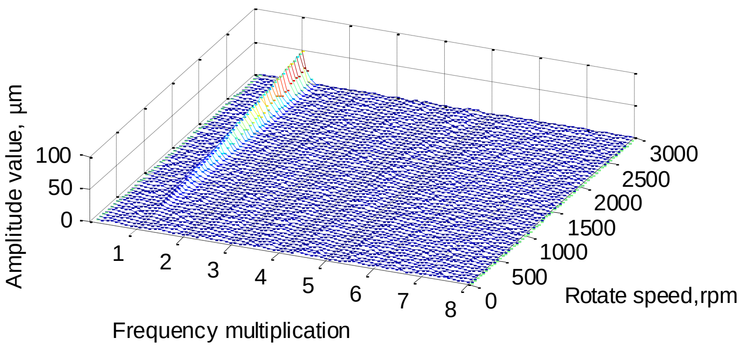


Figure 3. 3-D spectrograph of normal mode.

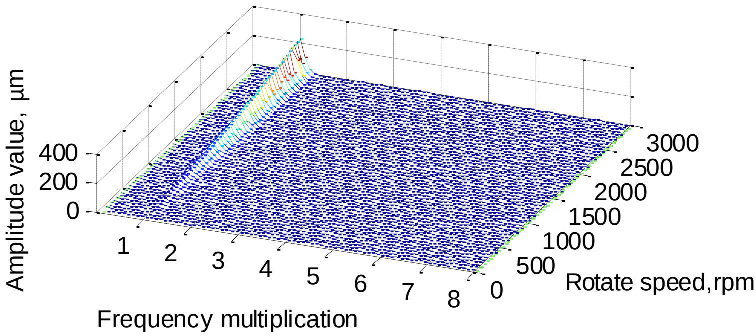


Figure 4. 3-D spectrograph of imbalance mode.

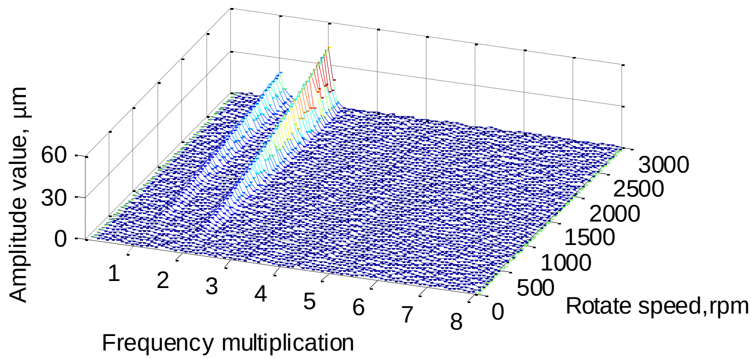


Figure 5. 3-D spectrograph of shaft misalignment mode.

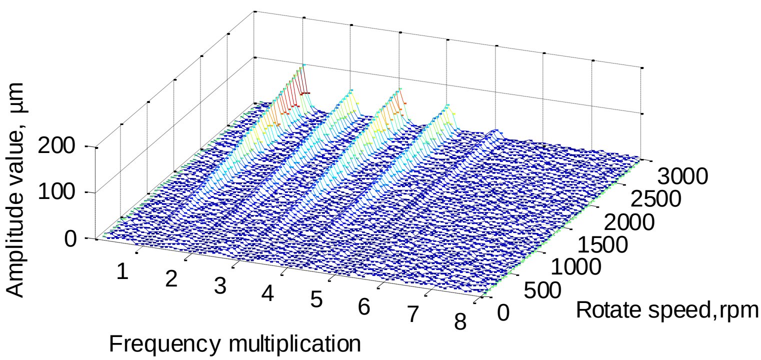


Figure 6. 3-D spectrograph of pedestal looseness mode.

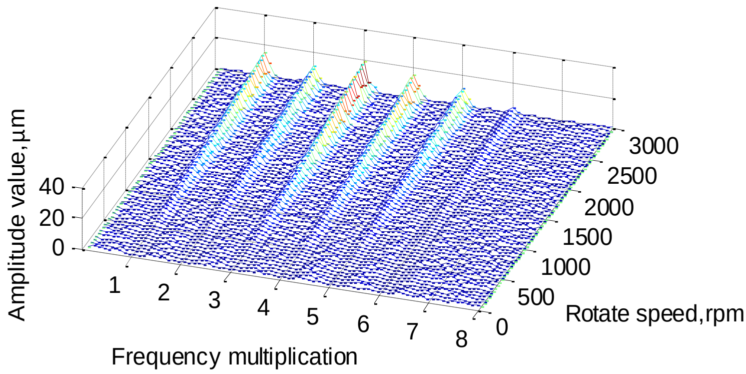


Figure 7. 3-D spectrograph of rubbing mode.

Analysis of Fault Data

As reveled in Figures 3~7, the main feature of rotor fault distributes in the rotated speed range [1 000 rpm, 3 000 rpm] with 100 rpm sampling interval. The speed band is selected to discuss the proposed PPSE-SVM method. Obviously, one group of vibration signals has 21 groups of vibration waveform under one measuring point and different speeds. Therefore, in once process of speed up or speed down experiment, we can gather 84 groups of vibration signal waveforms for each failure mode which reflects the process characteristics of rotor vibration faults. The speed up (or down) process of rotor is constituted of a number of states. Vibration shapes are different under disparate rotational speeds. In fact, the vibration waveforms record the full information of rotor vibration modes under different speeds and time. The vibration fault feature possesses some dispersiveness and randomness at each point. However, it can be predicted that the vibration fault feature in rotor vibration process is regular. Information entropy matrix absorbs information entropy values under multi-speed and multi-channel, reflecting the process regularity of vibration signals. Therefore, the information entropy matrix is used to describe the process regularity of rotor vibration signals. The original data which reflect the process characteristics of each fault can be gained on rotor vibration simulation experiment. According to PPSE method, the PPSE values of vibration waveform signals may be calculated. For one vibration fault, PPSE values, which full describe the process features of this vibration fault under multi-channel and multi-speed, are promising to be obtained for constructing one PPSE matrix which is regarded as the fault diagnosis samples of SVM model.

PROCESS DIAGNOSIS OF ROTOR VIBRATION FAULTS

Establishment of PPSE-SVM Model

In term of the presented PPSE-SVM method, the process diagnosis basic idea of rotor vibration faults is displayed as follows:

- ***Step 1***: Collect the rotor vibration data of four typical faults by conducting rotor vibration fault simulation experiment;
- ***Step 2***: Extract the characteristics (PPSE values) of four vibration faults based on PPSE method as the training and testing samples of SVM fault diagnosis modeling;
- ***Step 3***: Establish the SVM fault diagnosis model by the acquired training samples and accomplish the rotor vibration process fault diagnosis by distinguishing fault category, discriminating failure severity, judging fault location and validating the robustness of PPSE-SVM method.

The above process is named fusion fault diagnosis based on PPSE-SVM method. The detailed analytical procedure is shown in Figure 8.

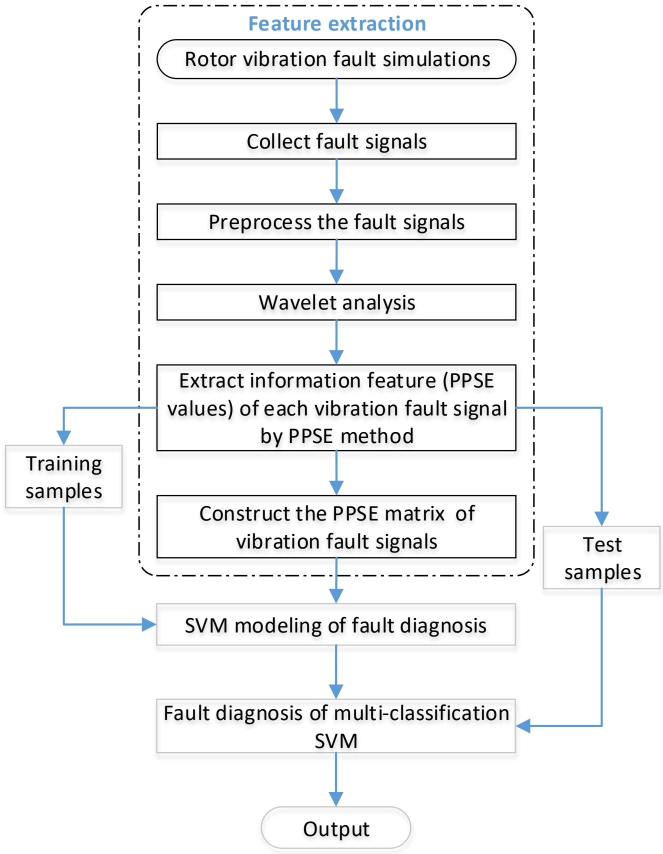


Figure 8. Procedure of rotor vibration process fault diagnosis with the PPSE-SVM method.

Extraction of PPSE Feature and Selection of Parameter

With respect to the developed PPSE method and MATLAB simulation software, the PPSE values of vibration fault reflecting the process features of speed up or speed down are gained. From the four measuring points in the rotor simulation experiments, four groups of PPSEs are integrated to present the variation of rotor vibration condition in speed up process. Each group of PPSE values of each channel has 21 information entropy values describing the process variation of vibration signal at the corresponding measuring point during speed up. All the PPSE values of each vibration fault are used to constitute one PPSE matrix (or PSE matrix), the three arises of which stands for PPSE value, rotate speed and measuring point (channel), respectively. The PPSE matrixes of four vibration faults are drawn in Figures 9~12.

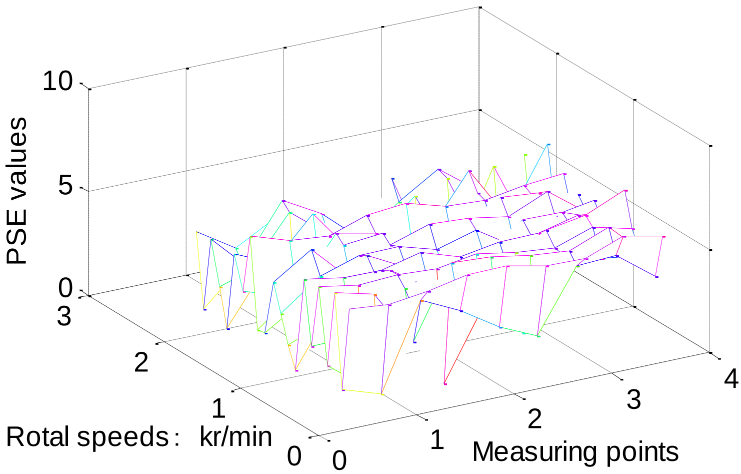


Figure 9. PPSE matrix of imbalance mode.

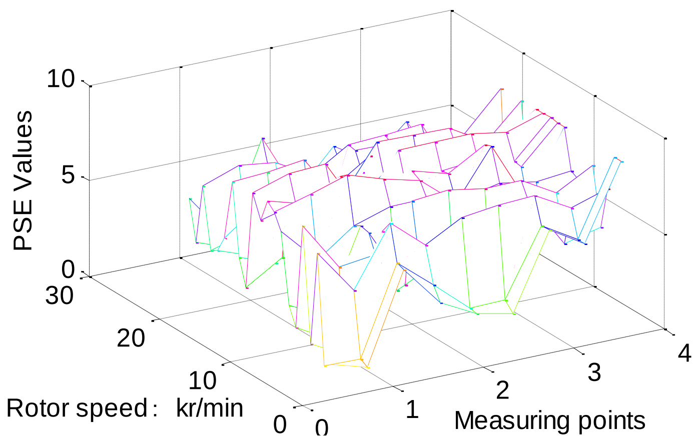


Figure 10. PPSE matrix of shaft misalignment mode.

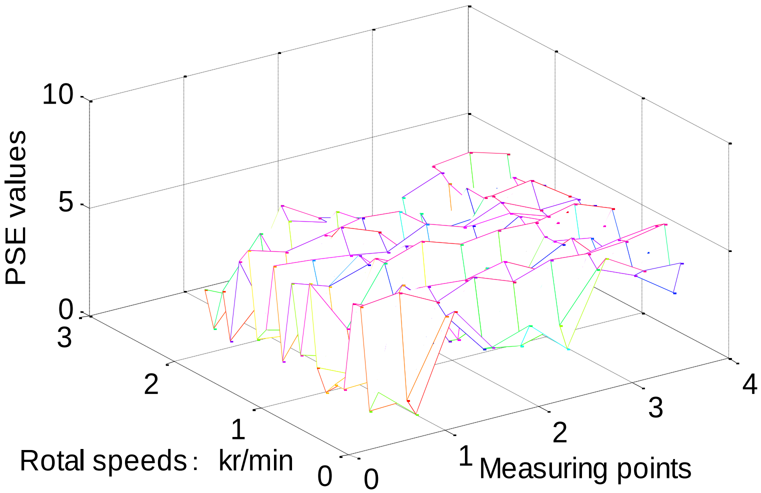


Figure 11. PPSE matrix of rotor rubbing mode.

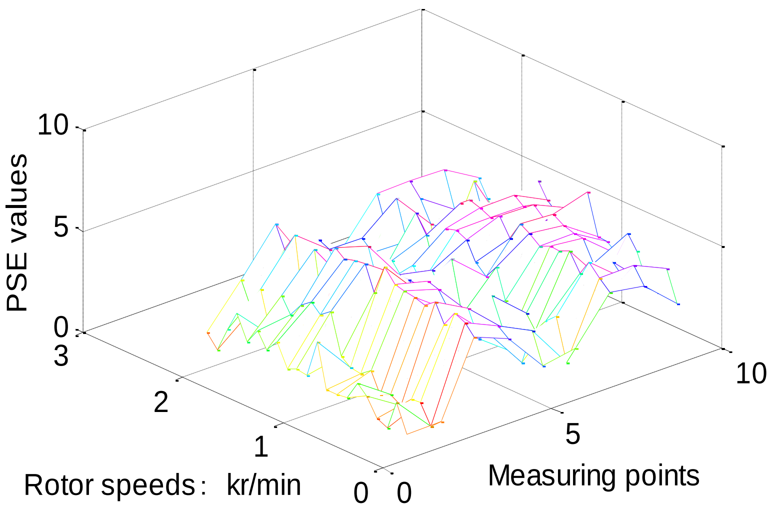


Figure 12. PPSE matrix of pedestal looseness mode.

Obviously, each measuring point has sequential 21 PPSE values. Hence, the 21 PPSE values were lined up as a feature vector $[E_1, E_2, \cdots, E_{21}]$. The vector was trained as the fault diagnosis sample of SVM modeling. Similarly, 40 fault vectors full reflecting rotor vibration process features were taken as the fault samples of one fault mode. The total sample number of rotor vibration fault diagnosis is 160.

The SVM fault diagnosis model is structured by MATLAB simulation environment and Lib-SVM toolbox. The kernel function of SVM is RBF kernel function $k(x_i, x) = \exp\left[-\frac{\|x - x_i\|^2}{\sigma^2}\right]$. By adopting the random automatically research method [3], it is gained that the optimal values of σ and C are 0.07 and 136, respectively. Experiences revealed that the classification effect of RBF is superior to others.

Diagnosis of Rotor Vibration Fault Category

We selected 10 fault samples of each vibration fault mode and considered them as training samples. The training samples are used to train the SVM model by obtaining the optimal classifications function to establish SVM diagnosis model. The selected training samples of each fault type are shown in Table 2.

To validate the learning ability and generalization ability of the established SVM model, the training samples of four faults were inputted into the SVM diagnosis model. It is illustrated from the results that the training samples are completely and correctly classified with the testing precision of 100%. In line with the trained SVM model, the rest 30 samples of each vibration fault mode are classified. The results are listed in Table 3.

As illustrated in Table 3, the average diagnostic precision of four testing samples is 96.67%, demonstrating that the trained SVM model has good learning ability and good generalization ability in the diagnosis of rotor vibration fault classification.

Table 2. Partial PPSE vectors of fault modes

Fault types	Lable of Sample	PPSE vector $[E_1, E_2, \cdots, E_{21}]$											Class number
Rotor imbalance	1	4.31	4.37	4.47	4.99	4.86	4.44	5.26	5.24	5.13	5.10	4.58	1
		4.51	4.31	4.94	4.38	4.92	4.77	4.43	4.87	4.98	5.16		
	2	4.40	4.73	4.47	4.62	4.71	3.96	5.04	5.08	5.0	4.44	4.53	1
		3.73	4.77	4.54	3.99	4.8	4.35	4.55	5.26	4.08	3.69		
Shaft misalignment	1	5.04	4.97	4.89	4.55	5.14	5.15	5.11	5.03	5.4	5.43	5.48	2
		5.02	5.13	5.03	5.2	4.95	4.82	4.75	4.50	4.59	3.98		
	2	4.79	4.84	4.75	4.12	5.33	5.08	5.29	5.09	5.34	5.4	4.94	2
		5.31	5.49	5.17	5.25	4.98	4.94	4.63	4.4	3.95	5.27		
Rotor rubbing	1	5.29	4.87	4.99	5.19	5.34	5.08	5.27	5.46	5.27	5.69	5.2	3
		5.12	5.38	4.84	5.21	5.62	4.79	5.17	5.04	5.37	5.15		
	2	4.91	5.16	5.14	4.36	5.23	5.45	5.4	5.02	5.36	5.20	5.46	3
		5.53	5.57	5.39	4.72	3.79	5.06	5.27	5.05	5.36	5.17		
Pedestal looseness	1	3.58	5.26	4.78	4.87	4.79	5.28	5.07	5.27	5.41	5.14	4.90	4
		5.32	5.36	5.31	5.19	5.019	5.42	4.36	5.29	4.18	5.14		
	2	4.51	4.91	5.17	5.37	5.03	5.36	5.11	5.29	5.57	5.04	5.02	4
		5.32	5.23	5.34	5.34	5.5	5.33	5.29	5.12	5.53	5.27		

Table 3. Diagnosis result of rotor vibration fault categories

Fault types	Testing sample number	Correct identification number	Precision	Mean precision
Rotor imbalance	30	29	96.67%	96.67%
Shaft misalignment	30	29	96.67%	
Pedestal looseness	30	28	93.33%	
Rotor rubbing	30	30	100%	

Diagnosis of Fault Degree

To verify the PPSE-SVM method in the fault severity diagnosis of rotor vibration, rotor pedestal looseness was selected as the object of study. Two failure status of rotor pedestal looseness (i.e., one pedestal looseness (suffering from B1 looseness) and two pedestal loosenesses (suffering from B1 and B2 loosenesses)) were simulated to collect fault data based on rotor test rig. All the PPSE values of the data were extracted. 40 groups of data of each failure status are selected as the fault samples, respectively. Similarly, 10 groups of fault vectors are considered as the training samples and the surplus 30 groups are regarded as the testing samples. By training and testing the SVM diagnosis model with the training samples, the results show that the testing accuracy with the training samples is also 100%. Thus, it is indicated that the trained SVM model holds good learning ability in rotor vibration fault severity diagnosis. The remainder 30 groups of testing samples of different fault degree are diagnosed by the trained SVM model. The diagnosis result of SVM model is listed Table 4. The result illustrates that only three samples are mistakenly judged with the diagnosis precision of 0.95. Therefore, the developed PPSE-SVM method can availably judge rotor vibration fault degree.

Table 4. Diagnosis results of fault severity

Fault degree	Test sample number	Correct identification number	Precision	Mean precision
One pedestal looseness	30	29	96.67%	95%
Two pedestal looseness	30	28	93.33%	

Diagnosis of Fault Points

Because different measuring points corresponds to different locations on rotor vibration simulation test bench, the measuring points of sensors are selected as the object of study. The four measuring points A, B, C and D are shown Figure 2. The data of measuring points are extracted as the location fault data. In a similar way, the PPSE values of the data were calculated based on the PPSE-SVM method and 40 groups of fault data of each measuring point were regarded as the vectors of samples, in which 10 fault data were regarded as training samples and the rest of 30 fault data were considered as test samples for each measuring point. Through rotor vibration fault point diagnosis, the results show that the testing accuracy with the training samples is 100% and the average diagnosis precision for the test samples is 93.33%, as shown in Table 5. It is indicated in Table 5 that the PPSE-SVM diagnosis method is feasible and effective in the diagnosis of rotor vibration fault points.

Table 5. Diagnosis results of rotor vibration fault locations

Fault points	Test sample number	Correct identified number	Precision	Mean precision
A	30	28	93.33%	93.33%
B	30	29	96.67%	
C	30	28	93.33%	
D	30	27	90%	

Verification of PPSE-SVM Robustness

To support the robustness of the proposed PPSE-SVM method in rotor vibration fault diagnoses including the judgments of fault category, severity and location, the original signals of the aforementioned test samples of three fault types were overlapped by Gaussian white noise with the mean value of 0 and the variance of 5. The PPSE values of the original signals of vibration fault were obtained as the new test samples based on the PPSE method. Lastly, the new test samples of three fault types were inputted into the corresponding SVM fault diagnosis models trained in the above. The analytical results are listed in Table 6.

Table 6. Robustness test of PPSE-SVM method in rotor vibration fault category, severity and locations

Fault type	Test sample number	Correct number	Precision	Reduction precision	Mean precision
Category	120	115	95.83%	0.83%	94.33%
Severity	60	56	93.33%	1.67%	
Location	120	112	93.33%	0	

As shown in Table 6, after overlapping Gaussian noise to the original vibration signals of test samples, the diagnosis precisions of three fault types (category, severity and location) are 0.9583, 0.9333 and 0.9333, respectively, and only reduce by 0.0083, 0.0167 and 0, relative to these before superposition, severally. Therefore, the mean precision of rotor vibration fault diagnosis reaches to 0.9433. The results reveal that the proposed PPSE-SVM method has good fault-tolerant capability and strong robustness in anti-noise-interference.

CONCLUSION

The objective of this effort is attempted to advance a process fault diagnosis method–Process Power Spectrum Entropy and Support Vector

Machine (PPSE-SVM) method, by fusing the advantages of information entropy method and SVM theory for rotor vibration fault diagnosis from a process perspective based on the information fusion technique. Through rotor vibration fault diagnosis based on the PPSE-SVM method, some conclusions are drawn as follows:

1) The Process Power Spectrum Entropy (PPSE) values can effectively reflect the process variation of rotor vibration signals.
2) PPSE-SVM model can be established by small samples-PPSE feature vectors extracted from the fault vibration data of rotor fault vibration simulation experiments.
3) The PPSE-SVM model trained by the PPSE feature vectors is demonstrated to be an efficient fault diagnosis model, which possesses strong learning ability, generalization ability and fault tolerance ability because of high testing precision (100%). High diagnosis precision (respectively 0.9667, 0.95 and 0.9333) and strong anti-noise-interference ability (0.9433 in precision) in the rotor vibration fault diagnosis and analysis on fault category, failure severity, fault points and robustness from a process perspective.
4) The presented PPSE-SVM method is also proved to be effective and reasonable, and this study provides a promising diagnosis technology for rotor vibration faults.

Some idealized factors were considered for rotor vibration fault diagnosis based on the PPSE-SVM method on rotor vibration simulation test bench in this study. For complex machinery like aeroengine, the validity of the presented PPSE-SVM method need further to be verified in vibration fault diagnosis.

ACKNOWLEDGMENT

This study is supported by the National Natural Science Foundation of China (Grant Nos. 51975124 and 51605016), Research Start-up Funding of Fudan University (Grant No. FDU38341) and Shanghai International Cooperation Project of One Belt and One Road of China (Grant No. 20110741700). The authors would like to thank them.

REFERENCES

[1] Fei Chengwei, and Bai Guangchen, Tang Wenzhong, and Ma Shuang. "Quantitative diagnosis of rotor vibration fault using process power spectrum entropy and support vector machine method." *Shock and Vibration* 2014 (2014): 957531.

[2] Ewins DJ. "Control of vibration and resonance in aeroengines and rotating machinery." *International Journal of Pressure Vessels and Piping* 87(2010): 504-510.

[3] Al-Badour F, Sunar M, and Cheded L. "Vibration analysis of rotating machinery using time–frequency analysis and wavelet techniques." *Mechanical Systems and Signal Processing*, 25 (2011): 2083-2101.

[4] Fei Chengwei, and Bai Guangchen. "Wavelet correlation feature scale entropy and fuzzy support vector machine approach for aeroengine whole-body vibration fault diagnosis." *Shock and Vibration* 20(2013): 341-349.

[5] Cui Hongxi, Zhang Laibin, Kang Rongyu, and Lan Xingyang. "Research on fault diagnosis for reciprocating compressor valve using information entropy and SVM method." *Journal of Loss Prevention in the Process Industries* 22(2009): 864-867.

[6] Tse PW, Gontarz S, and Wang XJ. "Enhanced eigenvector algorithm for recovering multiple sources of vibration signals in machine fault diagnosis." *Mechanical Systems and Signal Processing* 21(2007): 2794-2813.

[7] Wu Jianda, and Liu Chiuhong. "An expert system for fault diagnosis in internal combustion engines using wavelet packet transform and neural network." *Expert Systems with Applications*, 36(2009): 4278-4286.

[8] Overbey LA, and Todd MD. "Dynamic system change detection using a modification of the transfer entropy." *Journal of Sound and Vibration* 322(2009): 438-453.

[9] Qu Liangsheng, Li Liangming, and Lee Jay. "Enhanced diagnostic certainty using information entropy theory." Advanced *Engineering informatics* 17(2003): 141-150.

[10] Xing Xiusan. "Physical entropy, information entropy and their evolution equations." *Science in China Series A: Mathematics* 44(2001): 1331-1339.

[11] Endo H, and Randan RB. "Enhancement of autoregressive model-based gear tooth fault detection technique by the use of minimum entropy deconvolution filter." *Mechanical systems and Signal Process* 21 (2007): 906-919.

[12] Ai Yanting, Guan Jiaoyue, Fei Chengwei, Tian Jing, and Zhang Fengling. "Fusion information entropy method of rolling bearing fault diagnosis based on n-dimensional characteristic parameter distance." *Mechanical Systems and Signal Processing*, 88 (2017): 123-136.

[13] Wang Jun, Ai Yanting, Liu Xiufang, and Sun Xiaoqian. "Study on quantitative diagnosis method of rotor vibration faults based on process information entropy." Proceeding of the 2nd International Conference on Computer Engineering and Technology (ICCET), Chengdu, China, 2010, Vol. 5: 407-411.

[14] Ye Jun. "Fault diagnosis of turbine based on fuzzy cross entropy of vague sets." *Expert Systems with Applications* 36(2009): 8103-8106.

[15] Nicolás García-Pedrajas, and Domingo Ortiz-Boyer. "An empirical study of binary classifier fusion methods for multiclass classification." *Information Fusion* 12(2011): 111-130.

[16] Tian Jing, Ai Yanting, Fei Chengwei, Zhao Ming, Zhang Fengling, and Wang Zhi. "Fault diagnosis of inter-shaft bearings using fusion

information exergy distance method." *Shock and Vibration* 2018(2018): 7546128.

[17] Fei Chengwei, Choy YatSze, Tang Wenzhong, and Bai Guangchen. "Multi-feature entropy distance approach with vibration and AE signals for process feature extraction and diagnosis of rolling bearing faults." *Structural Health Monitoring-An International Journal* 17(2018):156-168.

[18] Fei Chengwei, Lu Cheng, and Liem P.Rhea. "Decomposed-coordinated surrogate modelling strategy for compound function approximation and a turbine blisk reliability evaluation." *Aerospace Science and Technology* 95 (2019): UNSP105466.

[19] Lu Cheng, Feng Yunwen, Fei Chengwei, and Bu Siqi. "Improved decomposed-coordinated Kriging modeling strategy for dynamic probabilistic analysis of multi-component structures." *IEEE Transactions on Reliability*. Online. Doi: 10.1109/TR.2019.2954379.

[20] Lu Cheng, Feng Yunwen, Fei Chengwei, and Bu Siqi. "Decomposed-coordinated framework with enhanced extremum Kriging for multi-component dynamic probabilistic failure analyses." *IEEE Access* 7(1) (2019): 163287-163300.

[21] Ding Weidong, and Yuan Jingqi. "Spike sorting based on multi-class support vector machine with super position resolution." *Medical & Biological Engineering & Computing* 46(2008): 139-145.

[22] Zhang Chunyi, Wang Ze, Fei Chengwei, Yuan Zeshan, Wei Jingshan, and Tang Wenzhong. "Fuzzy Multi-SVR Learning Model for Reliability-Based Design Optimization of Turbine Blades." Materials 12(2019): 2341.

[23] Widodo Achmad, and Yang Bo-Suk. "Support vector machine in machine condition monitoring and fault diagnosis." *Mechanical Systems and Signal Processing* 21(2007): 2560-2574.

[24] Yuan Shengfa, and Chu Fulei. "Support vector machines-based fault diagnosis for turbo-pump rotor." *Mechanical Systems and Signal Processing* 20 (2006): 939-952.

[25] Saimurugan M, Ramachandran, KI, Sugumaran V, and Sakthivel NR. "Multi component fault diagnosis of rotational mechanical system

based on decision tree and support vector machine." *Expert Systems with Applications* 38 (2011): 3819-3826.

[26] Fei Chengwei, Choy Yat-Sze, Hu Dianyin, Bai Guangcheng, and Tang Wenzhong. "Dynamic probabilistic design approach of high-pressure turbine blade-tip radial running clearance." *Nonlinear Dynamics* 86 (2016): 205-223.

[27] Fei Chengwei, and Bai Guangchen. "Distributed collaborative probabilistic design for turbine blade-tip radial running clearance using support vector machine of regression." *Mechanical Systems and Signal Processing* 49 (2014): 196-208.

[28] Tse Peter W, Peng YH, and Yam Richard. "Wavelet analysis and envelop detection for rolling element bearing fault diagnosis their effectives and flexibilities." *Journal of Vibration and Acoustics* 12 (2000): 303-310.

[29] Fei Chengwei, and Bai Guangchen. "Nonlinear dynamic probabilistic analysis for turbine casing radial deformation based on extremum response surface method-based support vector machine." *Journal of Computational and Nonlinear Dynamics* 8(4) (2013): 041004.

BIOGRAPHICAL SKETCH

Cheng-Wei Fei, PhD

Affiliation: Fudan University, Shanghai, China

Business Address: Rm 2517, East Main Building, Guanghua Building, Handan RD. 220, Yangpu District, Shanghai 200433, P.R. China

Research and Professional Experience:

Cheng-Wei Fei, PhD, is currently research professor at Department of Aeronautics and Astronautics, Fudan University, Shanghai, China. He received his B.S. degree in electrical engineering and automation from Fujian University of Technology, Fuzhou, China, in 2007, and the M.S.

degree in aerospace engineering from Shenyang Aerospace University, Shenyang, China, in 2010, and the Ph.D. degree in aerospace engineering from Beihang University, in 2014, Beijing, China. From 2014 to 2017, he was a postdoctoral fellow (Hong Kong Scholars) at Department of Mechanical Engineering, the Hong Kong Polytechnic University, Hong Kong, and then from 2017 to 2018, was a research fellow in Department of Mechanical and Aerospace Engineering, Hong Kong University of Science and Technology, Hong Kong. Since 2018, he has been a Research Professor at Department of Aeronautics and Astronautics, Fudan University, Shanghai, China.

His research interests include surrogate model, multidisciplinary design optimization (MDO), reliability-based design optimization (RBDO), AI and machine/deep-based learning structural health monitoring for aircrafts and aero-engines. He has published 2 books, more than 80 journal paper and about 10 conference papers as well as 3 inventions. He was once the leader guest editors of Advances in Acoustics and Vibration and Advances in Mechanical Engineering. He is the reviewer of more than 10 journals such as IEEE Access, Non. Dyn., Mech. Syst. Signal Proc., Aerosp. Sci. Tech., Reliab. Eng. Syst. Safety, etc.

Dr. Fei is a recipient of National Excellent Doctoral Dissertation in 2016, Hong Kong Scholars Awards in 2016, and The Academic Scholarship of Chinese Ministry of Education for Ph.D Candidates in 2012 and National Graduate Scholarship of China, 2012.

Professional Appointments:

Member of American Institute of Aeronautics and Astronautics (AIAA), Chinese Society of Aeronautics and Astronautics (CSAA) and American Society of Mechanical Engineers (ASME).

Publications from the Last 3 Years:

[1] Lu C, Feng YW, Fei CW, Bu SQ. Improved decomposed-coordinated Kriging modeling strategy for dynamic probabilistic analysis of multi-component structures [J]. IEEE Transactions on Reliability. Doi: 10.1109/TR.2019.2954379.

[2] Lu C, Feng YW, Fei CW, Bu SQ. Decomposed-coordinated framework with enhanced extremum Kriging for multi-component dynamic probabilistic failure analyses[J]. IEEE Access, 2019, 7(1): 163287-163300.

[3] Fei CW, Lu C, Liem R.P. Decomposed-coordinated surrogate modelling strategy for compound function approximation and a turbine blisk reliability evaluation [J]. *Aerospace Science and Technology*, 2019, 95: UNSP105466.

[4] Song LK, Bai GC, Fei CW, Tang WZ. Multi-failure probabilistic design for turbine bladed disks using neural network regression with distributed collaborative strategy [J], *Aerospace Science and Technology*, 2019, 92: 464-477.

[5] Tian J, Ai YT, Fei CW, Zhang FL and Choy YS. Dynamic modeling and simulation of inter-shaft bearings with localized defects excited by time-varying displacement [J]. *Journal of Vibration and Control*, 2019, 25(8): 1436-1446.

[6] Song LK, Bai GC, Fei CW, Wen J. Probabilistic LCF life assessment of turbine discs using DC-based wavelet neural network regression [J]. *International Journal of Fatigue*, 2019, 119: 204-219.

[7] Fei CW, Choy YS, Tang WZ, Bai GC. Multi-feature entropy distance approach with vibration and AE signals for process feature extraction and diagnosis of rolling bearing faults [J]. *Structural Health Monitoring-An International Journal*, 2018, 17(2):156-168.

[8] Lu C, Feng YW, Rhea. P Liem, Fei CW. Improved kriging with extremum response surface method for structural dynamic reliability and sensitivity analyses [J]. *Aerospace Science and Technology*, 2018, 76:164-175.

[9] Zhai X, Fei CW, Wang JJ, Choy YS. A stochastic model updating strategy-based improved response surface model and advanced Monte Carlo simulation [J]. *Mechanical Systems and Signal Processing*, 2017, 82(1): 323-338.

[10] Song LK, Fei CW, Bai GC, Yu LC. Dynamic neural network method-based improved PSO and BR algorithms for transient probabilistic analysis of flexible mechanism [J]. *Advanced*

Engineering Informatics, 2017, 33: 144-153.

[11] Song LK, Fei CW, Wen J, Bai GC. Multi-objective reliability-based design optimization approach of complex structure with multi-failure modes [J]. *Aerospace Science and Technology*, 2017, 64:52-62

[12] Fei CW, Choy YS, Hu DY, Bai GC, Tang WZ. Dynamic probabilistic design approach of high-pressure turbine blade-tip radial running clearance [J]. *Nonlinear Dynamics*, 2016, 86(1):205-223.

[13] Fei CW, Choy YS, Hu DY, Bai GC, Tang WZ. Transient probabilistic analysis for turbine blade-tip radial clearance with multiple components and multi-physics fields based on DCERSM [J]. *Aerospace Science and Technology*, 2016, 50: 62-70.

[14] Sun D, N Liu N, Fei CW, Hu GY, Ai YT and Choy YS. Theoretical and numerical investigation on the leakage flow characteristics of brush seals based on fluid-structure interaction [J]. *Aerospace Science and Technology*, 2016, 58:207-216.

[15] Zhang CY, Song LK, Fei CW, Lu C, Xie YM. Advanced multiple response surface method for reliability sensitivity analysis of turbine blisk with multi-physics coupling [J]. *Chinese Journal of Aeronautics*, 2016, 29(4): 962-971.

In: Support-Vector Machines
Editor: Pooja Saigal
ISBN: 978-1-53618-757-1

Chapter 4

HARDWARE ARCHITECTURES OF SUPPORT VECTOR MACHINE APPLIED IN PATTERN RECOGNITION SYSTEMS

Gracieth Cavalcanti Batista[1,*], Duarte Lopes de Oliveira[1], Washington Luis Santos Silva[2] and Osamu Saotome[1]

[1]Department of Eletronic Devices and Systems,
Technological Institute of Aeronautics,
São José dos Campos, SP, Brazil
[2]Department of Eletronics,
Federal Institute of Maranhão, São Luís, MA, Brazil

ABSTRACT

Support Vector Machine (SVM) has been applied in many areas, and it has been proved to be an essential tool to find and improve solutions. Pattern recognition systems are one of the areas where SVM has been used on a large scale, applied in software and hardware implementations. SVM is a technique of Machine Learning (ML) based on Statistical

[*] Corresponding Author's Email: gracieth.cavalcanti@gmail.com.

Learning, and it is a dichotomic algorithm (classification of only two classes/patterns at a time). It usually presents two problems: computational load and time-consuming, which are significant concerns, mainly in pattern recognition applications where response time needs to be as fast as possible. In order to solve those problems, researchers have separated the implementation of training from the classification phase (e.g., using training implementation in software and classification in hardware), and they have used some optimization algorithms. In this chapter, forms of SVM application are analyzed and discussed in Pattern Recognition systems. Besides, in a specific Automatic Speech Recognition (ASR) system, SVM is used and explained in more detail. In this specific system, the Particle Swarm Optimization (PSO) algorithm was used combined with the SVM training, and they were implemented in software, called PSO-SVM hybrid training. Also, PSO finds the best position of the data population in coordinate space, and SVM separates one class from another through an optimal hyperplane using the "One-vs-All" multi-class technique. About the classification phase, SVM is implemented in synchronous and asynchronous hardware, in Field Programmable Gate Array (FPGA) prototyping for circuit response analysis. Many ways of SVM hardware implementation are presented, as follows: combinatorial datapath with Finite State Machine (FSM) controller, linear pipeline architecture with synchronous controllers, and linear pipeline architecture with asynchronous controllers. As a result, we analyze the advantages and disadvantages of the asynchronous paradigm in SVM hardware application, which is a brand new idea. Besides, from a comparison among those circuit configurations, it is proved that this brand new idea is promising and useful.

Keywords: Support Vector Machine, hardware architectures, FPGA implementation, pattern recognition system

INTRODUCTION

Pattern recognition algorithms have been used commonly in order to accelerate manual processes and to automate entirely most of these. Speech recognition is one of the pattern recognition areas, and it is very important and used in many devices currently (Burges 1998). In that kind of algorithm, there are applications of Artificial Intelligence (AI) algorithms such as Neural Networks, Deep Learning, Machine Learning (ML), Heuristic methods, ... ML techniques employ a principle of inference called induction, in which generic conclusions are drawn from a particular set of examples (Haykin 2002). One of the ML techniques is the Support Vector Machine (SVM), which is based on the Statistical Learning Theory (SLT). This theory aims to establish mathematical conditions that allow us to choose a classifier, with excellent performance, for the data set available for training and testing. In other words, this theory seeks to find a good classifier with excellent generalization performance taking into consideration the whole dataset. However, this classifier refrains from particular cases, which defines the ability to predict correctly the class of new data from the same domain in which the learning took place (Hebb 1949). Classifiers that separate data using a hyperplane are called linear, and thus, the SVM algorithm fits in this definition. Then, the dataset to be used in training and classification phases is very important because the SVM algorithm must also deal with non-linear problems (Haykin 2002). The main goal of SVM is to find the optimal hyperplane of separation between two classes (patterns) because the classification can be done just two-by-two classes, i.e., just two classes are separated from each other at a time (Bresolin 2008). In order to solve non-linear separation problems, SVM uses a kernel function. In this function, there is a transformation in the data where they are initially in an original space of low dimensionality (called input space), and they are reallocated to a space of high dimensionality (called feature space) (Scholkopf, et al. 1997).

Vapnik and Chervonenkis (Vapnik, The nature of statistical learning theory 2000) (Vapnik and Chervonenkis, On the uniform convergence of relative frequencies of their probabilities to events 1967) introduced SVM. Since then, it has been applied in many areas. However, a problem in its employment is the computational load in reason of its mathematical model. Many techniques are combined with SVM training in order to decrease its processing time and data (Kanisha, et al. 1998), such as optimization algorithms. In addition to solve the load computation problem and to have the fastest time response possible with viability, many researchers work with part of the SVM algorithm in hardware. It is widespread to use a Field Programmable Gate Array (FPGA) device to prototype the hardware architecture. Because FPGA offers many design advantages (such as performance, time to market, cost, reliability, long term maintenance) when compared to Application Specific Integrated Circuit (ASIC) or Application Specific Standard Product (ASSP) (P. P. Chu 2008).

Then, in this chapter, we can find several examples from the literature of how to implement SVM applied in pattern recognition systems in different ways, using FPGA prototyping. Also, in more detail, there are synchronous and asynchronous SVM hardware architectures of the classification phase implemented in FPGA and applied in an Automatic Speech Recognition (ASR) system of thirty voice patterns.

RELATED WORKS

Literature Review

Pattern Recognition is a specialty that has been in full expansion, and there are many ways to perform pattern recognition using SVM, such as training and classification phases implemented in software; and just one of the phases in software and the other one in hardware. However, most existing hardware designs utilize the well-known pipeline technique, exploiting the parallel processing capabilities of the FPGA in order to accelerate the classification task. Other hardware designs are implemented using parallel systolic array architecture and a multiplier-less approach. In contrast, old versions of FPGAs are used for implementations without considering crucial embedded systems constraints like low power consumption constraints.

From a review in the literature, we identify studies that presented some excellent results in the subject of SVM hardware applications, all of them being in the synchronous paradigm. Beginning with (Papadonikolakis and Bouganis 2010), the authors proposed an SVM hardware application implemented in FPGA to recognize images, where the Kernel function is Gaussian, and it is a binary classification problem. However, such work was about the SVM training phase implementation in hardware. They used an Altera's Stratix III device operating at 160-200MHz. It is a useful contribution but in the circuitry area subject (DSPs/LUTs amount).

Patil et al. proposed systolic array architecture of multi-class SVM, where the training phase was done in Matlab software and the classification phase in a Xilinx Virtex-6 FPGA device, applied in a facial expression recognition system (Patil, et al. 2012). The Kernel function was the Polynomial, and they reached a power reduction of 3-5% due to the reconfiguration of the device, i.e., they achieved a power consumption of 10mW.

Koide et al. proposed a 2-stage pipeline architecture in the SVM classification phase of two classes applied in colorectal cancer detection (Koide, et al. 2014). They used a linear configuration of the SVM (i.e., without kernel function application). The chosen device was an Altera Stratix IV, operating at 100MHz of frequency. However, they achieved a latency of 4.7ms, which is high when compared to other pattern recognition systems like the proposed ones in this work.

Jallad and Mohammed reached 1% of use in LUTs of SVM classification phase implementation in a Xilinx Virtex-5 FPGA device application for two classes, which is much for a classification of just two classes (Jallad and Mohammed 2014), considering the results from other related works from the literature.

Pietron et al. used an SVM classification in a Xilinx Virtex-5 FPGA device applied in human skin classification (image recognition area) (Pietron, et al. 2013). They compared such fully pipeline architecture with the same algorithm applied in GPU (Nvidia Tesla m2090) and CPU, where it was possible to observe that FPGA implementation was four times faster and consumed 83% of LUTs at 200MHz - operating frequency.

Afifi, Hosseini, and Sinha proposed a cascade SVM classifier on FPGA in an embedded diagnosis system of melanoma (Afifi, Hosseini and Sinha 2018). They obtained good results such as accuracies of 73% to 98%, utilization of 1% slices (circuitry area), and power consumption of 1.5W in the "Xilinx Zynq-7000 All Programmable System on Chip (SoC)" device.

Some related works from literature are analyzed, so the uniqueness of SVM hardware application can be analyzed, and then, their results can be studied.

Low-Power Hardware Implementation on FPGA of SVM for Neural Seizure Detection

A system to detect neural seizures with low-power consumption and minimum area using SVM and FPGA prototyping is proposed by (Elhosary, et al. 2019). In this proposal, the following five main steps are done:

1. Electroencephalography (EEG) signals are measured, and techniques to improve their measurement efficiency are studied;
2. Signal preprocessing;
3. Feature extraction;
4. The training phase is then executed with these extracted features to create a hyperplane that separates two labeled sets of training examples, where the Sequential Minimal Optimization (SMO) algorithm was applied;
5. A classifier is used with the extracted features to detect seizures and to classify unlabeled testing examples into one of the two classes.

Besides, the proposed work performs the hardware implementation of the SVM (training and classification phases) using FPGA (Xilinx Virtex-7 FPGA) and ASIC implementations, using the Polynomial and Linear kernel functions. A 16-bit word length fixed-point implementation is used instead of the computationally expensive floating-point. The classifier output is generated after 16 clock cycles after receiving the input feature vector. Their FPGA implementation results of the classification phase were about 238 LUT slices, 137 register slices, and 6mW of power consumption.

A Cascade SVM Classifier on FPGA for Early Melanoma Detection

In (Afifi, Hosseini and Sinha 2018), the authors proposed an embedded system to enhance early detection of melanoma at primary healthcare using image-based diagnostic tools. They aimed recent hardware technology in reconfigurable computing of low cost implementing a cascade SVM classifier on FPGA. The FPGA device used was the Xilinx Zynq-7000 All Programmable System on Chip (SoC), where an SVM algorithm was designed to be written in C/C++ language using the UltraFast High-Level Synthesis (HLS) tool to implement an SVM model. In the Vivado HLS tool using C/C++ language, a top function module is designed as an HLS IP that basically computes the decision linear kernel function to be used for online classification on Zynq SoC. Then, this work is about a scalable multi-core architecture that consists of N SVM's cores/IPs in a single system on chip/device. A dual-core architecture (n = 2) is proposed as their case study to be employed as a two-stage cascade SVM classification architecture. The system obtained 98% and 73% accuracy, utilization of 1% slices (circuitry area), and power consumption of 1.5W.

Multi-Class SVM Classification for Epilepsy and Epileptic Seizure Detection on VLSI Technology

A system of epilepsy and seizure detection is elaborated in this paper, where a Very Large-Scale Integration (VLSI) architecture of three-class classification is proposed (Wang, et al. 2018). The authors used a Discrete Wavelet Transform (DWT)-based feature extraction module and the One-Against-One (OAO), or One versus One technique, multi-class Non-linear SVM (NSVM) to train and classify the Electroencephalogram (EEG) signal features. The designed system was implemented on a Field-Programmable Gate Array (FPGA) platform and evaluated using a publicly available epilepsy dataset. They implemented several different ways of how executing the SVM algorithm in software so they could find the best way to implement it in hardware. The kernel function used for the multi-class SVM classification in hardware was the Gaussian Radial Basis Function (RBF). In order to validate the circuit design based on the proposed architecture, the circuit was implemented on the Xilinx Virtex 5 FPGA XC5VLX110T using the ISE13.3 platform. The SVM classifier's processing time depends on the number of iterations and the feature vectors' dimensionality. Therefore, the iteration number can be set to 400 where it was observed that the classification accuracy tended to saturate, at which the average accuracy is 94.2%, and 686,675,325 clock cycles are taken. The classification accuracy is 95.8% in the software implementation, while the hardware implementation achieves 94.2%. Besides, the clock cycles are computed in software implementation, and 1,695,727,388 clock cycles are taken. Therefore, the system implemented in hardware is faster than that implemented in software. About the resources used in the FPGA device, it was used 71% of LUT slices. It is essential to highlight that the authors' contribution is about, to the best of their knowledge, the first work to implement a three-class SVM classifier in hardware with on-chip training capability.

CNN-SVM Hybrid Classifier in Hardware Implementation on FPGA

In (Gowda and Rashed 2017), the authors proposed a hybrid classifier to detect cancer cells in the Acute Lymphoblastic Leukemia dataset. There are images (size of 128x128 pixels) of leukemia and melanoma affected cancer cells. Convolutional Neural Networks are used to extract features and train the data, and the Support Vector Machine performs the classification process. The trained features of convolutional neural network block, processed by a processing unit of Zynq SoC FPGA, are fed to input to a support vector machine block. This block consists of multiplexers, registers, comparators, and adders/subtractors. This proposal classifies up to four classes: leukemia negative, leukemia positive, melanoma negative, and melanoma positive. Besides, it was used 5.6% of LUTs and consumed 1.63W of dynamic power, with the operating frequency of 100MHz.

State-of-the-Art Review of SVM Hardware Implementations in FPGA

In (Afifi, Hosseini and Sinha 2015), we can find several SVM hardware applications, 53 current conferences, and journal articles published from 2010 to 2015. Those applications are divided into three groups. The first group (12 papers) is about FPGA-based implementations of the SVM training phase, the second one (28 papers) is about implementations of the classification phase. Moreover, the last group (13 papers) is about application-specific implementations that use SVM hardware implementation as part of a larger algorithm.

The authors cited a statement in (Nayak, Naik and Behera 2015), a survey that shows SVM in data mining tasks. Such a statement says that there are considerable limitations of implementing the SVM model, such as processing speed and large memory space requirements. That is why there is a need for efficient SVM implementations. This necessity was their motivation to survey FPGA-based hardware implementations of SVM for various embedded applications.

Then, from the first group of this state-of-the-art review, it is possible to observe that various parallel digital designs were presented where the common pipelining approach was mostly applied, reaching a high level of parallelization. Additionally, some studies employed the dynamic reconfiguration technique for the designs, aiming speed improvement with more flexibility. Moreover, dynamic scheduling was exploited for efficient reconfiguration. Moreover, the hardware friendly kernel was used as well and implemented by a standard CORDIC algorithm based on shifters and adders replacing costive multipliers, which led to a remarkable reduction in resource utilization. Many proposed implementations for the SVM training phase achieved significant speedup results, which outperformed similar software implementations, where some reached acceptable accuracy with slightly rate loss. On the other hand, some hardware implementations gained remarkable speedup improvement compared to previous hardware designs. Furthermore, great area saving results were realized. An improvement in power consumption was shown about 77.3mW in (Peng, et al. 2014) and 6W in (Rabieah and Bouganis 2015), which could meet some significant embedded systems constraints.

From the second group, M. Ruiz-Llata et al. (Ruiz-Llata, Guarnizo and Yébenes-Calvino 2010) presented an FPGA-based hardware design of SVM for classification as well as regression (regression out of our scope). The proposed system of (Anguita, et al. 2006) adopted the hardware friendly kernel function, which significantly simplifies the hardware design of the feed-forward SVM classification phase by avoiding computationally intensive multiplications, providing excellent classification performance compared to the traditional Gaussian kernel. The proposed architecture employed the CORDIC iterative algorithm (Andraka 1998), based on only shift and add operations instead of multiplications required by the kernel computation. The implemented SVM classification system utilized 75% of the FPGA logic (Cyclone II), and external memory was used for storing support vectors leading to 2ms limitation in the classification speed, with an error rate of 4%.

Besides, the presented architecture proposed in (Kyrkou, et al. 2015) was implemented on Spartan-6 FPGA (replacing the old one used before), targeting embedded face detection using the higher resolution of 800x600 images than that in their previous work and other hardware implementations. The implemented hybrid architecture achieved real-time processing of 40 fps with 80% detection accuracy, 25% and 20% reduction in area, and peak power, respectively, with only a 1% reduction in classification accuracy. However, compared to their previous work, it seems that better figures were achieved for both area and accuracy. This fact occurred because they evaluate a big test set of higher resolution images, targeting real-time processing of online video classification as an embedded benchmark application.

Finally, from the third group, for example, an object detection processor was presented based on a proposed simplified HOG algorithm, which employed a simultaneous SVM calculation, targeting a reduction in the number of required computations (Mizuno, et al. 2013). The proposed simultaneous SVM calculation architecture was designed as 15 parallel classification cores, where each managed seven blocks of MAC operations and allowed the reuse of intermediate results. Accordingly, experimental results showed a 99% reduction of cycle counts for the proposed SVM module compared to architecture without parallelization and pipeline. The proposed system achieved object detection for SVGA resolution video at 72 fps with 40MHz. Besides, the proposed simplified algorithm reduced required computation (from 89.2 to 2.25 GOPS) with minimal memory usage and a 3% decrease in success accuracy.

Combinatorial Circuit Application of SVM in FPGA Implementation

In (Song, Wang and Wang 2014), the authors implemented a linear and non-linear SVM classifier for two classes, an FPGA-based system. Their approach is about the Support Vectors, Lagrange multipliers application, and the bias from the training phase into the FPGA internal memories. All that information is then processed according to the SVM mathematical modeling, using the combinatorial circuit technique. In the non-linear case, the exponential calculation of the Gaussian radial basis kernel function was made through a table-driven module in the FPGA internal memory, where $\sigma = 0.6$. Also, the signed fixed-point binary data format of 18 bits (2 bit before the decimal point and 14 bit after the decimal point) was applied instead of floating-point format. The targeted device for the proposed architecture was the Altera’s Cyclone II EP2C70F896. The targeted operating frequency is between 200 to 250MHz for the testing data; it was created four random sampling datasets for linear and non-linear SVM classifier separately. Each dataset's size is 400, 20 of which will be used for SVM training, with a testing size of 50. The recognition rate was between 97.5% and 100%; the recognition was about Smart Grid.

However, it is not automatic because its data entrance is made manually, neither optimized because its training data are all from the data preprocessing part; then, it took much time to do the SVM training phase. It is a combinational circuit that costs less power consumption but worse response time. Besides, it has data from training to classify just two classes.

Blocks of Combinatorial Circuits with an FSM of SVM in FPGA Implementation

Many types of research have been developed for SVM hardware implementations; one of them is about an ASIC chip design for Sequential Minimal Optimization (SMO) algorithm used to train SVM, proposed by (Kuan, Wang and Wang 2012). This chip was suitable for the use of an SVM-based recognition system of speech parameters. In order to test it, the system was used in a prototype on the Altera DE2 board with a Cyclone II 2C70 field-programmable gate array and synthesized by Verilog-HDL descriptions. The authors used a fixed-point design, a Finite State Machine (FSM), and blocks of combinatorial circuits in the datapath composing their architecture. The training phase was about 200 to 400 iterations, where the recognition accuracy tends to saturate and lies in its highest value. Their result tests (classification phase) were then obtained in 6 seconds for 400 iterations in training using Linear kernel function. I.e., 15ms for each iteration in average; the average accuracy of success in recognition was 93.4%; the used LUTs were 6,395 of 68,410 in total; it was used 58 embedded multipliers (18x18 bits) of 150 in total; the operating frequency was 50MHz.

SYNCHRONOUS HARDWARE ARCHITECTURES OF SVM CLASSIFIER APPLIED IN AN ASR SYSTEM

As it is known, most of the processing in the learning of the ML concept includes two phases, in general: training and classification phase. In the training phase, patterns are found for each class, and then, they are compared to the test data in the classification phase.

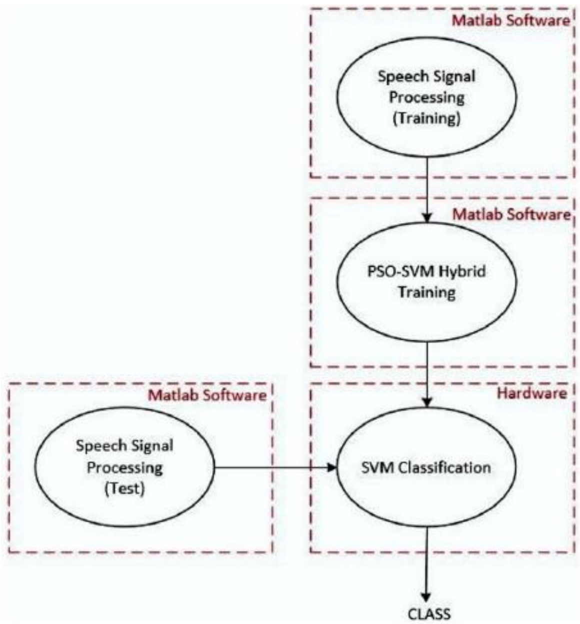


Figure 1. Schema of the complete ASR system.

For a real scenery case, we have an ASR system (Batista, Silva and Oliveira, et al. 2019), shown in Figure 1. There are four blocks where three operations are made, in general, as follows: Speech Signal Processing (Test and Training), PSO-SVM Hybrid Training, and SVM classification. The only part applied in hardware was the SVM classification phase, being an asynchronous implementation. The result is binary, '1' if what was spoken belongs to the class that has been analyzed, and it is '0' if it does not belong. Then, the classification is made using the “One-vs-All” technique of SVM, i.e., one class is selected from the rest of the data (other classes). The total of classes that are classified is thirty in this work, being 10 numeric digits and 20 voice commands, in the Brazilian Portuguese language: “zero” (zero), “um” (one), “dois” (two), ``três” (three), “quatro” (four), “cinco” (five), “seis” (six), “sete” (seven), “oito” (eight), “nove” (nine), “abaixo” (below), “abrir” (open), “acima” (above), “aumentar” (increase), “desligar” (turn off), “diminuir” (decrease), “direita” (right), “esquerda” (left), “fechar” (close), “finalizar” (finish), “iniciar” (start), “ligar” (turn on), “máximo” (maximum), “mínimo” (minimum), “para atrás” (go back), “para frente” (go ahead), “parar” (stop), “repousar” (pause), “salvar” (save).

Then, according to Figure 1, the first stage of the ASR is about speech signal processing, which is the subject addressed further down.

Speech Signal Processing

Before explaining the training phase and the classification phase's architectures, it is necessary to understand how the training phase's voice features were obtained.

The speech signal processing is the same one used in (Batista, Silva and Oliveira, et al. 2019). The step-by-step algorithm of the voice signal parameters extraction is shown below:

1. Application of Hamming windowing;
2. Estimation of the Mel-frequency Cepstral Coefficients (MFCC):
   a. Cepstral analysis;
   b. Calculation of the mel scale according to the sampling frequency;
   c. Output processing of log-energy for the triangle filters already spaced according to the mel scale;
3. Discrete Cosine Transform (DCT) application in order to reduce the MFCC data without information loss;

4. Organization and establishment of the CT_{ix2} matrices (matrices with the MFCC parameters reduced by DCT).

Training Phase: PSO-SVM Hybrid Algorithm

Particle Swarm Optimization (PSO) is a bio-inspired algorithm where the goal is to find the best position of a data population (Lazinica 2009). In this case, this best position is memorized and used to represent its respective dataset.

About SVM, the optimal hyperplane (decision surface) obeys to the following function: $(\Omega(x)^T w + b) = 0$, as shown in Figure 2; where x is the data to be separated; $\Omega(x)$ is the kernel function applied to data; w is a vector of weight values; and b is a bias value (Scholkopf, et al. 1997).

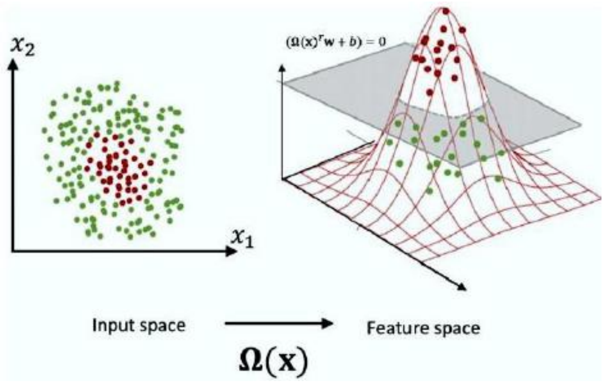


Figure 2. Transformation to the feature space of high dimensionality using the Kernel function of the SVM algorithm.

The most used Kernel functions in the SVM algorithm are:

$$\Omega(X_i,\, X_j) = \begin{cases} X_i \cdot X_j \; Linear \\ (\|X_i^T \cdot X_j\| + d)^p \; Polynomial \\ \exp\left(-\|X_i - X_j\|^2 / 2\sigma^2\right) \; Gaussian \\ tanh\left(a\|X_i^T \cdot X_j\| + d\right) \; Sigmoid \end{cases}$$

where $\boldsymbol{x} = [X_i; X_j]$ is a matrix of N lines by two columns (columns i and j) related to the two-dimensional application of SVM. Column i is about the data X_1 and column j about the data X_2 (see Figure 2). The *Sigmoid* kernel function is also called Multilayer Perceptron (MLP) kernel function, and the *Gaussian* function is also called Radial Basis Function (RBF) kernel.

Thus, PSO is applied before the SVM training phase to reduce data amount without loss of information and solve SVM's load computational and time-consuming problems. Furthermore, SVM training is executed, as shown in Figure 3, finding the best separation between two classes in the CT matrix of the Nx2 dimension. As explained before, this PSO-SVM hybrid training algorithm was proposed by (Batista, Silva and Oliveira, et al. 2019).

The type of learning technique covered in this system is supervised, i.e., there are two matrices in the input of the SVM algorithm: a matrix with real data and a matrix with desired response data. Besides, the learning is not noisy; that is, there is no noise in the desired response matrix. In SVM's kernel function application, the RBF function was chosen to be applied with variations of its σ value. This function was chosen because it was proven to be the best one in the work of (Batista and Silva, Using Support Vector Machines and two-dimensional discrete cosine transform in speech automatic recognition 2015).

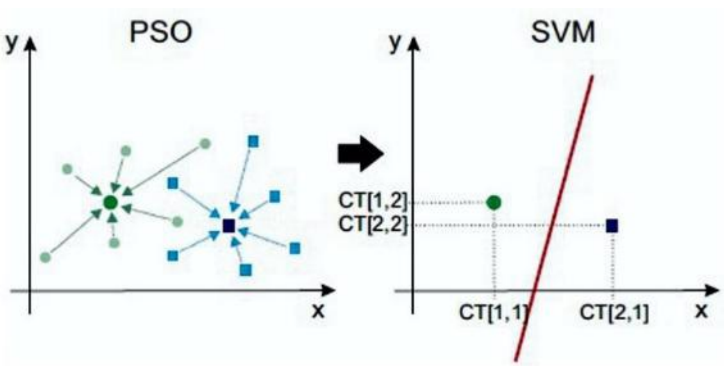


Figure 3. Illustration of the PSO-SVM hybrid training.

Classification Phase: Synchronous and Asynchronous Hardware Architectures

Before explaining the architectures, it is necessary to understand where SVM's training data in the classification phase is generated. It was then explained that SVM classifies through an optimal hyperplane finding the best separation between two classes. Such hyperplane obeys to Equation 1 in feature space.

$$w \cdot x + b = 0 \tag{1}$$

where x is the input data, b is the bias value, and the weights w for the optimal hyperplane in the feature space can be written as some linear combination of support vectors, as shown in Equation 2.

$$w = \sum_{support\ vectors}^{N} \alpha_n SV_n \tag{2}$$

where *n = 1,2,...,30*, α_n are the Lagrange multipliers, and SV_n are the support vectors (Cortes and Vapnik 1995).

For the development of SVM hardware implementation, it is necessary to have a datapath that follows the block diagram of SVM mathematical modeling using the Radial Basis Function (RBF) as its kernel function of σ in the "EXP" block of Figure 4. From the block diagram, we can observe that entrance data are: *Test Data* from speech signal processing; SV_N (Support Vector matrix where there are two support vectors to each class), α_N, and b (bias) from the training phase. Besides, in this work, the Support Vector matrix has *N=30*, and alpha matrices go to α_{30} (each matrix has 30 alpha values) because there are 30 classes to be separated/classified. The α values in the Alpha matrices are found through the Lagrange multipliers calculation, and they are used in the optimal hyperplane equation of the SVM. The 30 *bias* values (one per class) are from the optimal hyperplane equation, all data from the PSO-SVM hybrid training phase.

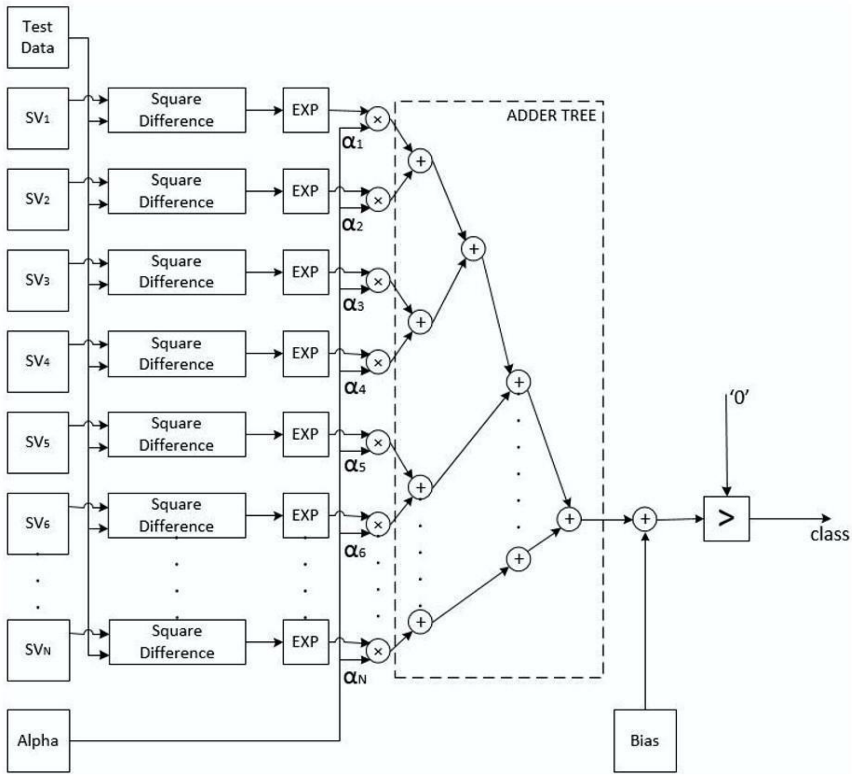


Figure 4. Block diagram of mathematical modeling of the SVM classification phase.

As we are working with an exponential calculation with high curvature because of the low value of σ, it was necessary to use the *ieee_proposed.fixed_pkg.all* package, which converts the floating-point to 18-bit fixed-point format (Bichop 2005). In this 18-bit fixed-point format, one first bit is used to represent the signal ('0' if negative or '1' if positive) of the real number, and the other 17 bits are used to represent the decimal part of the same real number. Then, the result calculations have sufficient precision compared to doing them in the floating-point format, as it is done in software.

The EXP() component was made from a look-up table obeying Equation 3, which originated from the RBF Kernel function.

$$EXP\left(\frac{-(SV-Test)^2}{2\times\sigma^2}\right) \tag{3}$$

where σ is the variable of RBF kernel function, SV is the Support Vector value, and $Test$ is the value from the voicebank test's speech signal parameters.

Once we know about the training phase process, the SVM classification phase's synchronous hardware architectures can now be explained.

Architecture I: Combinatorial Circuit

In reason of understanding how to implement the SVM classifier algorithm in hardware, a first and straightforward architecture was developed, a combinatorial circuit of the SVM classifier. Then, the flowchart of Figure 5 was generated from the block diagram of Figure 4.

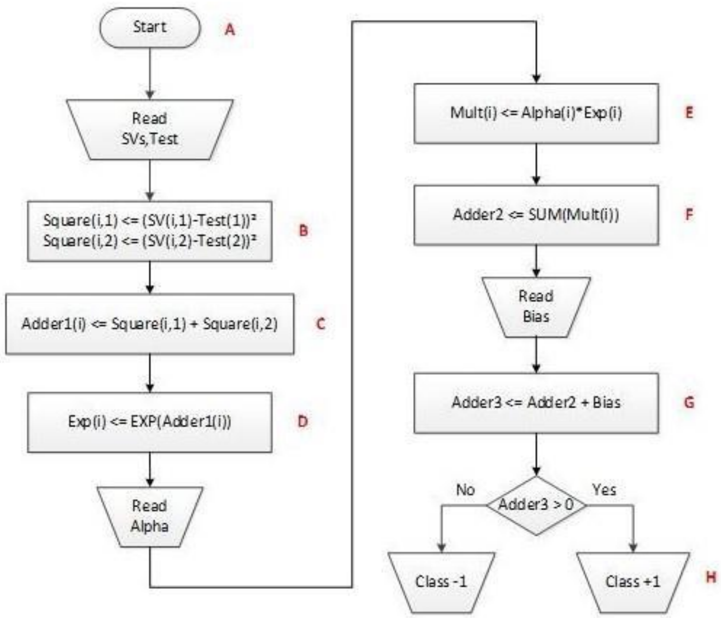


Figure 5. Flowchart of the SVM classifier.

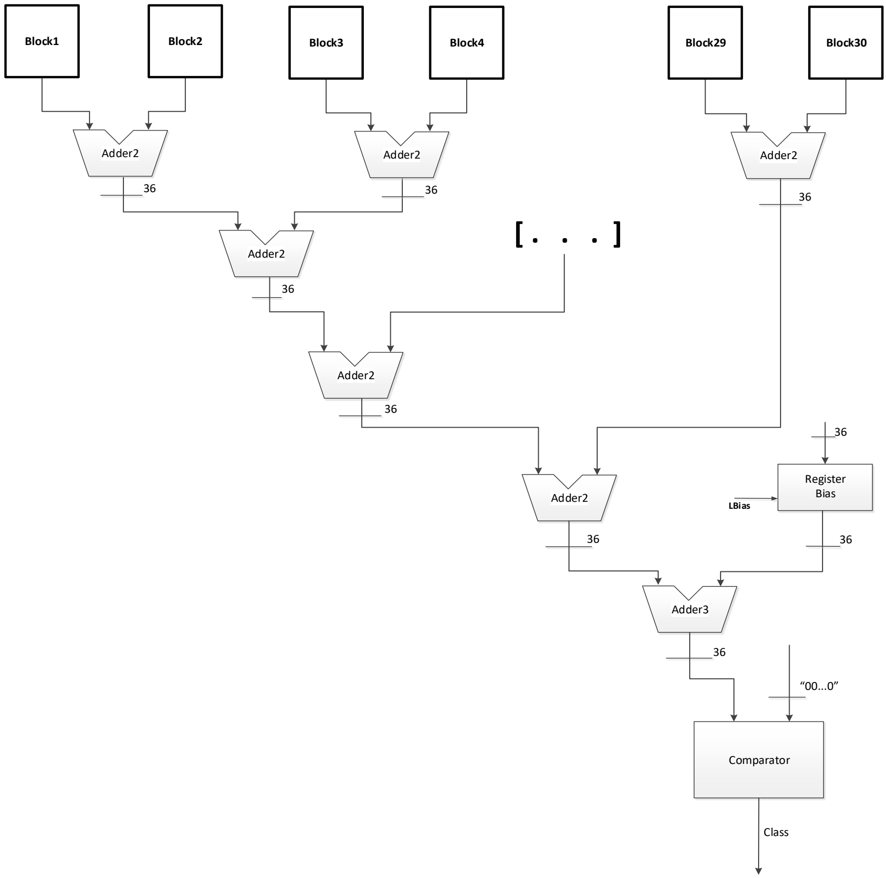


Figure 6. Datapath of the SVM classifier described in VHDL.

In order to do the VHDL description, the general datapath idea of Figure 6 was generated for 30 classes, according to the flowchart of Figure 5. *Block1*, *Block2*, *Block3*, *Block4*, …, *Block30* are related to the data entrance and preprocessing. There are 30 blocks because each block is about the process of each class data.

Figure 7 shows the Block1 of Figure 6, where there are the components: SQUARE, ADDER1, EXP, MULTIPLIER, ADDER2, ADDER3, and COMPARATOR. Besides, there are the *SVs* and *Test* matrices data entrances on the upside. On the bottom, there is the *Alpha* matrix entrance (Lagrange multiplier coefficients); all of these matrices are originated from the training phase compiled in Matlab software.

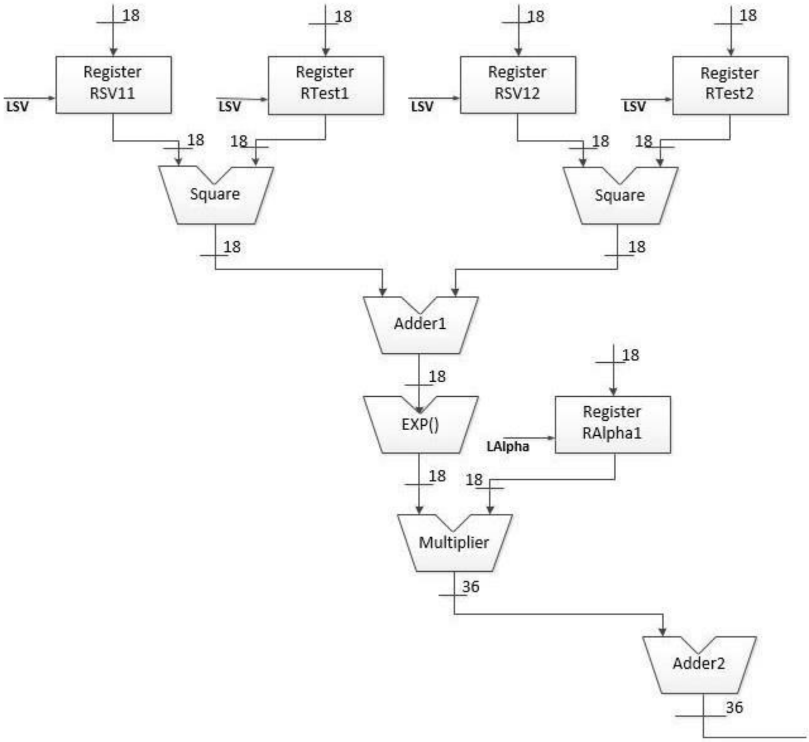


Figure 7. Block1 of the datapath of the VHDL description of the SVM classifier.

The classifications were made two by two (using “One Against All” - OAO technique) manually. The data for the classification phase is stored as follows. SV (Support Vector) data were stored in the FPGA device. Test Data were external input data. Thirteen Alpha values and one Bias value for each class were external input data stored in the FPGA device for each classification process.

Figure 8 shows the time table of the elaborated Architecture I (applied in a combinatorial circuit). In this time table, it is possible to observe the *clk* (clock), *rst* (reset), *Inicio* (start), *Final_Res* (final result of classification), *x1* (data from column 1 of the *Test* matrix), and x2 (data from column 2 of the *Test* matrix) signals.

Table 1 shows the result implementation of the SVM classifier implemented in the Intel-ALTERA Cyclone II and IV devices, where the Cyclone IV allowed a better performance in the power consumption, resulting in a dynamic power reduction of 55.3%. The accuracy in the recognition success rate of this implementation was 98.5 up to 100%.

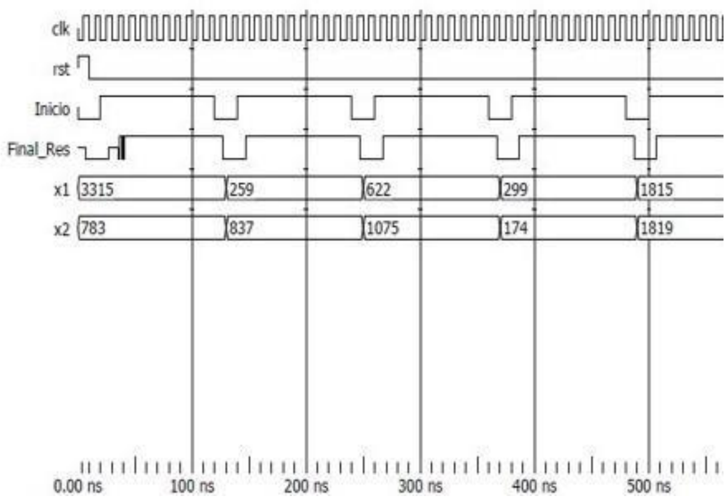


Figure 8. Timetable from post layout of the elaborated architecture I.

Table 1. Results of the combinatorial architecture of SVM classifier datapath implemented in Intel-ALTERA's devices configuration

Features:	**Cyclone IV**	**Cyclone II**
Operating Frequency	150MHz	110MHz
LUTs	3,286 / 114,480	3,287 / 68,410
Dynamic Power Dissipation	28.25mW	51.59mW

Architecture II: Pipeline Datapath Controlled by FSM

This architecture is about a pipelined datapath, including a Multiplier-Accumulator (MAC) unit instead of the Adder Tree (see Figure 4) controlled by a Finite State Machine (FSM) (Batista, Oliveira and Silva, et al. 2019). MAC unit uses a multiplier and an accumulator; it is very used in Digital Signal Processing (DSP). Equation 4 shows what this unit calculates, and Figure 9 shows a block diagram of its operation.

$$Z = Z + (X \times Y) \tag{4}$$

where X and Y are the input values of the MAC unit, and Z is the output.

In the pipelined datapath, all registers receive a clock signal from the circuit of Figure 10, i.e., registers do not have clock activity if there is no *Load* signal in the circuit entrance.

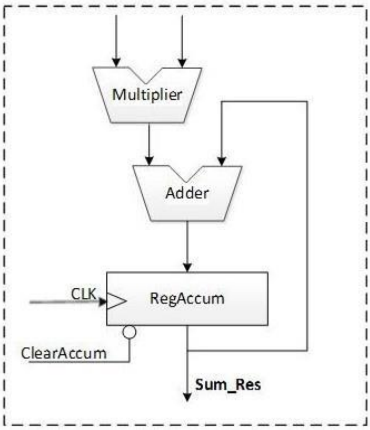


Figure 9. MAC unit of the pipelined datapath.

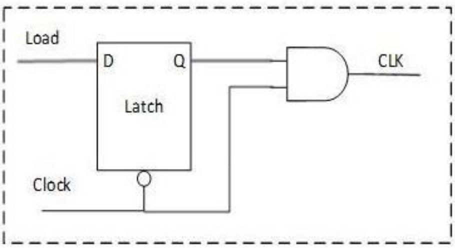


Figure 10. Circuit of the clock signal for the registers of pipeline datapath.

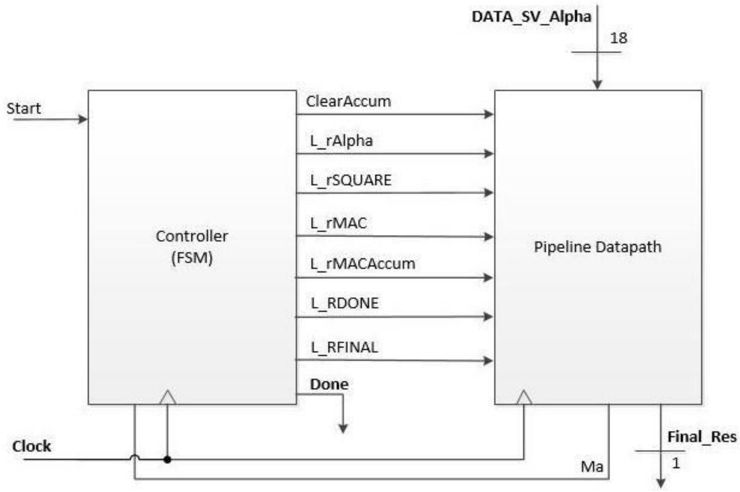


Figure 11. The architecture (FSM with pipeline datapath) for SVM classifier.

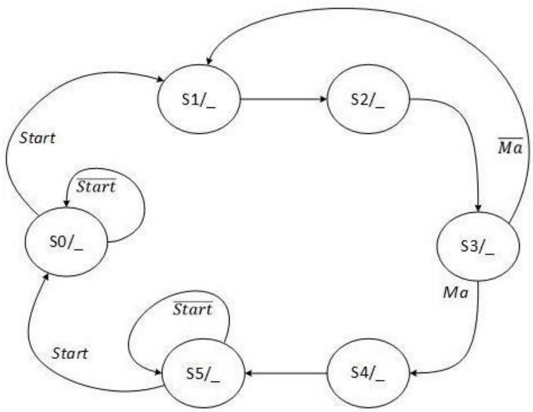


Figure 12. FSM specification for control of the architecture (see Figure 10).

Then, all *Load* signals of registers and the *clear* signal of the MAC unit's accumulator are controlled by an FSM, shown in Figure 11.

The FSM specification is shown in Figure 12, where *Ma* is the signal describing if all *Data_SV_Alpha* data (training data of 18 bits) were processed. Each stage of the FSM is related to a stage of the pipelined datapath, shown in Table 2.

Table 2. Stages and output variables of the FSM controller

Output Stages	**L_RDone**	**L_rMACAccum**	**L_rMAC**	**L_RFinal**	**L_rSquare**	**L_rAlpha**	**L_clearAccum**
S0 – Beginning	0	0	0	0	0	0	1
S1 – Square	0	0	0	0	0	1	0
S2 – EXP	0	0	0	0	1	1	0
S3 – MAC	0	0	1	1	1	1	0
S4 – Comparator	0	1	0	0	0	0	0
S5 – Final	1	0	0	0	0	0	0

Through Table 2, it is possible to observe the FSM behavior according to its control output signals, shown in Figure 10. In order to analyze the datapath operation stages, Figure 5 shows its flowchart. The first stage has subtractor and multiplier operators related to the square operation. The second one has an adder operator and a look-up table (exponential operation - EXP). The last has the MAC unit. Finally, the fourth stage has adder and comparator operators. The datapath entrance has data of 18 bits from the training phase of the PSO-SVM hybrid algorithm (obtained from software compilation using Matlab software). The *RegDone* is a register which receives data just when the MAC unit is done, i.e., when all training data is entered and processed in previous stages. Extracting the state diagram from the flowchart of Figure 5, we have the machine states. States S0 and S5 are, respectively, the beginning and the end of processing. State S1 performs the square operation. State S2 is related to the adder, exp, and multiplier operations. In S3, the adder operation is done. Finally, state S4 performs the comparator operation. Figure 12 and Table 2 show those states.

In Figure 13, it is possible to observe the time table of the elaborated architecture using the FSM controller with pipelined datapath where *clk* is the clock signal, *Start* is the signal to control the beginning and final stages of the FSM, *rst* is the reset signal, *X1* and *X2* are the test data, *Result* is the classification result (the same *Final_Res* signal from Figure 11) and *Done* is the output signal from FSM controller meaning the process end (see Figure 11).

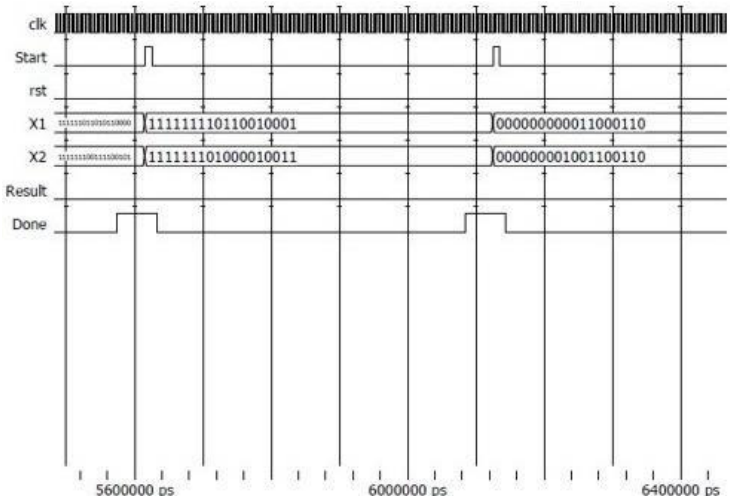


Figure 13. Timetable from the post-layout simulation of the elaborated Architecture II.

Table 3. Results of the pipelined datapath controlled by FSM of the SVM classifier implemented in Cyclone IV Intel-ALTERA's device

Features	**Architecture II**
Sigma value (RBF kernel function)	0.03
Operating Frequency	20MHz
Throughput	20MOPS
LUTs	1,282 / 114,480
Registers	302
Accuracy	98.5 - 100%
Dynamic Power Dissipation	41.52mW
Embedded Multiplier 9-bit elements	280 / 532

Concerning to this architecture, Table 3 shows the sigma value applied in the Gaussian (RBF) kernel function and the results of: used LUTs, dynamic power dissipation (power consumption), used registers, operating frequency, throughout, recognition success rate, and embedded multiplier elements.

Architecture III: Linear Pipeline with Synchronous Control

The elaborated basic pipeline architecture insensitive to latency is composed of processing modules, shown in Figure 14 (Batista, Oliveira, et al. New Approach of Pipelined Architecture of SVM Classifier for ASR System in FPGA Implementation 2019). In this architecture, there are registers based on D flip-flop. The control in each pipeline register has two types in its configuration. The first *Valid-i* signal is triggered by *Inicio* (meaning Start) signal.

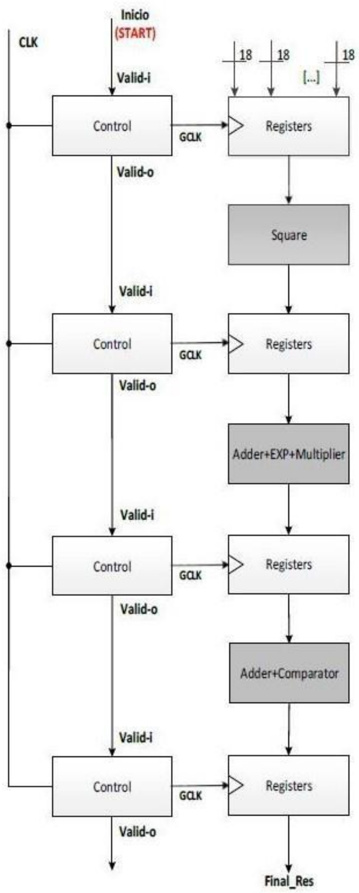


Figure 14. A basic synchronous linear pipeline does storage data only when there is a valid signal in the first control circuit regardless of the clock cycles.

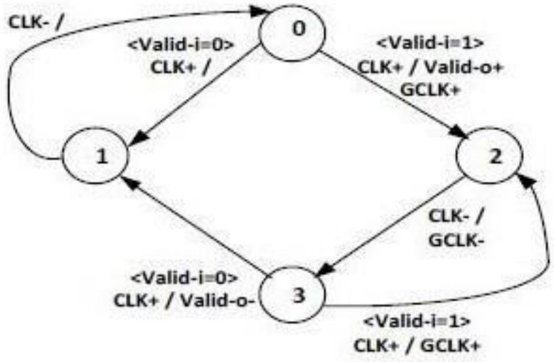


Figure 15. XBM specification: control for the elastic synchronous pipeline with only one valid signal to control *GCLK* output.

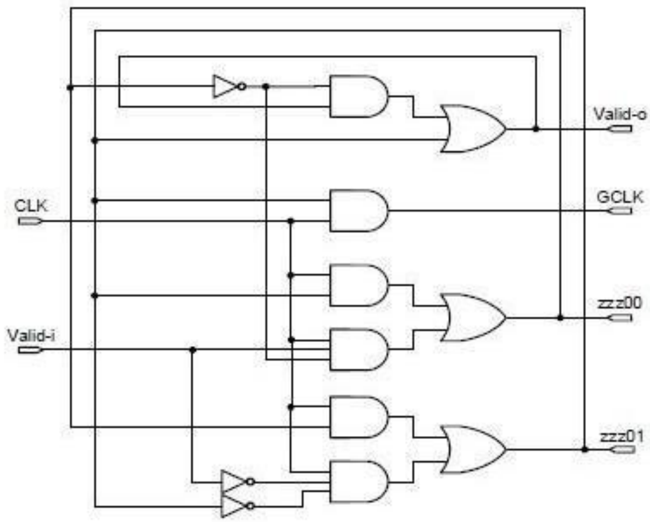


Figure 16. Gated-Clock XBM control - logic circuit.

The first configuration is about controller circuits to perform synchronization between the global clock signal and a signal sensitive to *the Valid-i* signal level in the Extended Burst Mode (XBM) specification, shown in Figure 15. The following signal compose it: *Valid-i* (valid data at the input stage - request-store), *Valid-o* (data valid for the next stage, a signal to ask for requesting to store), *CLK* (clock) signal, and *GCLK* (gated-clock signal) signal controlled by valid data in gated stores. Then, Figure 16 shows the asynchronous control circuit's logic circuit that it was synthesized by the 3D tool (Goldberg 1989).

Finally, the second configuration of control is a version based on transparent latches, which control is shown with its logic circuit in Figure 17.

Figure 18a shows the elaborated pipeline architecture timetable with the control circuit based on the transparent latches, post-layout simulated in Intel-ALTERA's Cyclone IV device. Through this figure, it is possible to observe the *clk* (clock signal), *rst* (reset signal), *Inicio* (start signal), *Final_Res* (the classification result), *x1* (data from column 1 of the Test matrix), and *x2* (data from column 2 of the Test matrix) signals.

Figure 18b shows the pipeline architecture timetable; however, with a control circuit based on XBM specification post-layout simulated at the same device as previously and, *Valid-i* signal of XBM specification obeying to *Inicio* signal behavior.

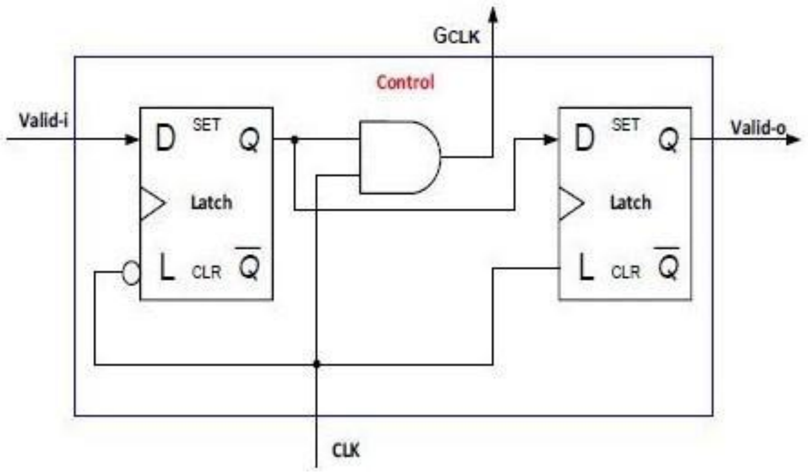
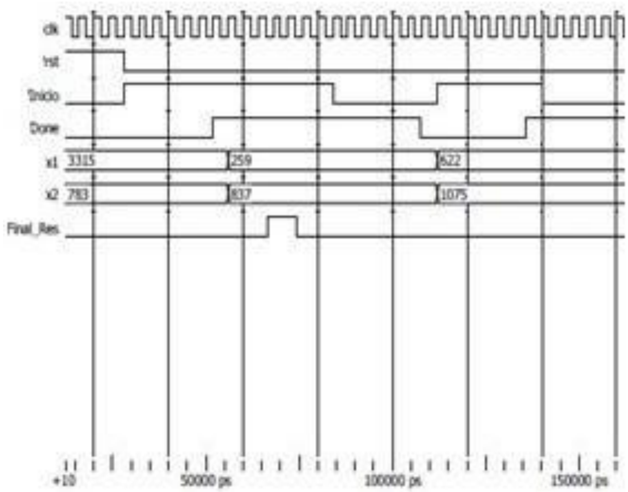
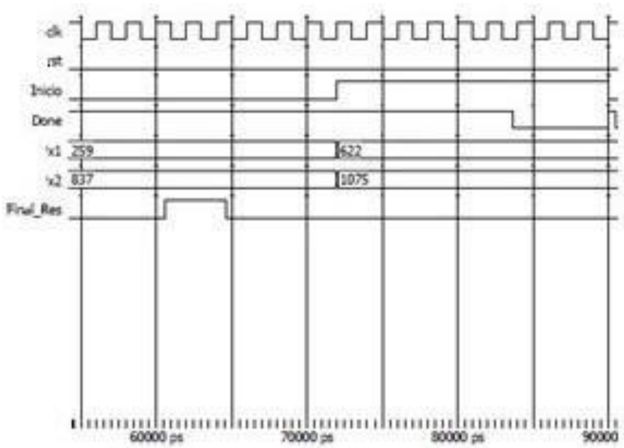


Figure 17. Gated-Clock control using transparent latches - logic circuit.



(a) Based on transparent latches



(b) Based on XBM specification

Figure 18. Timetables from post layout of the architecture III post layout on the same device.

Table 4 lists the post-layout simulation results with both control circuits application from the lowest circuit configuration. It is considered the worst circuit conditions like critical path and highest temperature. It is then possible to observe the following parameters: throughput, LUTs, used registers, recognition success rate, power consumption, operating frequency, and embedded multiplier elements.

Table 4. Results of the linear pipeline architecture controlled by synchronous circuit: with XBM specification and using transparent Latches

Features	**XBM**	**Latches**
Sigma value (RBF kernel function)	0.09	0.09
Operating Frequency	500MHz	250MHz
Throughput	50MOPS	500MOPS
LUTs	5,387 / 114,480	6,198 / 114,480
Registers	1,747	1,747
Accuracy	98.5 - 100%	98.5 - 100%
Dynamic Power Dissipation	174.15mW	224.92mW
Embedded Multiplier 9-bit elements	420 / 532	420 / 532

Architecture IV: Asynchronous Hardware Architecture of SVM Classifier Applied in an ASR System

First, some crucial concepts about the asynchronous paradigm are given below to understand how the SVM classifier is implemented within this area. Then, the novel asynchronous hardware architecture of the SVM classifier is explained later. It is essential to highlight that these four architectures were applied in the same ASR system approached previously.

Asynchronous Paradigm: Some Concepts

Asynchronous circuits can be synthesized in different classes (Myers 2004). The asynchronous pipeline class allows obtaining high performance due to its parallel processing nature, which is a speech recognition application. The bundled-data asynchronous pipeline style is attractive because it is composed of conventional data, i.e., it uses functional units of synchronous paradigm and the asynchronous control responsible for communication between pipeline stages (Myers 2004). The applied registers in most literature architectures are based on transparent D latches, and the critical path defines the delay elements. Figure 19 shows an asynchronous bundled-data linear pipeline architecture proposed by

(Oliveira, Garcia and D'Amore 2014). This style uses just components from the synchronous paradigm and uses delay elements between stages. It represents N bits of data with N+2 lines (called "bundled"), where the two additional lines are the request and signals of acknowledging (according to the handshaking protocol).

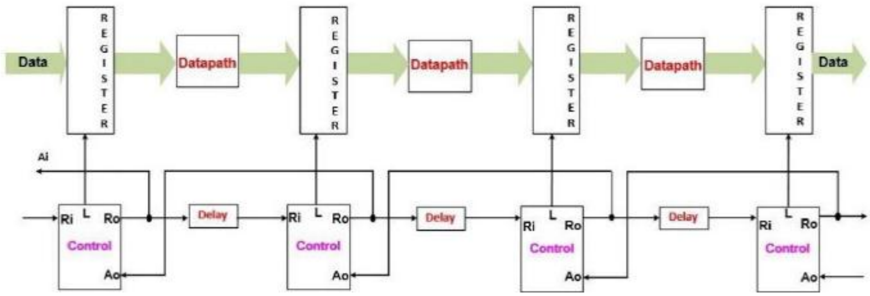


Figure 19. The basic structure of the asynchronous linear pipeline.

Asynchronous controllers perform communication and synchronization between pipeline stages, operating in a handshaking protocol. Two important handshaking protocols are 4-phase and 2-phase. The 4-phase protocol behaves in the following way: when the request signal goes from 0 to 1 means that there is data being sent, and when the acknowledge signal goes from 0 to 1 means that the data was received. To start a new data transfer, the request and acknowledge signals, firstly, must be zero. In the 2-phase protocol, the sent data and accepted data operations are performed on both transitions of the request and acknowledge signals, i.e., from 0 to 1 and 1 to 0.

The specifications more important to describe asynchronous controllers are Signal Transition Graph (STG) and Extended Burst-Mode (XBM) because they allow an optimized synthesis.

STG specification was proposed by (T. A. Chu 1987). It is a Petri-net description with the properties: free-choice, safeness, liveness, and output-persistent (T. A. Chu 1987). The STG is used to describe concurrence between inputs and outputs (I/O concurrence) and the sequencing of signals in asynchronous systems. The STG naturally describes the timing diagram that is quite used in the interface design, i.e., it can follow the diagram behavior according to its time precisely.

Burt-Mode (BM) specification proposed in the Hewlett-Packard (HP) enterprise and formalized by (Nowick 1995). It is represented by a diagram of state transitions, where transitions can occur for single or multiple input changes. It is necessary to define the initial state. An arc represents each state transition in the diagram, and it is labeled by a set of input signals and output signals. These sets are called input/output burst, where these input burst signals are activated in any order and any time. The signals are always TSS ($0 \rightarrow 1$ or $1 \rightarrow 0$). If any change occurs in the inputs, the machine stays in the same state. The burst inputs are monotonic, changing only once during each state transition. It is essential to know that BM specification has three polarity restrictions, a unique entry point, and maximal set property (Nowick 1995).

The Extended Burst-Mode (XBM) specification was proposed by (Yun and Dill 1999). It inherits all the characteristics of BM specifications. It incorporates the conditional signals, which depend on Level Sensitive Signals (LSS) with non-monotonic behavior and directed don't-care signals, which allow an input signal to change concurrently with output signals. LSS signal can freely switch if it is not labeled by a state transition (non-monotonic behavior). Otherwise, it has to meet the setup and hold time requirements. On the other hand, the direct don't-care signals are applied only on monotonic transitions. Every state transition in XBM presents a so-called compulsory signal; in the previous transition, there is no directed don't-care signal.

Two main approaches to design asynchronous controllers are logic synthesis (T. A. Chu 1987) (Nowick 1995) (Yun and Dill 1999) and Direct Mapping (DM) (D. L. Oliveira 2012). Logic synthesis works with the low-level specifications (STG or XBM), which capture a circuit's behavior at the signal transition level. This approach aims to obtain the minimized Boolean equations of the output and next state signals. This optimization requires an exploration of all possible ordinations of the specified events. The logic synthesis approaches are well established and supported by automatic synthesis tools. Direct mapping uses the graph specification of a system that is translated into a circuit netlist. Each node in the graph corresponds to a memory element (control cell), and the arcs represent the interconnections of the circuit. This method that starts from XBM specification was proposed by (D. L. Oliveira 2012).

Hardware Description: Linear Pipeline with Asynchronous Control

The elaborated asynchronous pipeline architecture belongs to the bundled-data class and follows the same behavior of the block diagram of Figure 4, applied in the same ASR system. Figure 19 shows its architecture design that was developed for prototyping in FPGA devices (Batista, Oliveira, et al. A New Asynchronous Pipeline Architecture of Support Vector Machine Classifier for ASR System 2019). The bundled-data pipeline architecture uses single-rail functional units and delays elements between stages. The delay is defined considering the critical path of each stage. In the architecture of (Oliveira, Garcia and D'Amore 2014), the control must satisfy the Fundamental Mode (FM) (Myers 2004). About FM mode, for activation of new input, the circuit must be stabilized. In our architecture, the control operates in I/O mode and the 2-phase protocol. Then, activating an output may allow the activation of new input, so it does not require timing analysis. Figure 20 shows the description of the proposed control that was specified in STG. Figures 21 and 22 show the STG specification of the control and the circuit synthesized in the Petrify tool (Cortadella, et al. 2004), respectively.

The classifications were made two by two (using OAO technique) manually, where external commands entered most of the data in the circuit. For a better explanation, the data was stored as follows. SV (Support Vector) data was stored in the FPGA device. Test Data was external input data. Thirteen Alpha values and one Bias value for each class were external input data stored in the FPGA device for each classification process.

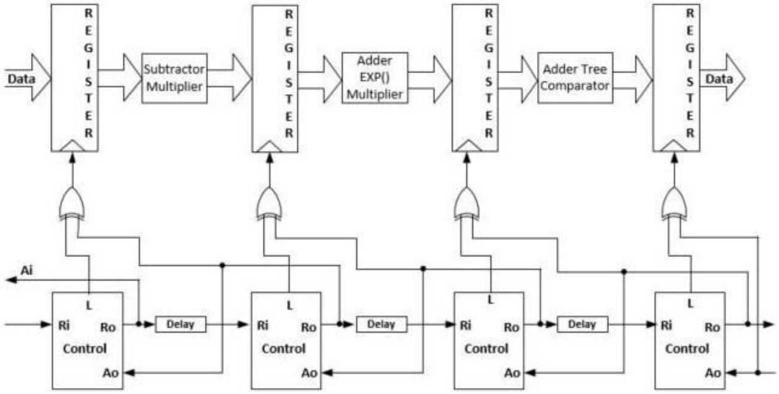


Figure 20. Proposed asynchronous linear pipeline.

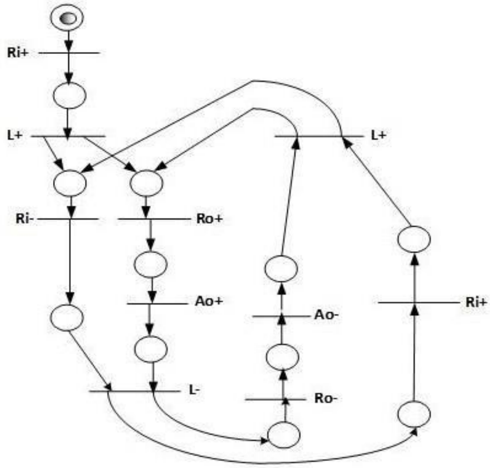


Figure 21. STG specification of the control.

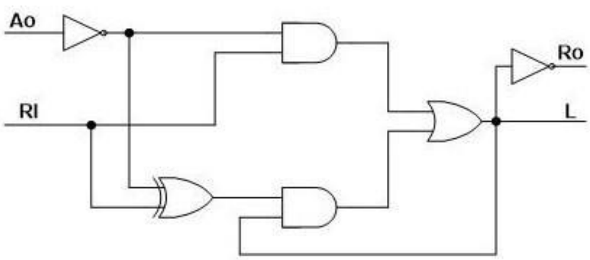


Figure 22. Netlist of the control.

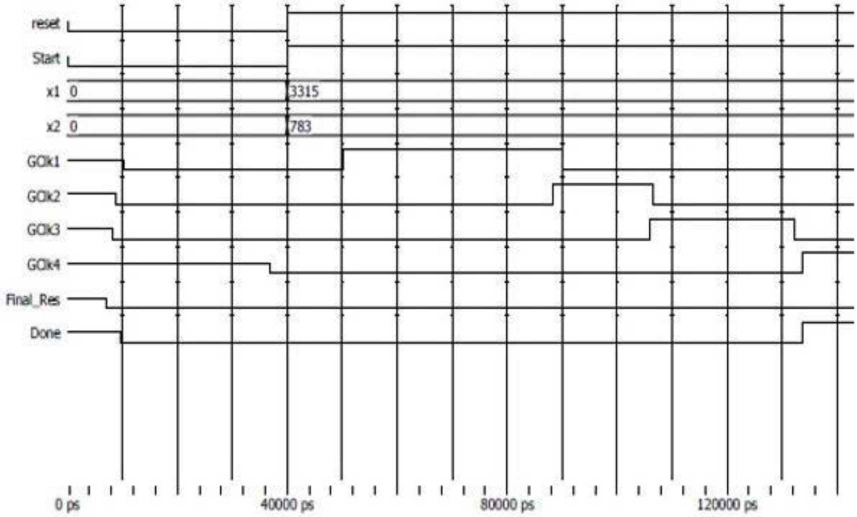


Figure 23. Timetable from post layout of the architecture IV (related to Figure 19).

Figure 23 shows the timetable of the post layout of the proposed architecture in Intel-Altera's Cyclone IV device is shown, where it is possible to observe: the *reset* (reset signal); *Start* (start signal); *x1* (data from column 1 of the Test matrix) and *x2* (data from column 2 of the Test matrix) signals; *GCLK1*, *GCLK2*, *GCLK3* and *GCLK4* (clock signals for the registers of the respective stages, generated from the control circuits of Figure 22); *Final_Res* (the classification result); and *Done* (the signal that indicates when the process is over).

This proposal used $\sigma = 0.05$ in the Gaussian (also called Radial Basis Function - RBF) kernel function. Besides, it was possible to acquire from its post-layout simulation, the following information:

- LUTs: 5,480 / 114,480 (1%)
- Registers: 1,773
- Embedded Multiplier 9-bit elements: 420 / 532 (79%)
- Dynamic Power: 109.95mW
- Throughput: 7.71MOPS
- Latency: 96.445ns

CONCLUSION AND FUTURE WORK

After the seminal works by Vapnik in 1995, SVMs (Support Vector Machines) had been applied in popular areas. It is known that the SVM algorithm is very computationally intensive because the optimization is based on Statistical Learning. This fact motivates the implementation of SVM algorithms with programable logic devices, FPGAs, and ASICs. In this chapter, research efforts to implement SVM algorithms with FPGAs are applied to diverse interest areas. From the idea of developing a portable embedded artificial intelligence system where low consumption is important, recent research of a fast and low consumption solution of digital circuit architecture based on an asynchronous paradigm is presented towards portable artificial intelligence devices.

REFERENCES

Afifi, S. M., H. G. Hosseini, and R. Sinha. 2015. "Hardware Implementation of SVM on FPGA: A State-of-the-Art Review of Current Practice." *International Journal of Innovative Science, Engineering and Technology* 1-20.

Afifi, S., H. G. Hosseini, and R. Sinha. 2018. "Dynamic hardware system for cascade SVM classification of melanoma." *Neural Computing and Applications* 1-12.

Andraka, A. 1998. "A Survey of CORDIC Algorithms for FPGA Based Computers." *Proceedings of the 1998 ACM/SIGDA sixth international symposium on Field programmable gate arrays*, 191-200.

Anguita, D., S. Pischiutta, S. Ridella, and D. Sterpi. 2006. "Feed-Forward Support Vector Machine without Multipliers." *IEEE Transactions on Neural Networks* 1328-1331.

Batista, G. C., and W. L. S. Silva. 2015. "Using Support Vector Machines and two-dimensional discrete cosine transform in speech automatic recognition." *International Joint Conference on Neural Networks (IJCNN),* October 1: 1-5.

Batista, G. C., D. L. Oliveira, D.A. Silva, O. Saotome, and L. Romano. 2019. "FPGA Implementation of a Pipeline Architecture of SVM Classifier for ASR System." *9th Workshop on Circuits and System Design (WCAS),* August 27: 1-4.

Batista, G. C., D. L. Oliveira, O. Saotome, and L. Romano. 2019. "New Approach of Pipelined Architecture of SVM Classifier for ASR System in FPGA Implementation." *X Southern Programmable Logic Conference (SPL),* 1-4.

Batista, G. C., D. L. Oliveira, O. Saotome, and T. S. Curtinhas. 2019. "A New Asynchronous Pipeline Architecture of Support Vector Machine Classifier for ASR System." *XXVI International Conference on Electronics, Electrical Engineering and Computing (INTERCON),* October 3: 1-4.

Batista, G. C., W. L. S. Silva, D. L. Oliveira, and O. Saotome. 2019. "Automatic speech patterns recognition of commands using SVM and PSO." *Multimedia Tools and Applications* 1-23.

Bichop, D. 2005. *Fixed and floating point packages for VHDL.* Rochester, NY: Eastman Kodak Company.

Bresolin, A. A. 2008. *Reconhecimento de voz através de unidades menores do que a palavra, utilizando Wavelet Packet e SVM, em uma nova estrutura hierárquica de decisão* [*Speech recognition through units smaller than the word, using Wavelet Packet and SVM, in a new hierarchical decision structure*]. Doctoral Thesis, Natal: Universidade Federal do Rio Grande do Norte.

Burges, Christopher J. C. 1998. "Tutorial on support vector machines for pattern recognition." *Data Mining and Knowledge Discovery* 121-167.

Chu, P. P. 2008. *FPGA Prototyping by VHDL Examples.* New Jersey: John Wiley & Sons, Inc.

Chu, T. A. 1987. *Synthesis of Self-Timed VLSI Circuits from Graph-Theory Specifications.* PhD thesis, Massachusets: Dept. of EECS, MIT.

Cortadella, J., A. Kondratyev, L. Lavagno, and C. Sotiriou. 2004. "Coping with the variability of combinational logic delays." *IEEE International*

Conference on Computer Design: VLSI in Computers and Processors, November 8: 1-4.

Cortes, C., and V. Vapnik. 1995. "Support-vector networks." *Machine Learning* 273-297.

Elhosary, H., M. H. Zakhari, M. A. Elgammal, A. A. Mohamed, N. S. Khaled, and M. Hassan. 2019. "Low-Power Hardware Implementation of a Support Vector Machine Training and Classification for." *IEEE Transactions on Biomedical Circuits and Systems* 1-15.

Goldberg, D. E. 1989. *Genetic algorithms in search, optimization and machine learning*. Boston: Addison-Wesley.

Gowda, M. N., and A. I. Rashed. 2017. "Hardware Implementation of Hybrid Classifier to Detect Cancer Cells." *14th IEEE India Council International Conference (INDICON),* October 11: 1-5.

Haykin, S. S. 2002. *Redes Neurais: Princípio e Prática* [*Neural Networks: Principle and Practice*]. Bookman.

Hebb, D. O. 1949. *The organization of behavior*. New York: Jonh Wiley & Sons, Inc.

Jallad, A. H. M., and L. B. Mohammed. 2014. "Hardware Support Vector Machine (SVM) for Satellite On-Board Applications." *NASA/ESA Conference on Adaptive Hardware and Systems (AHS),* August 21: 256-261.

Kanisha, B., S. Lokesh, P. M. Kumar, P. Parthasarathy, and G. C. Babu. 1998. "Speech recognition with improved support vector machine using dual classifiers and cross fitness validation." *IEEE Transactions on VLSI Systems* 643-655.

Koide, T., H. Anh-Tuan, T. Okamoto, S. Shigemi, T. Mishima, T. Tamaki, B. Raytchev, et al. 2014. "FPGA Implementation of Type Identifier for Colorectal Endoscopic Images with NBI Magnification." *IEEE Asia Pacific Conference on Circuits and Systems*, April 11: 651-654.

Kuan, T., Jhing-Fa Wang, and Jia-Ching Wang. 2012. "VLSI Design of an SVM Learning Core on Sequential." *IEEE Transactions on Very Large Integration (VLSI) Systems* 673 - 683.

Kyrkou, C., C. S. Bouganis, T. Theocharides, and M. M. Polycarpou. 2015. "Embedded Hardware-Efficient Real-Time Classification with

Cascade Support Vector Machines." *IEEE Transactions on Neural Networks and Learning Systems* 99-112.

Lazinica, A. 2009. *Particle Swarm Optimization.* London: In-Tech.

Mizuno, K., T. Terachi, K. Takagi, S. Izumi, H. Kawaguchi, and M. Yoshimoto. 2013. "Architectural Study of HOG Feature Extraction Processor for Real-Time Object Detection." *IEEE Workshop on Signal Processing Systems (SiPS),* February 14: 197-202.

Myers, C. J. 2004. *Asynchronous Circuit Design.* New Jersey: Wiley & Sons, Inc.

Nayak, J., B. Naik, and H. Behera. 2015. "A Comprehensive Survey on Support Vector Machine in Data Mining Tasks: Applications & Challenges." *International Journal of Database Theory and Application* 168-186.

Nowick, S. M. 1995. *Automatic Synthesis of Burst-Mode Asynchronous Controller*. PhD thesis, Stanford: Stanford University.

Oliveira, D. L. and Noé, A. and Faria, L. A. 2012. "Synthesis by Direct Mapping of Extended Burst-Mode Asynchronous Controller using RS Latch." *3rd IEEE Latin American Symposium on Circuits and Systems,* April 12: 1-4.

Oliveira, D. L., K. Garcia, and R. D'Amore. 2014. "Using FPGAs to Implement Asynchronous Pipelines." *5th IEEE Latin American Symposium on Circuits and Systems*, May 26: 1-4.

Papadonikolakis, M., and C. S. Bouganis. 2010. "A Heterogeneous FPGA Architecture for Support Vector Machine Training." *IEEE Symposium on Field-Programmable Custom Computing Machines-FCCM*, June 1: 211-214.

Patil, R., G. Gupta, V. Sahula, and A. Mandal. 2012. "Power Aware Hardware Prototyping of Multi-class SVM Classifier Through Reconfiguration." *25th International Conference on VLSI Design (VLSID),* March 12: 62-67.

Peng, C., B. Chen, T. Kuan, P. Lin, J. Wang, and N. Shih. 2014. "REC-STA: Reconfigurable and Efficient Chip Design With SMO-Based Training Accelerator." *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 1791-1802.

Pietron, M., M. Wielgosz, D. Zurek, E. Jamro, and K. Wiatr. 2013. "Comparison of GPU and FPGA Implementation of SVM aAlgorithm for Fast Image Segmentation." *Architecture of Computing Systems - ARCS*, 292-302.

Rabieah, M. B., and C. Bouganis. 2015. "FPGA based non-linear Support Vector Machine training using an ensemble learning." *25th International Conference on Field Programmable Logic and Applications (FPL)*, October 8: 1-4.

Ruiz-Llata, M., G. Guarnizo, and Yébenes-Calvino. 2010. "FPGA Implementation of a Support Vector Machine for." *International Joint Conference on Neural Networks (IJCNN)*, October 14: 1-5.

Scholkopf, B., O. Simard, A. Smola, and V. Vapnik. 1997. "Prior knowledge in support vector kernels." *NIPS'97 Proceedings of the 10th International Conference on Neural Information Processing Systems*, January 1: 640-646.

Song, X., H. Wang, and L. Wang. 2014. "FPGA Implementation of a Support Vector Machine based Classification System and its Potential Application in Smart Grid." *Proceedings of the IEEE 11th International Conference on Information Technology: New Generations*, June 2: 397-402.

Vapnik, V. N. 2000. *The nature of statistical learning theory*. New York: Spring-Verlag.

Vapnik, V. N., and A. Y. Chervonenkis. 1967. "*On the uniform convergence of relative frequencies of their probalities to events*." Springer-Verlag Berlin Heidelberg 7-12.

Wang, Y., Z. Li, L. Feng, H. Bai, and Wang C. 2018. "Hardware design of multi-class SVM classification for epilepsy and epileptic seizure detection." *IET Circuits, Devices and Systems* 108-115.

Yun, K. Y., and D. L. Dill. 1999. "Automatic Synthesis of Extended Burst-Mode Circuits: Part I (Specification and Hazard-.Free Implementation) and Part II (Automatic Synthesis)." *IEEE Trans. on CAD of Integrated Circuit and Systems* 101-132.

BIOGRAPHICAL SKETCHES

Gracieth Cavalcanti Batista

Affiliation: Electronic Engineer Division – Technological Institute of Aeronautics – ITA

Education: Electronics technician, Federal Institute of Maranhão – IFMA, Brazil, 2010; Electrical Engineer, Federal Institute of Maranhão – IFMA, Brazil, 2018; Master's degree in Electronics Engineer and Computing - Technological Institute of Aeronautics - ITA, Brazil, 2019.

Business Address: Rua Afonso César de Siqueira, 212. Apto 66 – Vila Adyana – São José dos Campos – SP, Brazil 12243-710

Research and Professional Experience: Machine Learning, Artificial Intelligence for Decision-making Systems, Digital Signal Processing, FPGA prototyping, Analog electronics, Embedded systems, Microeletronics.

Professional Appointments: Engineering doctoral student at Technological Institute of Aeronautics – ITA, Brazil.

Publications from the Last 3 Years (2016-2019):

13 Conference papers; 1 Journal paper:

Batista, G. C., and W. L. S. Silva. "Using Support Vector Machines and two dimensional discrete cosine transform in speech automatic recognition." *International Joint Conference on Neural Networks (IJCNN)*, October 1, 2015: 1-5.

Batista, G. C., D. L. Oliveira, D.A. Silva, O. Saotome, and L. Romano. "FPGA Implementation of a Pipeline Architecture of SVM Classifier for ASR System." *9th Workshop on Circuits and System Design (WCAS)*, August 27, 2019: 1-4.

Batista, G. C., D. L. Oliveira, O. Saotome, and L. Romano. “New Approach of Pipelined Architecture of SVM Classifier for ASR System in FPGA Implementation.” *X Southern Programmable Logic Conference (SPL)*, 2019: 1-4.

Batista, G. C., D. L. Oliveira, O. Saotome, and T. S. Curtinhas. “A New Asynchronous Pipeline Architecture of Support Vector Machine Classifier for ASR System.” *XXVI International Conference on Electronics, Electrical Engineering and Computing (INTERCON)*, October 3, 2019: 1-4.

Batista, G. C., W. L. S. Silva, and A. G. Menezes. “Automatic speech recognition using Support Vector Machine and Particle Swarm Optimization.” *IEEE Symposium Series on Computational Intelligence (SSCI)*, 2016: 1-5.

Batista, G. C., W. L. S. Silva, D. L. Oliveira, and O. Saotome. “Automatic speech patterns recognition of commands using SVM and PSO.” *Multimedia Tools and Applications*, 2019: 1-23.

Delsoto, H. A., D. L. Oliveira, G. C. Batista, D. A. SIlva, and L. Romano. “A Tools Flow for Synthesis of Asynchronous Control Circuits from Extended STG Specifications.” *IEEE 10th Latin American Symposium on Circuits & Systems (LASCAS)*, 2019: 225-229.

Oilveira, D. L., O. Verducci, V. L. V. Torres, R. L. Moreno, and G. C. Batista. “An New Implementation of Quasi Delay Insensitive Combinational Digital Circuits.” *25th IBERCHIP Workshop*, 2019: 1-4.

Oliveira, D. L., Batista G. C., D. A. Silva, and L. Romano. “A New Synchronous Control for Latency-Insensitive Pipeline Architecture.” *25th IBERCHIP Workshop*, 2019: 1-4.

Oliveira, D. L., D. A. Silva, G. C. Batista, and L. Romano. “ A State Assignment Method for Extended Burst-Mode gC Finite State Machines Based on Genetic Algorithm.” *25th IBERCHIP Workshop*, 2019: 1-4.

—. “Architecture for Low-Power Gated-Clock Synchronous FSMs Operating on Double-Edge Clock.” *25th IBERCHIP Workshop*, 2019: 1-4.

—. "Design of Locally-Clock Asynchronous State Machines Aimed at FPGAs Based in LUTs." *25th IBERCHIP Workshop*, 2019: 1-4.

Oliveira, D. L., D. A. Silva, G. C. Batista, and T. S. Curtinhas. "Design of Locally-Clocked Extended Burst-Mode AFSMs without Timing Analysis." *IEEE XXVI International Conference on Electronics, Electrical Engineering and Computing (INTERCON)*, 2019: 1-4.

Oliveira, D. L., O. Verducci, G. C. Batista, and T. S. Curtinhas. "A High Performance Implementation of Quasi Delay Insensitive Booleans Functions." *IEEE XXVI International Conference on Electronics, Electrical Engineering and Computing (INTERCON)*, 2019: 1-4.

Oliveira, D. L., O. Verducci, V. L. V. Torres, G. C. Batista, R. L. Moreno, and L. Romano. "An Implementation of Extended Burst-Mode Specifications as Quasi Delay Insensitive State Machines." *Latin American Symposium on Circuits & Systems (LASCAS)*, 2019: 177-221.

Silva, D. A., D. L. Oliveira, G. C. Batista, and L. Romano. "An Implementation of the DES Encryption Algorithm with High Throughput using FPGA." *25th IBERCHIP Workshop*, 2019: 1-4.

Duarte Lopes de Oliveira

Affiliation: Electronic Engineer Division – Technological Institute of Aeronautics – ITA

Education: He received the BSc degree in electrical engineering from University of Mogi das Cruzes, Brazil, in 1980, MSc degree in computer science from the Aeronautics Institute of Technology, SJC – São Paulo – Brazil, in 1988 and D.E.E degree in electrical engineering from Polytechnic Scholl of University of São Paulo in 2004.

Business Address: Praça Marechal Eduardo Gomes, 50 – Vila das Acácias | CEP 12228-901 São José dos Campos/SP – Brazil. Fone: (+55) 12-3947-6813 – E-mail: duarte@ita.br

Research and Professional Experience: His research interests include asynchronous circuits, computer-aided digital design, low-power digital systems, and logic synthesis.

Professional Appointments: Associate Professor at Technological Institute of Aeronautics – ITA – SP – Brazil

Publications from the Last 3 Years:

Delsoto, H.A., Oliveira, D. L., Batista, G. C., Silva, D. A., Romano, L. "A Tools Flow for Synthesis of Asynchronous Control Circuits from Extended STG Specifications" *IEEE 10th Latin American Symposium on Circuits & Systems (LASCAS),* 2019: 225-228.

Curtinhas, T. S., Oliveira, D.L., Batista, G.C., Torres, V.L. V., Romano, L. "A State Assignment Method for Extended Burst-Mode gC Finite State Machines Based on Genetic Algorithm" *IEEE 10th Latin American Symposium on Circuits & Systems (LASCAS),* 2019: 153-156.

Oliveira, D.L., Verducci, O., Torres, V.L.V., Batista, G.C., Moreno, R. L., Romano, L., "An Implementation of Extended Burst-Mode Specifications as Quasi Delay Insensitive State Machines" *IEEE 10th Latin American Symposium on Circuits & Systems (LASCAS),* 2019: 177-180.

Oliveira, D.L., Verducci, O., Torres, V.L.V., Moreno, R.L., Batista, G.C., "A New Implementation of Quasi Delay Insensitive Combinational Digital Circuits" *25th IBERCHIP Workshop, Armenia, Colombia*, 2019: 1-4.

Oliveira, D.L., Silva, D. A., Batista, G.C., Romano, L., "Architecture for Low-Power Gated-Clock Synchronous FSMs Operating on Double-Edge Clock" *25th IBERCHIP Workshop, Armenia, Colombia*, 2019: 1-4.

Oliveira, D.L., Batista, G.C., Silva, D. A., Romano, L., "A New Synchronous Control for Latency-Insensitive Pipeline Architecture" *25th IBERCHIP Workshop, Armenia, Colombia*, 2019: 1-4.

Silva, D. A., Oliveira, D.L., Batista, G.C., Romano, L., "An Implementation of the DES Encryption Algorithm with High Throughput using FPGA" *25th IBERCHIP Workshop, Armenia, Colombia*, 2019: 1-4.

Oliveira, D.L., Silva, D. A., Batista, G.C., Romano, L., "Design of Locally-Clock Asynchronous State Machines Aimed at FPGAs Based in LUTs" *25th IBERCHIP Workshop, Armenia, Colombia*, 2019: 1-4.

Batista, G.C., Oliveira, D.L., Saotome, O., Romano, L., "New Approach of Pipelined Architecture of SVM Classifier for ASR System in FPGA Implementation" *X Southern Programmable Logic Conference (SPL), Buenos Aires, Argentina,* 2019: poster

Oliveira, D.L., Silva, D. A., Batista, G.C., Curtinhas, T.S., "Design of Locally-Clocked Extended Burst-Mode AFSMs without Timing Analysis" *IEEE XXVI International Conference on Electronics, Electrical Engineering and Computing (INTERCON), Lima, Peru*, 2019:1-4.

Oliveira, D.L., Verducci, O., Batista, G.C., Curtinhas, T.S., "A High Performance Implementation of Quasi Delay Insensitive Booleans Functions" *IEEE XXVI International Conference on Electronics, Electrical Engineering and Computing (INTERCON), Lima, Peru*, 2019:1-4.

Batista, G.C., Oliveira, D.L., Saotome, O., Curtinhas, T.S., "A New Synchronous Pipeline Architecture of Support Vector Machine Classifier for ASR System" *IEEE XXVI International Conference on Electronics, Electrical Engineering and Computing (INTERCON), Lima, Peru*, 2019:1-4.

Oliveira, D.L., Cardoso, N.M., Silva, D.A., Batista, G.C., Romano, L., "An Asynchronous Wrapper with a Single Communication Port for GALS Pipeline Systems" *9th Workshop on Circuits and System Design (WCAS), São Paulo, Brazil,* 2019: 1-4.

Oliveira, D.L., Cardoso, N.M., Silva, D.A., Batista, G.C., Romano, L., "FPGA Implementation of a Pipeline Architecture of SVM Classifier for ASR System" *9th Workshop on Circuits and System Design (WCAS), São Paulo, Brazil,* 2019: 1-4.

Oliveira, D. L., Verducci, O., Torres, V. L. V., Silva, D. A., Romano, L., "Implementation of Self-Timed Combinational Digital Circuits without using C Elements" *9th Workshop on Circuits and System Design (WCAS), São Paulo, Brazil,* 2019: 1-4.

Oliveira, D.L., Cardoso, N.N.M., Batista, G.C., Silva, D.A., Romano, L., "A New Synchronous-Asynchronous Mixed Pipeline Architecture with Clock-Gating" *XXI Simpósio de Aplicações Operacionais em Áreas de Defesa (SIGE), São José dos Campos, Brazil*, 2019: 176-180.

Silva, D. A., Verducci, O., Oliveira, D. L., "Implementation of DES Algorithm in New Non-Synchronous Architecture Aiming DPA Robustness" 27th IFIP/IEEE International Conference on Very Large Scale Integration (VLSI-SOC), Cuzco, Peru, 2019: 228-229.

Batista, G.C., Silva, W.L.S., Oliveira, D.L., Saotome, O., "Automatic speech patterns recognition of commands using SVM and PSO" *MULTIMEDIA TOOLS AND APPLICATIONS - Springer, Vol. 78,* 2019:1-23.

Washington Luis Santos Silva

Affiliation: Federal Institute of Maranhão

Education:

Bachelor's degree: Federal University of Maranhão (UFMA- 1996)
Master Science: University of São Paulo (USP - 2002)
PhD: Federal University of Maranhão (UFMA -2015)

Business Address: Avenue Getulio Vargas, nº 4, Monte Castelo, CEP: 65000-00, São Luis-Ma.

Research and Professional Experience:

Dr. Santos Silva is Electrical technician from the Federal Technical School of Maranhão (1989), Graduated in Electrical Engineering from the Federal University of Maranhão (1996), Master in Electrical Engineering from the University of São Paulo-USP (2002) and Ph.D. in Electrical Engineering from the Federal University of Maranhão -UFMA (2015). Experience in Electrical Engineering, Electronic Engineering and Telecommunications Engineering, he worked for eight years as an engineer of electronic systems of high-performance aircraft, as well as electrical systems of cargo transport aircraft. Has to experience in radio navigation and radio communication electronic systems of military aircraft. Head of the Avionic Electronic Instrument and Equipment Maintenance Section; Head of the instrument calibration department of the Electronics Subdivision of the São Paulo Aeronautical Material Park, the Aeronautics Command. Has the course of the quality auditor of IFI (Institute of Development and Industrial Coordination) - DCTA. Professor of the Electronics and Higher Industrial Automation Technical Course at the Federal Institute of Education, Science, and Technology of São Paulo (IFSP). Engineering Professor at UNICID, UNIBAN, and FASP. Advisor of several academic works in the courses of Telecommunications Engineering and Computer Engineering. Experience of 15 years in the field of Professional Upgrading, Technical Education, Technological Education, and University Education. Coordinator of the Industrial Mechatronics Technology Course at the Faculty of Thermomechanical Technology. Laboratory Coordinator of the Federal Institute of São Paulo, Campus: Cubatão. Professor at the Federal Institute of Maranhão. Coordinator of the IFMA Digital Systems Laboratory, São Luís-Monte Castelo Campus. Director of the IFMA Infrastructure and Technology Directorate, São Luis-Monte Castelo Campus.

Professional Appointments:

1. Director of Infrastructure and Technology of the Federal Institute of Maranhão (2018-);

2. Mayor of Campus of the Federal Institute of Maranhão (2016 -2018).
3. Associate Professor at the Federal Institute of Maranhão (2010-).

Publications from the Last 3 Years:

Rocha, P.; Silva, Washington; Barros, Allan. Hierarchical Expert Neural Network System for Speech Recognition. *Journal of Control, Automation and Electrical Systems*, v. 7, p. 1-13, 2019.

Batista, G. C.; Silva, W. L. S.; Oliveira, D. L.; Saotome, O. Automatic speech patterns recognition of commands using SVM and PSO. *Multimedia Tools and Applications*, v. 78, p. 1-23, 2019.

Silva, W. L. S.; Serra, G. *Inteligência Computacional aplicada ao reconhecimento de voz: Uma abordagem utilizando modelos preditivos de baixa ordem* [*Computational Intelligence applied to speech recognition: An approach using low-order predictive models*]. 1. ed. Rio de Janeiro: OmniScriptum Publishing Group, 2019. v. 1. 165p .

Lima, Priscila; Barros, Allan; Silva, Washington . Neural Network Configurations Analysis for Identification of Speech Pattern with Low Order Parameters. In: Yaxin BiSupriya KapoorRahul Bhatia. (Org.). *Studies in Computational Intelligence*. 1ed.: Springer International Publishing, 2018, v. 751, p. 349-370.

Silva, W. L. S.; Rocha, P. L.; Filho, A. K. D. B. Neural Network Configurations Analysis for Multilevel Speech Pattern Recognition System with Mixture of Experts. In: Chatchawal Wongchoosuk. (Org.). *Intelligent System*. 1ed. London: InTech, 2018, v. 1, p. 145-164.

Rocha, W. A. C.; Silva, W. L. S. Projeto de um rastreador solar ativo microcontrolado de baixo custo com dois eixos e monitoramento da grandezas. In: XII CONNEPI, 2018, Recife. XII *CONNEPI Anais*, 2018.

Sampaio, D. D.; Silva, W. L. S. Sistema nebuloso para navegação autônoma de veículo aéreo não tripulado. In: XII CONNEPI, 2018, Recife. *XII CONNEPI Anais*, 2018.

Rocha, P. L.; Silva, W. L. S.; Barros, A. K. Análise de configurações de redes neurais para sistema de reconhecimento de voz utilizando

parâmetros TCD de baixa ordem. In: *Simpósio Brasileiro de Automação Inteligente* [*Brazilian Symposium on Intelligent Automation*, 2017, Porto Alegre-RS. SBAI Proceedings], 2017, Porto Alegre-RS. Proceedings SBAI 2017. São Paulo-SP: Sociedade Brasileira de Automática, 2017. p. 2207-2212.

Rocha, P. L.; Silva, W. L. S. Recognition System of Numerical Command of Speech Signal using Neural Networks based on DCT Parameters. In: IntelliSys 2016 Conference, 2016, Londres. *Proceedings of 2016 SAI Intelligent Systems Conference (IntelliSys)*. Londres: IEEE Proceedings, 2016. v. 1. p. 939-946.

Rocha, P. L.; Silva, W. L. S. Intelligent System of Speech Recognition using Neural Networks based on DCT Parametric Models of Low Order. In: *EEE World Congress on Computational Intelligence (IEEE WCCI)* 2016, 2016, Vancouver. WCCI 2016 Proceedings. Vancouver: IEEE Proceedings, 2016. v. 1. p. 788-795.

Osamu Saotome

Affiliation: Electronic Engineer Division – Technological Institute of Aeronautics – ITA

Education: Electronic Engineer, Technological Institute of Aeronautics - ITA, Brazil, 1974; Doctor of Engineering, Technological Institute of Tokyo – TITech, Tokyo, Japan, 1987

Business Address: DCTA – ITA – IEEA – São José dos Campos, SP – Brazil 12228-900

Research and Professional Experience: Embedded Systems, Microelectronics, Artificial Intelligence for Control Systems, Digital Signal Processing and Applications, Electronics for Aeronautics and Biomedical Applications.

Professional Appointments: Full Professor at Technological Institute of Aeronautics – ITA – SP – Brazil.

Publications from the Last 3 Years (2016 – 2019):

03 Journal Articles; 09 Conference Articles; 02 Book Chapters; 02 Patents

Full articles published in journals:

1. Costa, R. F.; Saotome, O.; Rafikova, E. Simulation and Validation of Satellite Attitude Control Algorithms in a Spherical Air Bearing. *Journal of Control, Automation and Electrical Systems*, v. 1, p. 1, 2019.
2. Batista, G. C.; Silva, W. L. S.; Oliveira, D. L.; Saotome, O. Automatic speech patterns recognition of commands using SVM and PSO. *Multimedia Tools and Applications*, v. 78, p. 1-2, 2019.
3. Falcari, T.; Saotome, O.; Pires, R.; Campo, A. B. Evaluation of multi-class support-vector machines strategies and kernel adjustment levels in hand posture recognition by analyzing sEMG signals acquired from a wearable device. *Biomedical Engineering Letters*, v. 1, p. 1-12, 2019.

Full papers published in congress proceedings:

1. Oliveira, D. L.; Brandolin, J. B.; Verducci, O.; Torres, V. L. V.; Saotome, O. An Approach for the Design of Asynchronous NULL Convention Logic (NCL) Circuits using FPGA. In: 24th International IBERCHIP Workshop, 2018, Puerto Villarta, Jalisco, Mexico. *Proceedings of the 24th International IBERCHIP Workshop*, 2018.
2. Oliveira, D. L.; Brandolin, J. B.; Torres, V. L. V.; Verducci, O.; Saotome, O. An Architecture for Self-Timed Asynchronous Pipeline Circuits using FPGAs. In: 24th International IBERCHIP Workshop, 2018, Puerto Villarta, Jalisco, Mexico. *Proceedings of the 24th International IBERCHIP Workshop*, 2018.
3. Curtinhas, T.; Oliveira, D. L.; Saotome, O. Vhdlasyn: A Tool for Synthesis of Asynchronous Systems from VHDL Behavioral Specifications. In: 2018 2nd Conference on PhD Research in Microelectronics and Electronics Latin America (PRIME-LA), 2018, Puerto Villarta, Jalisco, Mexico. *Proceedings of the 2018 2nd*

Conference on PhD Research in Microelectronics and Electronics Latin America (PRIME-LA), 2018.

4. Curtinhas, T.; Oliveira, D. L.; Saotome, O. Design of Extended Burst-Mode Asynchronous Controllers using Synchronous CAD Tools. In: 2018 2nd Conference on PhD Research in Microelectronics and Electronics Latin America (PRIME-LA), 2018, Puerto Villarta, Jalisco, Mexico. *Proceedings of the 2018 2nd Conference on PhD Research in Microelectronics and Electronics Latin America (PRIME-LA)*, 2018.
5. Curtinhas, T.; Oliveira, D. L.; Torres, V. L. V.; Saotome, O. Design Flow for Synthesis of XBM Direct Output Synchronous FSMs for Heterogeneous Systems. In: 24th International IBERCHIP Workshop, 2018. *Proceedings of the 24th International IBERCHIP W*orkshop, 2018.
6. Loubach, D.; Nobrega, E. G. O.; Sander, I.; Soderquist, I.; Saotome, O. Towards Runtime Adaptivity by using Models of Computation for Real-Time Embedded Systems Design. In: Aerospace Technology Congress, 2016, Stockholm, Sweden. *Proceedings of the Aerospace Technology Congress*, 2016.
7. Farias, S.V.; Saotome, O.; Nobrega, E. G. O.; Sande, I.; Soderquist, I.; Loubach, D. Multiple Damage Detection in Plate Structure using Piezoelectric Array Sensors and 2D-MUSIC Spectrum. In: Aerospace Technology Congress, 2016, Stockholm, Sweden. *Procceedings of the Aerospace Technology Congress*. Stockholm, 2016.
8. Romero, R.F. Saotome, O.; Loubach, D.; Nobrega, E. G. O.; Sander, I.; Soderquist, I. IMA Module Architecture Approach based on Partial Reconfigurable FPGA. In: Aerospace Technology Congress, 2016, Stockholm, Sweden. *Proceedings of the Aerospace Technology Congress*, 2016.
9. Saotome, O.; Farias, S. V.; Nobrega, E. G. O.; Sandeer, I.; Soderquist, I.; Loubach, D. Structure Health Monitoring using Discrete Wavelet Transform and Piezoelectric Sensor Array. In: Aerospace Technology Congress, 2016, Stockholm, Sweden. *Proceedings of the Aerospace Technology Congress*, 2016.

Chapters of books published:

1. Amaral, T. L.; Saotome, Osamu; Matieli, J. E. 11.Electronic device to identify surgical material retained in cavities after surgery. In: UERN. (Org.). 11.Electronic device to identify surgical material retained in cavities after surgery. *UERN*, 2019, v. 1, p. 50-65.
2. Estrela, V. V.; Magalhães, H. A.; Saotome, O. Total Variation Applications in Computer Vision. Research Handbook on Emerging Perspectives in Intelligent Pattern Recognition, Analysis, and Image Processing. *IGI Global*, 2016, v. 1, P. 41-64.

In: Support-Vector Machines
Editor: Pooja Saigal
ISBN: 978-1-53618-757-1

Chapter 5

SPEAKER RECOGNITION USING SUPPORT VECTOR MACHINE

Nivedita Palia[1,*], Deepali Kamthania[1,2] and Shri Kant[2]
[1]School of Information Technology,
Vivekananda Institute of Professional Studies,
Guru Gobind Singh Indraprastha University, Delhi, India
[2]Research and Technology Development Center,
Sharda University, Greater Noida, Uttar Pradesh, India

ABSTRACT

In this Chapter, an attempt has been made to develop Bilingual Speaker Recognition using Mel Frequency Cepstral Coefficient and Support Vector Machine. In the present study, two speech corpuses developed in Hindi and the English language have been considered for experimental evaluation. In order to study the impact of the recording device, recording language and data size on the accuracy of the speaker recognition system, seventeen sets of experiments have been performed. Experimental results are validated by calculating accuracy, precision, true positive rate, F1 score, and error rate.

[*] Corresponding Author's Email:nivedita134@gmail.com.

Keywords: Speaker Recognition, Bilingual Speech Corpus, Mel Frequency Cepstral Coefficient, Support Vector Machine, Confusion Matrix

INTRODUCTION

Speaker recognition (SR) is the task of recognizing an individual from the properties of speech samples. SR started getting attention in 1960s (Pruzansky 1963). Unlike other biometrics recognition system such as the face recognition, iris recognition, palm recognition, etc. the SR does not require the physical presence of an individual it can be accessed remotely using telephone networks or mobile phones. The accuracy of a SR system is influenced by the number of factors like background noise, sample size, health, age, emotion, language spoken, session variability etc. It comprises of two main tasks i.e., Speaker Identification and Speaker Verification. Speaker identification is a task of identifying the person from the group of known speakers while Speaker verification is a task of verifying a claimed identity of a speaker. The SR system worked in two different phases: training and testing. In the training phase, speaker-specific feature parameters are extracted from the speech samples and are used to construct a model for different speakers. In the testing phase, the speech samples from unknown speakers are classified by comparing the generated model using the classification technique.

The SR task primarily depends upon the small engrossed speech or speech samples collected from individuals. The SR on the basis of the text is broadly categorized into two categories: Text-Independent SR and Text-Dependent SR. In text-dependent SR, the same phrase is used for testing and training (Antonsen 2017). While in text-independent SR different phrases are used for testing and training.

Pruzansky at Bell Labs (1963) has initiated work in the field of SR by using filter banks and spectrograms for computing similarity measures. Doddington (1971) used formant analysis in place of the filter bank. Many researchers have worked independently to extract speaker-specific features for the purpose of text-independent SR. These features consisted of linear prediction coefficients (Sambur 1972), auto-correlation (Bricker et al. 1971), and fundamental frequency histograms (Beck, Neuberg, and Hodge 1977) etc. Furui (1981) has suggested the cepstral coefficient as a frame-based feature to enhance the robustness of a SR system and later it became a standard feature (Furui 2009). Burton (1987) and Soong et al. (1985) have introduced Vector Quantization. Vector Quantization is the simplest text-independent model in which the feature vector is assigned to the nearest cluster. Gaussian Mixture Model (GMM) has been proposed by Rose and Reynolds (1990) and it is an extension of Vector Quantization. It is one of the robust parametric models for text-independent SR. These clusters are overlapped. Currently, Support Vector Machine (SVM) is one of the common and robust classifiers used for SR. It is a discriminative classifier that has been implemented for both spectral and high-level features (Campbell et al. 2004 and Campbell et al. 2006). Gu and Thomas (2001) have carried out experiments on UK English database developed using public telephone networks and found that SVM performed better than GMM. EER of 1.2% has been achieved in the case of SVM as compared to 3.9% for GMM on the same training and testing data. Kamruzzaman (2007) identified a speaker using Mel Frequency Cepstral Coefficient (MFCC) and SVM has been founded that sequential minimum optimization learning technique for SVM performs better than traditional technique Chunking and Osuna for SVM. Nijhawan and Soni (2014) and Vibhhute and Hibare (2015) in their work compared the performance of the speaker identification system by using different kernel functions of SVM and found that Radial Basis Function provides the best result as compared to another kernel.

In this paper, the performance of a SR has been evaluated using MFCC and SVM. Experiments have been carried out using two Speech Corpus (SC) to study the impact of recording languages, recording devices and data size, that effect the accuracy of the system. These experiments have been carried out on two different Bilingual SC having speech samples collected from mobile phones, headset and desktop mounted microphone in a realistic environment. Earlier developed SC have been recorded using a microphone and for the English language and small work has been done for Hindi Language (Agrawal 2010).

In Section 2 brief reviews of two speech corpora have been discussed. Section 3 describes the feature extraction and classification technique followed by the model evaluation using different performance parameters and pseudo-code to perform experiments. Different experiments and results have been elaborated in section 4. The conclusion and future scope of the work have been concluded in section 5.

SPEECH CORPUS

In the area of speech processing, speech corpora play a significant role as it is base for the experimental study. On the basis of the review, it has been concluded that very little work has been carried out in the area of Hindi SC (Shrishrimal et al. 2012 and Nivedita et al. 2013). The existing Hindi SC is created in a clean environment; domain-specific and speech have been recorded using microphones. The SC is not easily accessible because they are generally developed for government and forensic purpose.

In the current study to perform different sets of experiments, the two SC which has been considered is Corpus 1 and Corpus 2. In both corpora, the speech samples have been sampled at 16 kHz, having a bit rate of 256 Kbits per second and mono channels are considered for experimental study.

Corpus 1

Corpus 1 consists of 2400 speech samples of 08 speakers (Sinha, Agrawal, & Olsen 2011; Agrawal et al. 2012). The speech utterances have been collected using three recording channels: a mobile phone, a headset and a desktop mounted microphone in a room environment. The recording sentences have been framed using words from English and Hindi language. 100 utterances of each individual per device have been recorded for experiments. The speech file length varies from 5 to 8 seconds, in terms of memory it uses 1280000 to 2048000 bits (5*16000*16 to 8*16000*16) to represents an individual utterance in the system. Each wave file is represented by a unique name as described in Table 1.

Table 1. Naming structure & format used to represent an individual utterance in Corpus 1

Speakerid_deviceid_utterancenumber.wav	
Parameters	**Range**
Speaker Id	1 to 8
Device Id	0 to 2
Utterance number	1 to 100

Corpus 2

Table 2. Naming structure and format used to represent an individual utterance in Corpus 2

Speakerid_deviceid_language_recordingtext_utterancenumber.wav	
Parameters	**Range**
Speaker Id	1 to 31
Device Id	1 to 2
Language	English: eng; Hindi: hindi
Recording Text	Pangram: pan; Spontaneous: intro
Utterance number	1to 2

Bilingual Mobile Speech Corpus (Palia et al. 2017) has consisted of speech samples recorded from thirty-one Hindi native speakers. Corpus 2 comprises of 372 utterances recorded simultaneously in both Hindi and English languages using two mobile phones of different make in offline mode in a room environment. Pangram and spontaneous text in both the languages have been used as recording text. After recording the data has been transferred to a system and segregated using Gold Wave audio editor. The number of bits required to store a speech sample varies from 256000 to 5120000 bits due to the varied lengths of each wave file from 1 to 20 seconds. To perform the experiment each speech sample has been represented by a unique name as explained in Table 2.

FEATURE EXTRACTION AND CLASSIFICATION

Feature extraction is a technique to procure speaker-specific information from the speech signal. In the proposed work MFCC has been extracted as a feature. Classification is a process of mapping the features vectors of a speaker training model to identify the input utterance SVM has been used as a classifier.

Mel Frequency Cepstral Coefficient

MFCC is based on human hearing instead of the linear scale. It is a descriptor of the short-term power spectrum of the speech signal by implementing linear cosine transform of the log power spectrum of Mel Scale (Davis and Mermelstein 1980; Deller, Hansen, and Proakis 2000; Jain and Tripathi 2018). MFCC involves the following six steps:

1. The acquired speech signal is first digitized and then pre-processed to amplify high-frequency formats, which has suppressed during the sound production mechanism of a human. Using equation 1 the speech signal is pre-processed with $\alpha = 0.95$.

$$s(n) = s(n) - \alpha \times s(n-1) \tag{1}$$

where's' is the speech signal and α is the pre–emphasis factor.

2. Pre-processed speech signal is converted into the frames. The frame length is 20ms, with 50% overlap with consecutive frames. There are 100 frames per second.

3. After framing the speech signal is smoothed by Hamming Window (Nuttall 1981) using equation 2, in order to maintain the continuity of the first and the last point of the frame.

$$w(n,\alpha)=(1-\alpha)-\alpha\cos\left(\frac{2n\alpha}{N-1}\right) \tag{2}$$

$0 \leq n \leq N-1$, where $\alpha = 0.46$ and $N = 20$ ms

The speech signal *s* (*n*) after Hamming Window *w* (*n*) is represented by equation 3:

$$s(n)=s(n)\times w(n) \tag{3}$$

4. The Fast Fourier Transformation is performed on speech signal s (n). It is used to measure the power spectral estimation as represented by equation 4:

$$s(n)=\sum_{n=0}^{n=N-1} s(n)e^{-j2\Pi nk/N} \tag{4}$$

5. Map the power spectrum on the mel scale, using the set of triangular band- pass filter. The number of band- pass filters varies from 20 to 35. Filters are uniformly implemented along the mel-frequency. The relation between linear and mel frequency is represented in equation 5.

$$mel(f)=1125\times ln\left(1+\frac{f}{700}\right) \tag{5}$$

6. Mel scale cepstral coefficients are calculated by applying discrete cosine transformation of log filter bank energies (Huang, Acero and Hon 2001)

$$C_m = \sum_{k=1}^{n} \cos\left[m \times (k - 0.5) \times \Pi / N\right] \times E_k \quad (6)$$

where m = 1, 2 ……. L and L = 12.

12 MFCC features vector is extracted to make model feasible for real-time implementation.

Support Vector Machine

Support Vector Machine is a discriminative slow classifier with high accuracy used to classify both linear and nonlinear unseen data (Cortes and Vapnik 1995) which is implemented with spectral, high level and prosodic features (Shriberg et al. 2005; Ferrer et al. 2007).

SVM binary classifiers make the decision on the basis of equation (7)

$$g(x)=w\cdot\varphi(x)+b \quad (7)$$

where φ(x) is an on-linear function that maps vector x to a high dimension feature space of linearly separable classes,with w,as separating hyper-planeinspace.

SVM is resolved by presenting limitations in minimizing function using Lagrange multipliers, leading to the maximization of the Wolfe dual

$$L_D = \sum_{i=1}^{N} a_i - \sum_{i=1}^{N} \sum_{j=1}^{N} a_i a_j y_i y_{ij} x_i^T x_j \quad (8)$$

An SVM (Bengio 1995) is a two-class classifier constructed from sums of a kernel function K (·,·),

$$g(x) = \sum_{i=1}^{N} \lambda_i y_i K(x_i x) + b \quad (9)$$

where the y_i are the ideal outputs,

$$\sum_{i=1}^{N} \lambda_i y_i = 0 \text{ and } \lambda_i > 0 \quad (10)$$

The vectors x_i are support vectors obtained from the training set by an optimization process (Bourlard and Morgan 1994). The ideal outputs are either 1 or −1, depending upon whether the corresponding support vector is in class 0 or class 1, respectively. For classification, a class decision is based upon whether the value, g(x), is above or below a threshold.

SVM is mainly a binary classification (Cortes and Vapnik 1995) and requires fixed dimension input. Martin et al. (2005) proposed Hidden Markov Model (HMM) based segmentation for fixing fixed dimension to isolated recognition. SVM apply sparse and discriminant technique to return same hyper plane parameters. It solves over fitting issues and groups multiple features to have maximal margin classifier (Awad and Khanna 2015). The kernel is used as inner product in feature space to have large margin hyperplane. The margin of hyper plane is identified by support vectors achieved by optimization considering Langarangian relaxation. The kernel trick is used for mapping data in higher dimensional space before optimization for maximum distance of different classes.

The most widely used kernel functions are the gaussian radial basis function,

$$K\left(x_i, x_j\right) = \exp\left(-\frac{||x_i - x_j||^2}{2\gamma^2}\right) \tag{11}$$

with an associated feature space of infinite dimensionality, and the polynomial kernel

$$\mathrm{K}\left(\mathrm{x_i}, \mathrm{x_j}\right) = \left(1 + \mathrm{x_i}.\mathrm{x_j}\right)^{\mathrm{p}} \tag{12}$$

which associated feature space are the polynomials up to grade p.

The problem of multiclass classification can be solved considering one-vs-one (OVO), one-vs-all (OVA) (Hsu and Lin 2002). In the OVA method k SVMs (where k is the number of classes) are constructed whereas in the OVO k(k −1)/2 classes are constructed. In OVO approach the computational effort is less as compared to OVA as a number of training vectors is less (Hsu and Lin 2002; Huang, Acero& Hon 2001).

SVM can be applied in various applications like disease, weather and stock prediction, power estimation, defect classification, handwriting identification, image and audio processing, speaker and speech recognition, etc. or in other words, it has a wide range of applications.

SVM classifiers provides a suitable solution to speaker identification (Kamruzzaman et al. 2007, Campbell et al. 2006) automatic speech recognition which essentially is a pattern recognition problem as SVM is very well adapted to high-dimensional classification problems. (Martın et al. 2005).

Mittal and Sharma (2015) applied two-step classification using one versus all SVM classifier using linear predictive coding and MFCC for recognition of Hindi numerals having 96.8% accuracy. Campbell et al. (2006) have applied SVM for speaker and language identification. Besbes and Lachin (2016) developed a multiclass SVM for automatic stressed recognition system based on kernel classification.

Karam and Campbell (2008) extended the maximum likelihood linear regression and Gaussian mixture model framework to the multiclass case on the NIST SRE 2006 corpus. It has been observed that multi-class MLLR improves on global MLLR and enhances the performance of the system.

In this chapter SVM based on maximum margin has been applied for speech recognition as a multiclass classification problem. Speech has a high dimensionality sequence vector of featured vectors that need to be normalized as the kernel can deal with vectors of fixed size. The sequence of feature vector generated in speech is of variable length as different speakers take different time to speak the same text, due to the different durations of acoustic units and constant frame rate. A normalizing kernel has been used to achieve the adaptation and results show superior discrimination ability of SVM (Ganapathiraju, 2002; Smith and Gales 2002). Garc´ıa-Cabellos et al. (2004) provided a solution for dimension normalization in the case of non-uniform distribution instances generated from internal states of HMM.

The extracted acoustic features (MFCC) are given as input to the SVM. The classification is performed in two steps. In the first step, a one-versus-all (OVA) SVM classifier is used to identify the language and in the second step, it is used to recognize the speaker. The RBF kernel (eq. (12)) has been used in all the experiments, finding values for γ and regularization parameter C of the SVM.

In OVA k binary classifier are created, having estimated training time as per power law (Platt 1998) for OVA is defined as

$$TrainingTime_{OVA} \approx k \propto M^2 \text{where M is training set of classes} \quad (13)$$

Considering each k class has an equal number of training samples OVO requires 2M/k samples.

$$TrainingTime_{OVO} \approx\propto \frac{k(k-1)}{2}\left(\frac{2M}{k}\right)^2 \approx 2 \propto M^2 \quad (14)$$

In the testing phase, OVA require k binary SVM evaluations and OVO necessitate requires $\frac{k\,((k-1)}{2}$ binary SVM evaluations. (Sidaoui and Sadouni 2011).

Model Evaluation Using Confusion Matrix

To estimate how accurately the classifier identified the speaker, a confusion matrix (Furui 2007) has been considered for model evaluation. Table 3 shows the basic structure of the MXM confusion matrix.

On the basis of multidimensional confusion matrix, the following parameters have been evaluated as:

- True Positive (TP) for a class: TP_m
- TP for a system: $TP_1 + TP_2 + TP_3 + \ldots + TP_{m-1} + TP_m$
- True Negative (TN) for a class: Sum of all rows and columns except that class column and row.
- TN for a system: $TN_1 + TN_2 + TN_3 + \ldots + TN_{m-1} + TN_m$
- Total number of test sample per class: $TP_m + TN_m$

- False Negative (FN) for a class: Sum of all values in the corresponding row except TP of a class
- False Positive (FP) for a class: Sum of all values in the corresponding columns except TP of a class.
- Positive (P) sample of actual class: $TP_m + FN_m$
- Negative (N) sample of actual class: $TN_m + FP_m$
- Positive (P') samples of predicted class: TP_m+FP_m
- Negative (N') sample of predicted class: TN_m+FN_m

Using above described parameters, in the current study the performance of the system has been evaluated on the basis of several performance parameters such as accuracy/recognition rate (Mishra and Lotia 2014), sensitivity/recall (Furui 2007), precision and F1 score (Hansen and Hasan 2015). An elaborate explanation of each parameter is provided in the next section.

Table 3. Basic structure of MXM Confusion Matrix

	Predicted Class								
	Class	1	**2**	**3**	**4**	**. . . .**	**m-2**	**m-1**	**M**
Actual Class	1	TP1	E12	E13	E14	. . . .	E1(m-2)	E1(m-1)	E1m
	2	E21	TP2	E23	E24	. . . .	E2(m-2)	E2(m-1)	E2m
	3	E31	E32	TP3	E34	. . . .	E3(m-2)	E3(m-1)	E3m
	4	E41	E42	E43	TP4	. . . .	E4(m-2)	E4(m-1)	E4m
	.	.	.	.	.	. . . .	.	.	.
	.	.	.	.	.	. . . .	.	.	.
	.	.	.	.	.	. . . .	.	.	.
	m-2	E(m-2)1	E(m-2)2	E(m-2)3	E(m-2)4	. . . .	TP(m-2)	E(m-2)(m-1)	E(m-2)m
	m-1	E(m-1)1	E(m-1)2	E(m-1)3	E(m-1)4	. . . .	E(m-1)(m-2)	TP(m-1)	E(m-1)m
	M	Em1	Em2	Em3	Em4	. . . .	Em(m-2)	Em(m-1)	TPm

TP: True Positive, E: Error, TPm: m class is classified as m, Em(m-1): m is classified as (m-1)).

Performance Parameters

Accuracy: the percentage of correctly classified samples.

$$Accuracy = \frac{TP + TN}{P + N} \tag{15}$$

Error Rate: the percentage of misclassified samples.

$$error\ rate = 1 - accuracy \tag{16}$$

True positive rate/recall (TPR): The percentage of positive tuples that correctly identified.

$$true\ positive\ rate = \frac{TP}{P} \tag{17}$$

True negative rate (TNR): The percentage of negative tuples that correctly identified.

$$true\ negative\ rate = \frac{TN}{N} \tag{18}$$

False positive rate (FPR): the percentage of negative tuple misclassified as positive.

$$false\ positive\ rate = \frac{FP}{N}\ or\ 1 - true\ negative\ rate \tag{19}$$

False negative rate (FPR): the percentage of positive tuple misclassified as negative.

$$false\ negative\ rate = \frac{FN}{P}\ or\ 1 - true\ positive\ rate \tag{20}$$

Precision: the measure of exactness.

$$\Pr ecision = \frac{TP}{TP + FP} \tag{21}$$

F_1 score: harmonic mean of precision and true positive rate.

$$F_1 = \frac{2 \times precision \times recall}{precision + recall} \tag{22}$$

Proposed Algorithm

Table 4. Proposed algorithm

Algorithm (MFCC-SVM)
Initialization: s: speech signal fs: sampling frequency (16 kHz) n: number of bits per sample (16 bits) a: pre-processing factor w: Hamming window
Step 1. Record the speech samples using different devices and store it into speaker folder for segregation. Step 2. Segregate the speech sample on the basis of recording language, recording text and utterance after removing unvoiced using the audio editor. Step 3. Name it and save into respective speaker folder. Step 4. Load a wave file (s, fs, n) Step 5. Pre-process the signal (s, fs, n, a) Step 6. Split it into frames frame size: 20 ms distance between two frames :10 ms Step 7. Apply Hamming window (s, w) Step 8. Extract 12 MFCC features and store it into excel file Features= data (1:12), Number of frames Step 9. Interpolate MFCC values Step 10. Features (0.01,0.02,0.001,0.002) Step 11. Load mfcc features file Step 12. Split the feature and label into training_set and testing_set 66% of data = training_set 33% of data = testing_set Step 13. Train SVM model Step 14. Predict values for testing set Step 15. Prepare Confusion Matrix (TP, TN, FP, FN) Step 16. Generate classification report (accuracy, true positive rate, true negative rate, error rate, precision, F1 ratio)

The proposed algorithm comprised of feature extraction followed by classification techniques is presented below in Table 4. The captured analog signal is converted and stored in digital form for further processing. MFCC features are extracted from the speech signal. The extracted features are then classified using SVM. The final decision is taken on the basis of various Performance parameters evaluated using a confusion matrix.

EXPERIMENTS AND RESULTS

Seventeen sets of experiments have been performed on both the SC to study the impact of recording language, recording device, size of speech corpus, length of speech corpus and number of utterances on the accuracy of SR. All the experiments have been conducted on a system with configuration Intel core i5, 5th generation processor with 8GB DDR3 RAM and 1TB of disk storage and confusion matrix has been prepared for each experiment. On the basis of the confusion matrix, a classification report has been generated. The details of these experiments have been elaborated below.

Experiment on Corpus 1

Experiment 1: Trained and Tested on Speech Samples Collected from Device 0

In this experiment, 67 percent speech sample of each speaker recorded using device 0 has been used to train the model. Remaining 33% of the dataset of device 0 has been utilized for testing purposes. The system has taken 25 hours and 15 minutes to create training models from 67% of the dataset. The confusion matrix for this experiment is tabulated in Table 5.

Based on Table 5 the parameters of the Sp1 have been measured as:

- TP of Sp1: 16550
- TN of Sp1: Sum of all the values excluding values in row 1 and column 1 of Table 5 as shown in Table 6.
- FP of Sp1: 1923+1541+399+533+880+1023+1225 =724
- FN of Sp1 =2319+2592+976+952+1174+936+1328 =10277.

Table 5. Confusion Matrix for Experiment 1 trained and tested using speech sample recorded from device 0 of Corpus 1

	Predicted Values								
Actual Values	SpId	Sp1	Sp2	Sp3	Sp4	Sp5	Sp6	Sp7	Sp8
	Sp1	16550	2319	2592	976	952	1174	936	1328
	Sp2	1923	21134	3004	1032	1196	2704	824	661
	Sp3	1541	2036	28069	1856	1564	3006	2006	1330
	Sp4	399	713	1193	31433	1131	1307	733	367
	Sp5	533	834	1612	1266	29518	478	1111	442
	Sp6	880	2341	2405	1606	448	25622	1731	1300
	Sp7	1023	1060	3245	1298	1402	6619	12494	1069
	Sp8	1225	927	2439	1048	777	3517	1008	20184

Sp-Speaker.

Table 6. Representing the values which constitute True Negative of Sp1 for Experiment 1 trained and tested using speech sample recorded from device 0 of Corpus 1

Sp Id	Sp2	Sp3	Sp4	Sp5	Sp6	Sp7	Sp8
Sp2	21134	3004	1032	1196	2704	824	661
Sp3	2036	28069	1856	1564	3006	2006	1330
Sp4	713	1193	31433	1131	1307	733	367
Sp5	834	1612	1266	29518	478	1111	442
Sp6	2341	2405	1606	448	25622	1731	1300
Sp7	1060	3245	1298	1402	6619	12494	1069
Sp8	927	2439	1048	777	3517	1008	20184

Table 7. Speaker wise details for Experiment 1 trained and tested using speech sample recorded from device 0 of Corpus 1

Sp Id	TP	TN	FP	FN	P= TP+FN	N= FP+TN	P’ = TP+FP	N’ = TN+FN
Sp1	16550	235100	7524	10277	26827	242624	24074	245377
Sp2	21134	226743	10230	11344	32478	236973	31364	238087
Sp3	28069	211553	16490	13339	41408	228043	44559	224892
Sp4	31433	223093	9082	5843	37276	232175	40515	228936
Sp5	29518	226187	7470	6276	35794	233657	36988	232463
Sp6	25622	214313	18805	10711	36333	233118	44427	225024
Sp7	12494	232892	8349	15716	28210	241241	20843	248608
Sp8	20184	231829	6497	10941	31125	238326	26681	242770
Total	185004	1801710	84447	84447	269451	1886157	269451	1886157

TP: True Positive; TN: True Negative; FP: False Positive; FN: False Negative; P: Positive sample of actual class; N: Negative sample of actual class; P’: Positive sample of predicted class; N’: Negative sample of predicted class.

Following the same steps, the values of the parameters for the other speakers have been computed as shown in Table 7.

Table 7 summarized the details of every speaker for experiment 1. Using the values tabulated in Table 7 the value of performance parameters for every speaker has been computed. The computed values of performance parameters have been tabulated in Table 8.

Table 8. Performance Parameters for every speaker and system for the Experiment 1 trained and tested using speech sample recorded from device 0 of Corpus 1

Sp Id	Accuracy	Error rate	TPR	TNR	FPR	FNR	Precision	F1 Score
Sp1	93.39%	6.61%	0.62	0.97	0.03	0.38	0.69	0.65
Sp2	91.99%	8.01%	0.65	0.96	0.04	0.35	0.67	0.66
Sp3	88.93%	11.07%	0.68	0.93	0.07	0.32	0.63	0.65
Sp4	94.46%	5.54%	0.84	0.96	0.04	0.16	0.78	0.81
Sp5	94.90%	5.10%	0.82	0.97	0.03	0.18	0.80	0.81
Sp6	89.05%	10.95%	0.71	0.92	0.08	0.29	0.58	0.63
Sp7	91.07%	8.93%	0.44	0.97	0.03	0.56	0.60	0.51
Sp8	93.53%	6.47%	0.65	0.97	0.03	0.35	0.77	0.70
Average	92.16%	7.84%	0.69	0.96	0.04	0.31	0.67	0.69

Table 9. Training and testing conditions for experiment 2 and 3 performed on Corpus 1

Exp. No.	Experimental Condition				Time
	Training		Testing		
	Device	Size	Device	Size	
2	Device 1	67%	Device 1	33%	27 hours
3	Device 0	100%	Device 1	100%	27 hours

Table 10. Performance of the system for all the experiments conducted on Corpus 1

Performance Parameters	Experiment 1	Experiment 2	Experiment 3
Accuracy	92.16%	91.90%	82.41%
Error Rate	7.84%	8.10%	17.59%
FPR	0.04	0.05	0.12
TPR	0.69	0.67	0.48
TNR	0.96	0.94	0.88
FNR	0.31	0.33	0.52
Precision	0.69	0.67	0.48
F1	0.69	0.67	0.48

Similarly, two more experiments have been conducted on Corpus 1. The experimental condition of experiment 2 and 3 is tabulated below in Table 9. One more experiment has been conducted on the smaller set for Device 2 and it is not giving results up to mark. The average value of performance parameters for all three experiments conducted on Corpus 1 has been displayed in Table 10.

On the basis of the experiments performed on Corpus 1, it has been observed that the impact of the recording devices is very high. The accuracy of the system degrades approximately by 10% in the case of experiment 3 as compared to experiments 1 and 2. The reason for this degradation is due to speech samples used for training a model and testing have collected using different devices. The same scenario has been visualized for the other parameters. The results for experiments 1 and 2 are close to each other. The performance of the system declines for the speech samples recorded using a mobile phone as compared to the speech samples recorded using microphones.

It has also been observed that training time and memory required to process a speaker identification system with the above conditions is not feasible in real-time if more speakers are added. Keeping this in mind the same experiments have been performed on speech Corpus 2 with the limited number of utterances but the number of speakers has been increased by four times as compared to speech Corpus 1.

Experiment Performed on Corpus 2

The first experiment performed on Corpus 2 is to study how much minimum number of utterances is required to achieve a good accuracy of the system.

Experiment: To Study the Impact of the Number of Utterances on the Accuracy of the System

This experiment has been performed on speech samples recorded through device 1 for English pangram text. MFCC features vectors have been extracted from each utterance. To increase the number of training samples MFCC features vectors per frame for both the utterances are extrapolated by the value 0.01, 0.02, 0.001, 0.002, 0.003, and 0.03 for each speaker.

Figure 1 shows the trend of the accuracy of the system for the different number of utterances when trained and test with English speech samples for device 1 of Corpus 2. It has been observed that there is a remarkable increase in the accuracy of the system until the number of utterances is reached 10. After 10 utterances there is only 0.009% change in accuracy for 12 utterances and 0.01% for 14 utterances. On the basis of the above experiment performed on Corpus 2 and Corpus 1, it has been concluded that system performance is ultimately dependent on the number of utterances. Figure 2 has shown a relationship between time-space and accuracy of the system. From Figure 2, it is concluded that if the number of utterances increased the time and space required increase linearly but the accuracy of the system becomes saturated after a few utterances. So, on the basis of this for all the experiments performed on Corpus 2, MFCC features vectors for each utterance have been extrapolated by these values 0.01, 0.02, 0.001 and 0.02 to attain MFCC features vectors for 10 utterances.

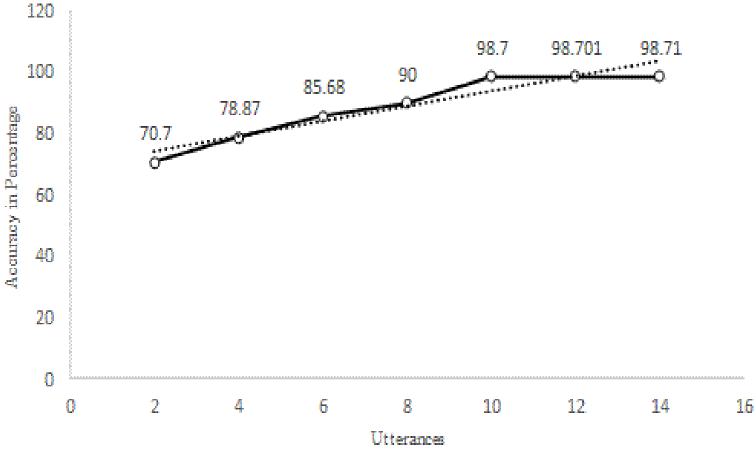


Figure 1. Accuracy trend of Corpus 2 on device 1.

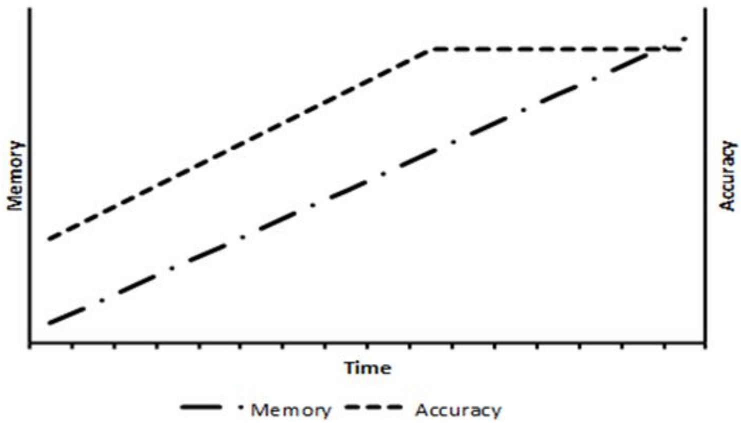


Figure 2. A plot displaying the Time – Space – Accuracy trade off.

Experiment: Impact of Recording Devices and Recording Languages

12 different experiments have been performed on Corpus 2. The training and testing conditions as well as result for all the experiments performed on Corpus 2 has been shown in table 11. These experiments have been performed to study the impact of recording devices and recording languages on the accuracy of the system.

These experiments have been further categorized into 4 parts:

- *Recording Language Dependent and Recording Device Dependent*: In this group of experiments (1 to 4) the speech samples used for training and testing belongs to the same language and recorded using the same device. 67% of the data set has been used for training and left 33 percent for testing.
- *Recording Language-Independent and Recording Device Dependent*: In this instance of experiments (5 to 8) the speech samples recorded in one language have been used for training while testing samples have been belongs to another language. The recording device is the same for testing and training.
- *Recording Language Dependent and Recording Device Independent*: Training and testing speech samples of the same language have been used while the recording device used in both phases is different as shown in experiment number 9 to 10. This experiment has been carried to analyze the impact of recording devices on the accuracy of the system.
- *Recording Language-Independent and Recording Device Independent*: The experimental condition for experiment numbers 11 and 12 are to train and tested with speech samples recorded using different languages and different devices.

On the basis of all these experiments, it has been found that the maximum accuracy attained by the system is 98.73% for device-dependent and language-dependent in case of the English language while for the

Table 11. Performance Parameters of Experiment performed on Corpus 2

Exp No		**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
		Language Dependent and Device Dependent				**Language In-Dependent and Device Dependent**				**Language Dependent & Device Independent**		**Language Independent & Device Independent**	
Training Condition	Device	1	1	2	2	1	2	2	1	1	1	1	2
	Language	Hindi	English	Hindi	English	English	English	Hindi	Hindi	English	Hindi	Hindi	Hindi
	Size	67%	67%	67%	67%	100%	100%	100%	100%	100%	100%	100%	100%
Testing Condition	Device	1	1	2	2	1	2	2	1	2	2	2	1
	Language	Hindi	English	Hindi	English	Hindi	Hindi	English	English	English	Hindi	English	English
	Size	33%	33%	33%	33%	100%	100%	100%	100%	100%	100%	100%	100%
Performance Parameter	Time	6 hrs	13 min	5 hrs	20 min	40 min	13 min	6 hrs 10 min	5hrs	20 min	5 hrs	4 hrs 20 min	5 hrs 40 min
	Accuracy in percentage	98.70	98.73	97.25	98.01	87.94	84.40	73.80	91.21	85.76	82.20	73.26	85.57
	Error Rate in Percentage	1.30	1.27	2.75	1.99	12.06	15.60	26.20	8.79	14.24	17.80	26.74	14.43
	FPR	0.02	0.01	0.02	0.01	0.09	0.11	0.18	0.06	0.08	0.10	0.15	0.08
	TPR	0.65	0.83	0.62	0.76	0.27	0.22	0.13	0.33	0.25	0.23	0.14	0.28
	TNR	0.98	0.99	0.98	0.99	0.92	0.89	0.82	0.94	0.92	0.90	0.85	0.92
	FNR	0.35	0.17	0.38	0.24	0.73	0.78	0.87	0.67	0.75	0.77	0.86	0.72
	Precision	0.65	0.83	0.62	0.76	0.27	0.22	0.13	0.33	0.27	0.23	0.15	0.28
	F1	0.65	0.83	0.62	0.76	0.27	0.22	0.13	0.33	0.26	0.23	0.14	0.28

Hindi language it attained an accuracy of 98.7%. Accuracy degrades by approximately 7% in the case of language-independent or device-independent as shown in experiment number 5 to 10 while the performance of the system further reduced by 14% approximately in experiments 11 & 12 in case of language-independent and device-independent. When the system is trained using the Hindi language it performs better then system trained with English this is because the data set size is more for the Hindi Language as compared to English confusion matrix.

CONCLUSION AND FUTURE SCOPE

In this chapter, the *Bilingual Speaker Recognition* system has been evaluated using MFCC-SVM approach. 17 different sets of experiments have been performed on two different SC. The accuracy of a system using MFCC-SVM for Bilingual is remarkably good. Designing of Automatic speaker recognition system is a tough task due to the number of factors affecting human speech like emotion, recording device, language, background noise, session variability, health etc. To build a robust speaker identification system there is a need to develop a large size speech corpus which includes speech sample recorded in a different environment, using different recording devices and collect samples in the different time span.

REFERENCES

Agrawal, Shyam, S. 2010. "Recent Developments in Speech Corpora in Indian Language." *Paper presented at Oriental-COCOSDA Conference*, Kathmandu, Nepal, November 24-25.

Agrawal, Shyam, S., Sinha, Shweta, Singh, Pooja & Olsen, Jesper. 2012. "Development of text and speech database for Hindi and Indian English specific to mobile communication environment" *Paper* *presented at International Conference on the Language Resources and Evaluation Conference*, Turkey, May 21-27.

Antonsen, Jorgan, Johan. 2017 "*Open Set Speaker Identification*" Master of Science in Electronics Department of Electronic Systems, Norwegian University of Science and Technology.

Awad, Mariette and Khanna, Rahul. 2015. "Support Vector Machines for Classification." In *Efficient Learning Machines*, 39-66.Berkely: Apress.

Beck, B., Neuberg, E. & Hodge, D. 1977. "An assessment of the technology of automatic speech recognition for military applications." *IEEE Transactions Acoustics, Speech, Signal Processing*, 25(4), 310-322.

Bengio, Yoshua.1996. *Neural networks for speech and sequence recognition*. Canada: International Thomson Computer Press.

Besbes, Salsabil and Lachiri, Zied. 2016, Multi-class SVM for stressed speech recognition, *Paper presented at International Conference on Advanced Technologies for Signal and Image Processing*, Monastir, Tunisia, March 21-23.

Bourlard, Herve and Morgan, Nelson. 1994. *Connectionist Speech Recognition: A Hybrid Approach*, Dordrecht: Kluwer Academic Publisher.

Bricker, P. D., Gnanadesikan, R., Mathews, M.V., Pruzansky, S., Tukey, P.A. Wachter, K.W. & Warner, J.L. 1971. "Statistical techniques for talker Identification." *The Bell System Technical Journal* 50(4), 1427-1454.

Burton, David. 1987. "Text-dependent speaker verification using vector quantization source coding." *IEEE transaction Acoustics, Speech, Signal Processing* 35(2), 133-143.

Campbell, William, Campbell, Joseph, P. Reynolds, Douglas, A., Jones, Douglas, A. &Leck, Timothy, R. 2004 "Phonetic speaker recognition with support vector machines." In *Advances in Neural Information Processing Systems*, edited by Thrun, S., Saul, L., Schokopf, (16).Cambridge: MIT Press.

Campbell, William. Campbell, Joseph, P. Reynolds, Douglas, A. Singer, Elliot & Torres-Carrasquillo, Pedro, A. 2006. “Support Vector Machines for speaker and language recognition.” *Computer Speech Language* 20(2-3), 210-229.

Cortes, Corinna. &Vapnik, Vladimir. 1995. “Support-Vector Networks.” *Machine Learning* 20(3), 273-297.

Davis, Steven, B. & Mermelstein, Paul. 1980. “Comparison of parametric representation for monosyllabic word recognition in continuously spoken sentences.” *IEEE Transaction Acoustic, Speech, Signal Processing* 28(4), 357-366.

Deller, John, R. Hansen, John, H. L. & Proakis, John, G. 2000. *Discrete-Time Processing of Speech Signals.* New York: IEEE Press.

Doddington, George, R. 1971. “A method of speaker verification.”*The Journal of the Acoustical Society of America* 49, 139 (A).

Ferrer, Luciana, Shriberg, Elizabeth, Kajarekar, Sachin & Sonmez, Kemal.(2007). “Parameterization of prosodic feature distributions for SVM modelling in speaker recognition.” *Paper presented at International Conference on Acoustics, Speech, and Signal Processing*, Honolulu, HI, USA, April 15-20.

Furui, Sadaoki. 1981. “Cepstral analysis technique for automatic speaker verification” *IEEE Transaction Acoustics, Speech, Signal Processing*. 29(2), 254-272.

Furui, Sadaoki. 2007. “Speech and speaker recognition evaluation” In *Evaluation of Text and Speech Systems* edited by L. Dybkjær, J. Hamsen and W. Minker, 1-27. Dordrecht: Springer.

Furui, Sadaoki. (2009). “40 Years of Progress in Automatic Speaker Recognition” *ICB Proceedings of third International Conference on Advances in Biometric,* Italy, June 2-5.

Ganapathiraju, Aravind. 2002. “*Support vector machines for speech recognition.*” PhD Thesis, Mississipi State Universisty.

Garcıa-Cabellos, Jose, Miguel, Pel´aez-Moreno, Carmen, Gallardo-Antolın, Ascension, Pérez-Cruz, Fernando, and Díaz de María, Fernando. 2004. “SVM classifiers for ASR: A discssion about parameterization." *Proceedings of European Signal Processing Conference*, Vienna, Austria, September 6-10.

Gu, Yong & Thomas, Trevor. 2001. "A Text-Independent Speaker Verification System using Support Vector Machines Classifiers." *Paper presented at 7th European conference on Speech Communication and Technology*, Aalborg, Denmark, September 3-7.

Hansen, John, H. L. &Hasan, Taufiq. 2015. "Speaker Recognition by Machines and Humans: A Tutorial Review." *IEEE Signal Processing Magazine*. 32(6), 74-99.

Hsu, Chih-Wei and Lin, Chih-Jen. 2002. "A comparison of methods for multi-class support vector machines." *IEEE Transactions on Neural Networks.* 13(2), 415-425.

Huang, Xuedong. Acero, Alex. & Hon, Hsiao-Wuen. 2001. *Spoken Language Processing: A Guide to Theory, Algorithm and System Development* New Jersey: Prentice-Hall.

Huang, Tzu-Kuo, Weng, Ruby, C. and Lin, Chih-Jen. 2006. "A generalized Bradley-terry model and multiclass probability estimates."*Journal of Machine Learning Research* 7:85-115.

Jain, Vinay, Kumar & Tripathi, Neeta. 2018. "Multilingual Speech Analysis using Cepstrum Coefficient Features for Text-Dependent Speaker Identification System." *International Journal of Pure and Applied Mathematics* 118(16), 707-722.

Kamruzzaman, S.M., Karim, Rezaul, A.N.M., Islam, Md. Saiful&Haque, Md. Emdadul. 2007. "Speaker Identification using MFCC-Domain Support Vector Machine" *International Journal of Electrical and Power Engineering* 1, 274-278.

Karam, Zahil, N. and Campbell, William, M. 2008. "A multi-class mllr kernel for svm speaker recognition." *IEEE International Conference on Acoustic, Speech and Signal processing*, Las Vegas, USA, March 31-April 4.

Kinnunen, Tomi & Li, Haizhou. 2010. "An Overview of Text- Independent Speaker Recognition: from Features to Supervectors." *Speech Communication* 52, 12- 40.

Martin-Iglesias. D., Bernal-Chaves. J., Pelaez,-Moreno. C., Gallardo-Antolin. A. and Diaz-de-Maria, A, 2005, A Speech Recognizer based on Multiclass SVMs with HMM-Guided Segmentation. In *Nonlinear Analyses and Algorithms for Speech Processing.Lecture Notes in Computer Science*, edited by Faundez-Zanuy M., Janer L., Esposito A., Satue-Villar A., Roure J., Espinosa-Duro V., 257-266. Berlin: Springer.

Mishra, Piyush & Lotia, Piyush. 2014. "Comparative performance analysis of SVM Speaker Verification System using Confusion Matrix." *International Journal of Science and Research* 3(12), 1419-1422.

Mittal, Teena and Sharma, Rajendra, Kumar. 2015. "Multiclass SVM based Spoken Hindi Numerals Recognition." *TheInternational Arab Journal of Information Technology* 12(6), pp 666-671.

Nijhawan, Geeta & Soni, M.K. 2014. "Speaker Recognition using Support Vector Machine." *International Journal of Computer Applications*. 87(2), 7-10.

Nivedita, Ahmed, P., Dev, Amita & Agrawal, Sahyam, S. 2013. "Hindi Speech Corpora: A Review." *Paper presented at International Conference Oriental COCOSDA jointly with Conference on Asian Spoken Language Research and Evaluation*, O-COCOSDA/CASLRE, Gurgaon, India, November 25-27.

Nuttall, Albert, H. 1981. "Some Windows with very good Side Lobe Behaviour." *IEEE Transactions on Acoustics, Speech, Signal Processing*. 29, 84–91.

Palia, Nivedita, Ravulakollu, Kiran, Dev, Amita & Kamthania, Deepali. 2017. "Bilingual Mobile Speech Corpus" *International Conference on Computing for Sustainable Global Development*, Delhi, March 1-3.

Platt, John, C. 1998 "Fast training of support vector machines using sequential minimal optimization." In *Advances in Kernel Methods - Support Vector Learning*, 185-208.Cambridge: MIT Press.

Pruzansky, Sandra. 1963. "Pattern-matching procedure for automatic talker recognition." *The Journal of the Acoustical Society of America* 35, 354-358.

Rose, Richard, C. & Reynolds, Douglas, A. 1990. “Text independent speaker identification using automatic acoustic segmentation.” *Paper presented at International conference on Acoustics, Speech, and Signal Processing*, Albuquerque, NM, USA, April 3-6.

Sambur, Marvin, R. 1972. “*Speaker recognition and verification using linear prediction analysis.*” PhD diss., M.I.T.

Sidaoui, Boutkhil and Sadouni, Kaddour. 2011. “Efficient Approach Multiclass SVM for Vowels Recognition.” *CEUR Workshop Proceedings*.

Smith, N., D. and Gales, Mark, J., F. 2002. “Using SVMs and discriminative models for speech recognition.” *Paper presented at IEEE International Conference on Acoustics, Speech and Signal Processing*, Orlando, FL, USA, May 13-17.

Shrishrimal, Pukhraj, P., Deshmukh, Ratnadeep, R. & Waghmare, Vishal, B. 2012. “Indian Language Speech Database: A Review.” *International Journal of Computer Applications* 47 (5), 17 – 21.

Sinha, Shweta, Agrawal, Sahyam, S. & Olsen, Jesper. 2011. “Development of Hindi Mobile Communication Text and Speech Corpus.” *Paper presented at International Conference on Speech Database and Assessment*, Hsinchu, Taiwan, October 26-28.

Soong, Frank, K. Rosenberg, A. E. Juang, Biing, Hwang & Rabiner, L.R. 1985. “A vector quantization approach to speaker recognition.” *Paper presented at IEEE International Conference on Acoustic, Speech, and Signal Processing*, Tampa, FL, USA, April 26-29.

Shriberg, Elizabeth, E., Ferrer, Luciana, Kajarekar, Sachin, S., Venkataraman, Anand & Stolcke, Andreas. 2005. “Modeling prosodic feature sequences for speaker recognition.” *Speech Communication* 46(3-4), 455-472.

Vibhhute, Anup & Hibare, Rekha. 2015. “Speaker Identification System for Hindi and Marathi Languages using Wavelet and Support Vector Machine.” *International Journal on Recent and Innovation Trends in Computing and Communication* 3(4), 2198-2201.

BIOGRAPHICAL SKETCHES

Nivedita Palia

Affiliation: School of Information Technology, VIPS, Delhi, India

Education: PhD student

Business Address: AU Block, Outer Ring Road, Pitampura, Delhi-34, India

Research and Professional Experience: 12 years

Professional Appointments: Assistant Professor

Honors: She has received Rajya Purksar from Haryana State in girls guide and scout.

Publications from the Last 3 Years:

Books:

1. Nivedita and Aakansha, *SSC Tier-1 CGL 35 Practice Sets*, MTG Publication, ISBN: 9789387747319
2. Nivedita and Aakansha, *SSC Tier-1 CHSL 25 Practice Sets*, MTG Publication, ISBN: 9789387747524
3. Nivedita and Aakansha, *SSC Tier-1 MTS 25 Practice Sets*, MTG Publication, ISBN: 9789387949270

International Journals:

1. Palia, N., Shri Kant and Dev, A. (2019) Performance Evaluation of Speaker Recognition System. *Journal of Discrete Mathematical Science and Cryptography, Taylor and Francis*. 22(2), 203-218. Available from: doi: 10.1080/09720529.2019.1582868
2. Palia, N., Shri Kant, Dev, A., and Khullar, S. (2018) Feature Extraction and Classification Techniques for Speaker Recognition. *Journal of*

Advanced Research in Dynamical and Control Systems, 10-11, 826-835.

International Conferences:

1. Palia, N., Kamthania, D. and Ashish "Mobile based Bilingual Speech Corpus", *Three Days International Conference on Smart Sustainable Intelligent Computing and Applications*, Delhi, 4th - 6th Feb, 2020.
2. Palia, N., Shri Kant and Dev, A. Multimodal Biometric System :A Review *International Conference on Artificial Intelligence and Speech Technology (AIST 2019)* 14-15 November, 2019, Delhi, India, pp.217-223.
3. Palia, N., Ravulakollu, K., Dev, A. and Kamthania, D. "Bilingual Mobile Speech Corpus", INDIACom-2017, 4th International Conference on "Computing for Sustainable Global Development, Delhi, 01st -03rd March, 2017, ISBN: 978-93-80544-24-3, pp. 642:647.

Prof. Deepali Kamthania

Affiliation: School of Information Technology, VIPS, Delhi, India

Education: PhD (IITD)

Business Address: AU Block, Outer Ring Road, Pitampura, Delhi, India

Research and Professional Experience: 19 years

Professional Appointments:
PhD Supervisorat Mewar University, Ghaziabad and Amity University, Rajasthan. Recognized Reviewer, International Journal of Computers and Electrical Engineering, Elsevier, International Journal of Energy and Buildings, Elsevier, International Journal Engineering Science and Technology, Elsevier, International Journal of Advanced Intelligence Paradigm (IJAIP), Inderscience.

Honors:

She has received Merit scholarship in B. Sc. and Bharat Seva Scholarship in MCA. She is the recipient of Academic Excellence Awards. Best Paper Award in international conference in 2019, Best Teacher Award in 2018 and 2019, Active Participation Award (Woman) in 2016-17 from Computer Society of India and Institution of Engineering Technology Awards in 2013 and 2014 for significant contribution to the IT field. Dr. Kamthania is a life time member of IEEE, CSI and ISTE.

Publications from the Last 3 Years:

Book Chapters:

1. “A Novel Framework for Intelligent Spaces” *Internet of Things in Business Transformation: Developing an Engineering and Business Strategy for Industry 5.0* in publication by Scrivener Publishing LLC Wiley (SCOPUS INDEXED).
2. *“Intrusion Detection and Security System, Big Data Analytics and Intelligence: A Perspective for Health Care*, in publication Emerald Group Publishing, U.K.
3. “Fundamentals of Solar Photovoltaic Technology”, *Renewable Energy Sources & Environment Protection (International Edition)*, International Research Publication House, 2018, ISSN: 978-93-87388-19-2, pp. 449-464.
4. “Hybrid Photovoltaic Thermal Systems”, *Renewable Energy Sources & Environment Protection (International Edition)*, International Research Publication House, 2018, ISSN: 978-93-87388-19-2, pp. 465-487.
5. “Study and Comparative Analysis of Various Image Spamming Technique, Soft Computing: Theories and Applications, *Advances in Intelligent Systems and Computing*, Springer, ISSN: 2194-5357, Vol. 583, pp-351-365, April 2018. Online ISBN978-981-10-5687-1, https://doi.org/10.1007/978-981-10-5687-1_3.

International Journals:

1. "A Novel Approach to Retrieve Unlabeled Images", *International Journal of Business Intelligence and Data Mining*, Inderscience, Scopus Indexed, H index-17 in publication.
2. "Quantitative Analysis of Historical Data for Prediction of Job Salary in India-A Case Study", *Journal of Statistics & Management Systems*, Taylor & Francis, Volume 22 (2), pp. 187-198, 2019, ISSN: 2169-0014, ESCI.
3. "Sentiment Analysis for Finding Factors Effecting Social Trends for Indian Taxi Aggregator", *Journal of Advanced Research in Dynamical and Control Systems*, Institute of Advanced Scientific Research, Scopus Indexed, Vol.10 (11), 508-519, Sep, 2018, ISSN: 1923-023X.
4. "Market Segmentation Analysis and Visualization using K-Mode Clustering Algorithm for E-Commerce Business", *Journal of Computing and Information Technology*, Vol. 26, No. 1, March 2018, pp. 57–68 DOI: 10.20532/cit.2018.1003863, Scopus Indexed, ISSN: 13301136. H index-22, University of Zagreb.
5. "Smart Cities for Sustainable Development: Smarter Solutions for Better Tomorrow", *International Journal of Innovations & Advancement in Computer Science IJIACS* ISSN 2347 – 8616 Vol. 6(9), pp. 185-191, September 2017.
6. "Educational Data Mining – A Case Study", *International Journal of Information and Decision Sciences, Inderscience,* indexed in: Scopus (Elsevier), ISSN: 1756-7025, Vol. 8(2), pp. 187-201, May, 2017.

International Conferences:

1. "ANN Model for Forest Cover Classification", *International Conference on Data Analysis and Management (ICDAM-2020),* 18th June, 2020. To be published in Lecture Notes on Data Engineering and Communications Technologies, Springer.
2. Visual Analytics on Public Grievance Data using Tableau, *Proceedings of the International Conference on Innovative Computing & Communications (ICICC 2020)*, 18th June, 2020.

3. "Negator: Hate Speech Detection", *Three Days International Conference on Innovative Computing and Communication*, Delhi, 21st -23rd February, 2020,
4. "Mobile based Bilingual Speech Corpus", *Three Days International Conference on Smart Sustainable Intelligent Computing and Applications*, Delhi, 4th - 6th Feb, 2020.
5. "Education 4.0 The journey so far", *31st AIMS Annual Management Convention 2019 on Management Education 4.0: Strategies for Global Competitiveness*, Delhi, 29th -31st August, 2019.
6. "Impact of Smart Devices", *Two Days International Conference on Transforming IDEAS (Inter-Disciplinary Exchanges, Analysis and Search) into Viable Solutions*, 29th -30th March, 2019, pp. 310-317, ISBN: 938882695-7, Macmillan Education
7. "Intelligent Object Detection and Avoidance System", *Two Days International Conference on Transforming IDEAS (Inter-Disciplinary Exchanges, Analysis and Search) into Viable Solutions*, 29th -30th March, 2019, pp.342-351, ISBN: 938882695-7, Macmillan Education.
8. "Sentiment Analysis: Google Playstore's Applications", *IEEE International Conference on Signal Processing, VLSI and Communication Engineering (ICSPVCE-2019)*, Delhi Technological University, 28th -30th March, 2019. To be published in IEEE Explore.
9. Shivam Mukherjee and Deepali Kamthania, "Creating 3D Facial Model from a 2D Facial Image", ICCCS 2018; *2nd International Conference on Communication and Computing Systems*, Dronacharya College of Engineering, Gurgaon, 01st – 2nd December, 2018, Published Proceeding of Communication and Computing Systems, Taylor & Francis Group, London, ISBN: 978-0-367-00147-6, pp. 52-57, 2019.
10. "Bilingual Mobile Speech Corpus", INDIACom-2017, *4th International Conference on "Computing for Sustainable Global Development*, Delhi, 01st -03rd March, 2017, ISBN: 978-93-80544-24-3, pp. 642:647.
11. Reliable Facades for Sustainable Environment, *3rd International Conference on Computational Intelligence & Communication*

Technology (CICT), 10th -11th Feb, 2017, Ghaziabad, published in IEEE Xplore Digital Library. ISSN 978-1-5090-6218-8, pp. 1-5. DOI: 10.1109/CIACT.2017.7977352.

Magazine:

1. "Ancient Indian (Vedic) Intellectual Heritage Knowledge-base: Preserving Heritage Language", *CSI Communications-Knowledge Digest for IT Community*, ISSN 0970-647X, Vol. 37 (12), pp. 26-28, December 2018.

National Conferences:

1. "Database Attacks" National Conference NGCTND 2020, Ghaziabad, 20th -21st March, 2020. To be published in *International Journal of Computer Applications*, NY, USA.
2. An Implementation Based Comparative Analysis of Various Vulnerabilities in Online Forms", *TECHNO TRYST 2017: "Digital Transformation: Computational and Technological Advancements"*, Delhi, 25th March, 2017, ISBN: 987-81-927406-8-3, pp-58-69.
3. "Auto Metro Train to Shuttle Between Stations", *National Conference on Recent Trends in Electronics & Electrical Engineering (NCRTEEE-2017)*, Ghaziabad, 16th -17th February, 2017, Excellent Publication, ISBN: 978-93-86238-13-9, pp. 74-78

Dr. Shri Kant

Affiliation: Sharda University

Education: PhD (Mathematics)

Business Address: Room No. 111, Block No. 1, 32,34 Knowledge Park-III, Sharda University, Greater Noida, Gautam Budh Nagar, PIN-201306, India

Research and Professional Experience: More than 40 years

Professional Appointments: Professor

Honors: Recipient of several commendation certificates from SA to RM and best scientist of the year Lab award for exhibiting the excellence in pattern recognition application to cryptology.

Publications from the Last 3 Years:

1. Prabhat Kumar Ray, Shri Kant, Bimal K. Roy, and Ayanendranath Basu., Classification of Encryption Algorithms using Fisher's Discriminant Analysis, *Defence Science Journal*, Vol. 67, No. 1, January 2017, pp. 59-65.
2. Olaleye S. B., Ranjan I. and Ojha S. K., SoloEncrypt: A smartphone storage enhancement security model for securing users' sensitive data, *Indian Journal of Science and Technology*, Vol. 10, No. 8, pp 1-8, February 2017. DOI: 10, 17485/ijst/2017/v10i8/ 104511, ISSN (print):0974-6846, ISSN (online) 0974-5645.
3. Olaleye S. B. and Ojha S. K., Compact software implementation of AES on atomic smartphones architecture, *International Journal of Innovative Science, Engineering and Technology*, Vol. 4, Issue 2, February 2017, pp. 102-112.
4. Olaleye, S. B. and Ojha S. K., A security study to compare cryptographic algorithms based on performance in mobile cloud computing, *International Conference on Recent Research in "Mechanical, Electrical, Electronics, Civil, Computer Science and Information Technology" (MECIT-2017)*, Jawaharlal Nehru University, New Delhi, India, 22nd January 2017.
5. Olaleye, S. B. and Ojha S. K. Improved Advanced Encryption Using Four Square Cipher for User Anonymity and Untraceability in Mobile Cloud Computing, *Int. J. Innov. Sci. Eng. Technol* 4 (2), 113-121
6. Alok Katiyar, Monica Singh, Shri Kant (2018) "A Self Regulating Model to relieve DDoS Intrusion in Pervasive Computing Environment," *IEEE Conference ID: 42835 at International Conference on "Computing for Sustainable Global Development, INDIACom*-2018 on 14-March 2018. (Scopus Indexed)

7. Alok Katiyar, Suman Kumari, Shri Kant, (2018) "Towards a Visionary Software Framework for Integrating Context-Based Security in Pervasive Computing Smart Home Environments" at *International Journal of Computing for Sustainable Global Development*, p-2606-2612.ISSN 0973-7529; ISBN 978-93-80544-28-1 (Open Access, Google Scholar)
8. Alok Katiyar, Monica Singh, Shri Kant, (2018) "A Self Regulating Model to relieve DDoS Intrusion in Pervasive Computing Environment," *International Journal of Computing for Sustainable Global Development*, p2601-2605, ISSN 0973-7529; ISBN 978-93-80544-28-on 14-March 2018. (Open Access, Google Scholar)
9. Alok Katiyar, Rishi Sharma, Shri Kant, YDS Arya(2018)," Pervasive Online Spam Review Detection based on Ontology using Naïve Bayesian" *IEEE 2018 International Conference on Advances in Computing, Communication Control, and Networking, ICACCCN* 2018 (Scopus Indexed)
10. Solomon Babatunde Olaleye and Shri Kant: " Secure Use of Cloud Storage of Data on Smartphone's using Atomic AES on ARM Architectures" *International Journal of Applied Engineering Research* ISSN 0973-4562 Volume 13, Number 5 (2018) pp. 2569-2581.
11. Adamu Bin Ibrahim and Shri Kant: "Penetration Testing Using SQL Injection to Recognize the Vulnerable Point on Web Pages" *International Journal of Applied Engineering Research* ISSN 0973-4562 Volume 13, Number 8 (2018) pp. 5935-5942.
12. Nivedita Palia, Shri Kant, Amita Dev "Features Extraction and Classification Techniques for Speaker Recognition" *Journal of Adv Research in Dynamical & Control Systems*, Vol. 10, 11-Special Issue, 2018, ISSN 1943-023X 826.
13. Sinha J., Ravulakollu, K.K, and Kant, S., "Software Development Approaches Significant for Runtime Software Evolution: A Review" *IEEE International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*, 2018.
14. Katembo Kituta Ezéchie, Shri Kant, Ruchi Agarwal: (2018) "Analysis of Database Replication Protocols" International Journal of Latest

Trends in Engineering and Technology, Special Issue ICRM-2018, pp 075-083, e-ISSN: 2278-621X. Presented in: *ICRMR-2018: International Conference on Recent Multidisciplinary Research*, TKP Conference centre, Kuala Lumpur, Malaysia, 22-23 June 2018.

15. Kituta, E. K., Kant, S. and Agarwal, R. (2018). Serialization of updates broadcasting and reconciliation of lazy replicated databases over a decentralized Peer-to-Peer architecture. (Accepted for Journal Publication) Presented in: *FOBE-2018: Third International Conference on Facets of Business Excellence*, IMT Ghaziabad, India, Nov 22-26 2018.
16. Sinha J. and Kant, S "Performance Tuning and Evaluation of Fuzzy Agent Model using ANFIS for Consumer-Relationship Management" *International Journal of Engineering Research and Technology*. ISSN 0974-3154 Volume 11, Number 12 (2018), pp. 2183-2197.
17. Bindu Rani, and Shri Kant "A Comparative Study of Big Data Analytics into Decision Making Process" *IEEE 8th International Advance Computing Conference (IACC 2018)*, 14-15th Dec. BENNETT University, Greater Noida, Poster presentation.
18. Alok Katiyar, Subhi Shukla and Shri Kant (2019), " Smart Mobile Device Authentication using Keystroke Dynamics Based behavior Classification" *International Journal of Scientific Research and Review (IJSRR)*, Vol.7, Issue 03, March 2019, ISSN No. 2279-543x, UGC Serial No: 64650.
19. Katembo Kituta Ezéchiel, Shri Kant, Ruchi Agarwal: "A Systematic Review on Distributed Databases Systems and Their Techniques" *Journal of Theoretical and Applied Information Technology*, Jan 2019, Vol.96, No.1, pp 236-266.
20. Harshit Sharma and Shri Kant: "Early Detection of Ransom ware by Indicator Analysis and Win-API Call Sequence Pattern" Satapathy S., Joshi A. (eds) *Information and Communication Technology for Intelligent Systems. Smart Innovation, Systems and Technologies*, 2019, Vol 107. pp 201-211, Springer, Singapore
21. Nivedita Palia, Shri Kant and Amita Dev "Performance Evaluation of Speaker Recognition System" 2019, *Journal of Discrete Mathematical*

Sciences and Cryptography, Taylor & Francis Group. Vol. 22, No. 2, pp 203-218.

22. Nomaun Rathore, Shri kant "Effective Role of Block Chain Technology for E-Commerce: A Case Study in Pub/Sub" *International Journal of Research in Engineering, Science and Management*, Volume-2, Issue-3, pp 90-93, March-2019 www.ijresm.com | ISSN (Online): 2581-5792.
23. Ifra Salaudin, Shri Kant, Supriya Khaitan "Block Chain and its Application in Electronic Health Records" *International Journal of Research in Engineering, Science and Management*, Volume-2, Issue-3, pp 137-139, March-2019 www.ijresm.com | ISSN (Online): 2581-5792.
24. Hagos, M.T. and Kant, S., (2019). *Transfer learning based Detection of Diabetic Retinopathy from Small Dataset*. arXiv preprint arXiv:1905.07203.
25. Kituta, E. K., Kant, S. and Agarwal, R. (2019). A synchronizer-mediator for lazy replicated databases over a decentralized P2P architecture. *2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*, ISBN: 978-1-7281-4826-7/19/$31.00 ©2019 IEEE, pp 199-213.
26. Deepali Salwan and Shri Kant "Role of Artificial Intelligence and Machine Learning in Resolving the Issues and Challenges with Prosthetic Knees" Springer Nature Singapore Pte Ltd. 2019, A. B. Gani et al. (Eds.): *ICICCT 2019, CCIS*, 1025, pp. 147–155, https://doi.org/10.1007/978-981-15-1384-8_12
27. Amrita and Shri Kant "Machine learning and feature selection approach for anomaly based intrusion detection: a systematic survey and a novice approach" *International Journal of Innovative Technology and Exploring Engineering (IJITEE)* ISSN: 2278-3075, Volume-8 Issue-6S, pp434-443, April 2019.
28. Bindu Rani, and Shri Kant "A Roadmap from Unsupervised Clustering to Semi Supervised Clustering" *Proceeding of International Conference on Contemporary Engineering and Technology 27th* and 28th APRIL, 2019,pp 29, Springer journal. Advs in Intelligent Syst.,

Computing, Vol. 1005: New Paradigm in Decision Science and Management, 978-981-13-9329-7, 469650_1_En, (19).

29. Bindu Rani and Shri Kant (2019), "A research on Big data clustering with improvisation in K Means Clustering using Semi Supervised Clustering" presented in *ICCET 2019 (International Conference on Contemporary Engineering and Technology)* 2019, Inderprastha Engineering College, Ghaziabad. on 28th April, 2019 and published in International Journal of Innovative Research in Applied Sciences and Engineering (IJIRASE) Volume 2, Issue 11, DOI: 10.29027/ IJIRASE.v2.i11.2019, 362-367, May 2019.
30. S.A Bala, S. Kant and K. Kumar," Deep Learning: From Medical Imaging to Drug Design ", 2019, *Proc. National Conference on Recent Trends and Advancements in Chemical Sciences (RTACS 2019)*, pp.68.
31. S.A Bala and S. Kant, " Deep Learning Based Model Architectures for Cardiac MRI Segmentation: A Survey", 2019, *Proc. 2nd International Conference on Recent Trends in Science, Engineering and Management (ICRTSEM 2019)*, pp.29-33.
32. S.A Bala, S. Kant and K. Kumar, "Impact of Deep Learning in Medical Imaging: A systematic Survey with New Proposed Model", 2019, Presented, *International Conference on Recent Advancements in Management, Science and legal issues ICRAMSTEL* 2019.
33. Kituta, K., Kant, S. Agarwal, R. (2019). A New Non-Blocking Validation Protocol for Eager Replication of Databases over a Decentralized P2P Architecture, *International Journal of Recent Technology and Engineering (IJRTE)*, Blue Eyes Intelligence Engineering & Sciences Publication, 8(2S7), 575-591.
34. Katembo Kituta Ezéchiel, Shri Kant & Ruchi Agarwal (2019) Mediation of Lazy Update Propagation in a Replicated Database over a Decentralized P2P Architecture, *Global Journal of Computer Science and Technology*, 19(3), 1-31.
35. Suman Goyat, Shri Kant and Neha Dhariwal (2019) "Dynamic Heterogeneous scheduling of GPU-CPU in Distributed Environment", *Proc. of Second International Conference on Smart Systems and Inventive Technology (ICSSIT 2019)*, pp: *332-339*, DVD Part Number:

CFP19P17-DVD; ISBN: 978-1-7281-2118-5, 978-1-7281-2119-2/19/$31.00 ©2019 IEEE.

36. Godfrey Gandawa and Shri Kant (2019), “The Adaptation of Emerging Technologies in Healthcare and Biomedical Research in SADC”, *Proc. of Second International Conference on Smart Systems and Inventive Technology (ICSSIT 2019)*, pp: 561-564, DVD Part Number: CFP19P17-DVD; ISBN: 978-1-7281-2118-5, 978-1-7281-2119-2/19/$31.00 ©2019 IEEE.
37. Godfrey Gandawa and Shri Kant (2019), “Artificial Intelligence (AI) Chronic Ailment Management Algorithm: *The SmartMeds Hub Proc. of Second International Conference on Smart Systems and Inventive Technology (ICSSIT 2019)*, pp: 565-568, DVD Part Number: CFP19P17-DVD; ISBN: 978-1-7281-2118-5, 978-1-7281-2119-2/19/$31.00 ©2019 IEEE.
38. Godfrey Gandawa and Shri Kant (2019) A Blended Diabetes Diagnostic Algorithm The Use of Logistic Regression & ANN in Python” *Biomedical Research* 2019; 30 (5): ISSN 0970-938X, www.biomedres.info.
39. Godfrey Gandawa and Shri Kant (2019) Diagnosis and treatment of diabetes using artificial neural network, logistic regression and natural language processing: A physician approach. *Biomedical Research* 2019; 30 (5): ISSN 0970-938X, www.biomedres.info.
40. Misgina Tsighe Hagos, Shri Kant and Surayya Ado Bala (2019) “ Automated Smartphone Based System for Diagnosis of Diabetic Retinopathy, *International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*, ISBN: 978-1-7281-4826-7/19/$31.00 ©2019 IEEE, pp 256-261
41. Katembo Kituta Ezéchiel, Shri Kant Ojha, Ruchi Agarwal (2020) “A New Eager Replication Approach Using a Non-Blocking Protocol over a Cecentralized P2P Architecture” *International Journal of Distributed Systems and Technologies, Special Issue : Distributed Computing and Systems & Applications*, 11(2), pp. 69-100. Indexed in ESCI, Scopus, Compendex,

42. Bindu Rani, and Shri Kant (2020) “An Approach Toward Integration of Big Data into Decision Making Process” Springer Nature, © Springer Nature Singapore Pte Ltd. 2020, S. Patnaik et al. (eds.), New Paradigm in Decision Science and Management, *Advances in Intelligent Systems and Computing* 1030, pp 207-215, https://doi.org/10.1007/978-981-13-9330-3-19.
43. Ifra Salaudin, Shri Kant, Supriya Khaitan “Application of Block Chain in Ehrs for Maintaining the Privacy of Patients Record” presented in TEMT-2019 and Accepted for Publication in journal Lecture Notes in Mechanical Engineering, a Springer publication. Abstract in Springer *souvenir of International Conference on Emerging Trends in Electro-Mechanical Technologies and Management (TEMT 2019)*, pp43
44. Nomaun Rathore, Shri kant “Enhanced Blockchain Application For Pub-Sub model” presented in TEMT-2019 and Accepted for Publication in journal Lecture Notes in Mechanical Engineering, a Springer publication. Abstract in Springer souvenir of *International Conference on Emerging Trends in Electro-Mechanical Technologies and Management (TEMT 2019)*, pp42
45. Rajendra Kumar, Ram Chandra Singh and Shri Kant, (2020) Dorsal Hand Vein-Biometric Recognition using Convolution Neural Network, *Advances in Intelligent Systems and Computing*, DOI: 10.1007/978-981-15-5113-0_92 (in press) Scopus
46. Edward Nwiah and Shri Kant (2020): “Securing Payload In MQTT Protocol Using Ultra- Lightweight Cryptography” Accepted *International Journal of Recent Technology and Engineering*™ *(IJRTE)*.
47. Bindu Rani and Shri Kant (2020) “A Cluster Validity Index Combining Compactness Measure and Overlap Measure For Fuzzy Clustering” *Journal Communicated Int. J. of Business Intelligence and Data Mining*.
48. Bindu Rani and Shri Kant (2020) “Semi-Supervised Learning Based Churn Prediction Model for Decision Making In Telecom Sector” Accepted *International Journal of Computer Information Systems and Industrial Management Applications (IJCISIM)*.

49. Rajendra Kumar, Ram Chandra Singh, And Shri Kant (2020) "Dorsal Hand Vein Recognition Using Very Deep Learning" Accepted for presentation in MIDAS-2020 *International Conference on Machine Intelligence and Data Science Applications*. IIPM Dehradun: and for publication in "Algorithms for Intelligent Systems" by Springer.

In: Support-Vector Machines
Editor: Pooja Saigal
ISBN: 978-1-53618-757-1

Chapter 6

Application of Support Vector Machine (SVM) in Classification of Iron Ores in Mines

***Ashok Kumar Patel*[1]*, Snehamoy Chatterjee*[2]**
***and Amit Kumar Gorai*[3,*]**

[1]Department of Computer Science and Engineering,
C.V. Raman Global University, Bhubaneswar, Orissa, India
[2]Department of Geological and Mining Engineering and Sciences,
Michigan Technological University, MI, US
[3]Department of Mining Engineering, National Institute of Technology,
Rourkela, Orissa, India

Abstract

Support vector machine (SVM) is a non-parameterized based supervised classification algorithm. SVM has a strong potential for learning a machine vision system for quality monitoring in various types of industries, including mining industries. In this study, an SVM-based algorithm was designed for the prediction of the quality of iron ores based on the chemical composition. For designing and validating the SVM-based classifier for the proposed application, samples of iron ore were collected from a mine and analyzed in the laboratory for determining the compositions. Based on the composition, ores were classified into five different classes. A total of 812 image samples were captured with the consideration of uniform representations of each class of ore. For classification, 280 image features (color and texture-based information) were take out from each of the captured image and subsequently, principal component analysis (PCA) was conducted to reduce the feature dimension for SVM-based classification model development. The model was optimized using a radial basis kernel function with the parameter box constrains at 388.2 and kernel scale at 0.0011. It was optimized in one vs. one coding technique for multiclass classification. The model performance was examined using the four indices viz. accuracy, sensitivity, misclassification, and specificity. All these indices were determined from confusion matrix parameters. The study results indicated that the accuracy, sensitivity, misclassification and specificity of the PCA-based SVM model were 0.9951, 0.9913, 0.0049 and 0.9966, respectively for testing samples. Therefore, the proposed model can be successfully used for quality monitoring in mines.

Keywords: support vector machine, iron ore classification, feature dimension reduction

INTRODUCTION

With the advance of computing power and learning algorithm, the system is able to work as an expert in the developed domain area. The industries of the world are more dependent on this system for reducing humane intervention in the monotonous and uninterrupted application. These nonstop tasks reduce humane capability and prone to error in identifying the problems. The breakdown of the industry for an hour will make a loss of huge amount. The supply of varying quality of product to the client make a huge penalty. Poorly controlled production makes a huge loss due to poor quality product development.

In the mining industry, regular ore quality monitoring is essential to supply specific quality of ore to the client industries as per agreement. If the quality is less then mentioned, the client may slap a huge penalty, and if the quality is high, the loss will be induced by selling on cheap cost. Therefore, regular quality monitoring is must need in the mining industry. In the industry, sophisticated instruments can be installed, which is costly in installation and maintenance. Therefore, an alternative solution is proposed by many researchers using machine vision-based technology[1].

The machine vision-based technology is incorporated with the process of image acquisition, pre-processing, feature selection, feature optimization, and machine learning. The machine learning model performance has purely dependent on the capability of the algorithm and system[2]. It has started with perceptron learning[3], also known as a single neuron. Later in 1960, Widrow and Hoff [4] developed ADALINE (ADAptive LInear NEuron) using the least mean square (LMS) learning procedure. In 1962, Rosenblatt introduced the feedforward neural network algorithms with the perceptron rule [5]. This network was further extended with the backpropagation of error into the multilayer network[6]. Then the

---

[1] Claudio A. Perez et al., "Ore Grade Estimation by Feature Selection and Voting Using Boundary Detection in Digital Image Analysis," *International Journal of Mineral Processing* 101, no. 1–4 (November 2011): 28–36, https://doi.org/ 10.1016/j.minpro. 2011.07.008; Snehamoy Chatterjee, "Vision-Based Rock-Type Classification of Limestone Using Multi-Class Support Vector Machine," *Applied Intelligence* 39, no. 1 (July 12, 2013): 14–27, https://doi.org/10.1007/s10489-012-0391-7; Francisco J. Galdames et al., "Classification of Rock Lithology by Laser Range 3D and Color Images," *International Journal of Mineral Processing* 160 (March 2017): 47–57, https://doi.org/10.1016/ j.minpro.2017.01.008; Francisco J Galdames et al., "Rock Lithological Classification by Hyperspectral, Range 3D and Color Images," *Chemometrics and Intelligent Laboratory Systems*, 2019, https://doi.org/10.1016/j.chemolab.2019.04.006; Ashok Kumar Patel, Snehamoy Chatterjee, and Amit Kumar Gorai, "Development of Machine Vision-Based Ore Classification Model Using Support Vector Machine (SVM) Algorithm," *Arabian Journal of Geosciences* 10, no. 5 (March 1, 2017): 107, https://doi.org/10.1007/s12517-017-2909-0.

[2] Nicu Sebe et al., *Machine Learning in Computer Vision*, vol. 29, Computational Imaging and Vision (Berlin/Heidelberg: Springer-Verlag, 2005), https://doi.org/10.1007/1-4020-3275-7.

[3] F Rosenblatt, "1," *Psychological* The Perceptron: A Probabilistic Model for Information Storage and Organization In *The Brain Review*, vol. 65, 1958, https://www.ling.upenn.edu/ courses/cogs501/Rosenblatt1958.pdf.

[4] B Widrow and M E Hoff, "Adaptive Switching Circuits," in *1960 IRE WESCON Convention Record, Part 4* (New York: IRE, 1960), 96–104, https://books.google.co.in/ books?id=Vs4EAAAAIAAJ.

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

[6] P J Werbos, *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences* (Washington, D.C.: Harvard University, 1975), https://books.google.co.in/ books?id=z81XmgEACAAJ.

inference engine is developed with the Bayesian algorithm for classification[7]. After that induction algorithm, named ID3, was developed by Quinlan to generate a classification rules-based decision tree to classify the objects correctly[8]. Later ID3 is extended to develop a new algorithm named C4.5[9]. At the same time, the support vector machine (SVM) and adaptive neuro-fuzzy inference system (ANFIS) has been introduced in the machine learning community[10].

The support vector machine has shown its efficiency in the linearly and nonlinearly separable problem by obtaining maximum margin hyperplane[11]. SVM was designed as a binary classifier but subsequently used as a multiclass classifier using one vs. rest and one vs. one method[12]. In the past, the SVM algorithm was used by many researchers[13] for different purpuses. Mining industry problems were also solved using

---

[7] Judea Pearl, "Reverend Bayes on Inference Engines: A Distributed Hierarchical Approach," in *Proceedings of the Second AAAI Conference on Artificial Intelligence*, AAAI'82 (AAAI Press, 1982), 133–36, http://dl.acm.org/citation.cfm?id=2876686.2876719.

[8] J.Ross Quinlan, "Learning Efficient Classification Procedures and Their Application To Chess End Games," in *Machine Learning* (Elsevier, 1983), 463–82, https://doi.org/10.1016/B978-0-08-051054-5.50019-4.

[9] J R Quinlan, *C4.5: Programs for Machine Learning*, Morgan Kaufmann Series in Machine Learning (Elsevier Science, 1993), https://books.google.co.in/books?id=HExncpjbYroC.

[10] J.-S.R. Jang, "ANFIS: Adaptive-Network-Based Fuzzy Inference System," *IEEE Transactions on Systems, Man, and Cybernetics* 23, no. 3 (1993): 665–85, https://doi.org/10.1109/21.256541; Vladimir N Vapnik, "The Nature of Statistical Learning Theory," 1995.

[11] Vladimir Vapnik, "Pattern Recognition Using Generalized Portrait Method," *Automation and Remote Control* 24 (1963): 774–80; Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik, "A Training Algorithm for Optimal Margin Classifiers," in *Proceedings of the Fifth Annual Workshop on Computational Learning Theory - COLT '92* (New York, New York, USA: ACM Press, 1992), 144–52, https://doi.org/10.1145/130385.130401.

[12] Vladimir Naumovich Vapnik and Vlamimir Vapnik, *Statistical Learning Theory*, vol. 1 (Wiley New York, 1998); Chih Wei Hsu and Chih Jen Lin, "A Comparison of Methods for Multiclass Support Vector Machines," *IEEE Transactions on Neural Networks* 13, no. 2 (March 2002): 415–25, https://doi.org/10.1109/72.991427.

[13] Yu-Chieh Wu, Yue-Shi Lee, and Jie-Chi Yang, "Robust and Efficient Multiclass SVM Models for Phrase Pattern Recognition," *Pattern Recognition* 41, no. 9 (September 2008): 2874–89, https://doi.org/10.1016/j.patcog.2008.02.010; Giorgos Mountrakis, Jungho Im, and Caesar Ogole, "Support Vector Machines in Remote Sensing: A Review," *ISPRS Journal of Photogrammetry and Remote Sensing* 66, no. 3 (May 2011): 247–59, https://doi.org/10.1016/j.isprsjprs.2010.11.001; Sujay Raghavendra. N and Paresh Chandra Deka, "Support Vector Machine Applications in the Field of Hydrology: A Review," *Applied Soft Computing* 19 (2014): 372–86, https://doi.org/10.1016/j.asoc.2014.02.002; Rommel M. Barbosa et al., "Recognition of Organic Rice Samples Based on Trace Elements and Support Vector Machines," *Journal of Food Composition and Analysis* 45 (2016): 95–100, https://doi.org/10.1016/j.jfca.2015.09.010.

various machine learning algorithms and SVM has shown its strength for effective use in various problems[14].

Machine learning algorithms are dependent on the size and relevancy of the features used in the model development[15]. With the advancement of technology, a large number of image data can be easily generated and processed, but there is a high possibility that the dataset contains many irrelevant and redundant data. Feature selection or feature dimension reduction is one of the solutions by reproducing the maximum variability of data with a small dimension[16]. In the last few decades, many feature selection or dimension reduction methods like Principal component analysis (PCA), sequential forward selection (SFS), sequential forward floating selection (SFFS), etc. have been developed by various reseatrchers. All these methods are either based on the filter-based or wrapper-based or embedded-based algorithm. In the filter method, the selection of features can be made separately using any criterion function and the selected features can be used in a separate machine learning algorithm, and thus, the selected features can be used in model development using any algorithm[17]. In the wrapper method, the optimum subset of features is selected using the induction algorithm as part of the evaluating function[18]. The embedded-based feature selection algorithm is integrated as part of the learning algorithm. PCA is a widely used feature

---

[14] Perez et al., "Ore Grade Estimation by Feature Selection and Voting Using Boundary Detection in Digital Image Analysis"; Chatterjee, "Vision-Based Rock-Type Classification of Limestone Using Multi-Class Support Vector Machine"; Patel, Chatterjee, and Gorai, "Development of Machine Vision-Based Ore Classification Model Using Support Vector Machine (SVM) Algorithm"; Ashok Kumar Patel, Snehamoy Chatterjee, and Amit Kumar Gorai, "Development of a Machine Vision System Using the Support Vector Machine Regression (SVR) Algorithm for the Online Prediction of Iron Ore Grades," *Earth Science Informatics*, November 9, 2018, https://doi.org/10.1007/s12145-018-0370-6; Galdames et al., "Rock Lithological Classification by Hyperspectral, Range 3D and Color Images."

[15] Mineichi Kudo and Jack Sklansky, "Comparison of Algorithms That Select Features for Pattern Classifiers," *Pattern Recognition* 33, no. 1 (January 2000): 25–41, https://doi.org/10.1016/S0031-3203(99)00041-2.

[16] Mikhail Belkin and Partha Niyogi, "Laplacian Eigenmaps for Dimensionality Reduction and Data Representation," *Neural Computation* 15, no. 6 (June 2003): 1373–96, https://doi.org/10.1162/089976603321780317.

[17] Kudo and Sklansky, "Comparison of Algorithms That Select Features for Pattern Classifiers."

[18] Ron Kohavi and George H. John, "Wrappers for Feature Subset Selection," *Artificial Intelligence* 97, no. 1–2 (1997): 273–324, https://doi.org/10.1016/S0004-3702(97)00043-X.

dimension reduction technique, reproducing the feature by linear approximation[19]. Thus, the present study demonstrates the performance of the PCA-based SVM algorithm in iron ore classification.

METHODS

Sample Collection

It is very important to suitably select the proper sampling technique to obtain the heterogeneous samples in order to include the maximum variation of the characteristics. It is important to include all the variations in the model analysis for reducing the error levels[20]. For the current study, samples of iron ore were collected using a stratified random sampling method from the Gua Iron Ore Mine. In the stratified random sampling method, the ore samples were taken from different locations of the ore deposit to represent the ore heterogeneity exist in the reserves of the iron ore deposit. The mine is located in the district of Paschim Singhbhum of Jharkhand state in India (shown in Figure 1).

Image Acquisition

Iron ore samples collected from the mine was used for image acquisition. Images of iron ore samples were captured in a fabricated image acquisition system. The set-up consists of a motor to drive the roller of conveyor and HD web camera to capture the images[21]. The speed of

[19] I.T. Jolliffe, *Principal Component Analysis*, Springer Series in Statistics (New York: Springer-Verlag, 2002), https://doi.org/10.1007/b98835.

[20] D. D. Sarma, *Geostatistics with Applications in Earth Sciences* (Dordrecht: Springer Netherlands, 2009), https://doi.org/10.1007/978-1-4020-9380-7.

[21] Ashok Kumar Patel, Snehamoy Chatterjee, and Amit Kumar Gorai, "Development of Online Machine Vision System Using Support Vector Regression (SVR) Algorithm for Grade Prediction of Iron Ores," in *2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA)* (IEEE, 2017), 149–52, https://doi.org/10.23919/

conveyor was maintained in such a way that the captured images have less distortion. The images were captured in an isolated environment in order to reduce the distortion. All the images were passed through the adaptive median filter to remove the noise if any exists, even after maintaining the isolated environment[22].

Feature Extraction

An image 'I' is considered to be a rectangular arrangement of pixels of size p × q. Pixels are having a range of intensity value f(x, y) for each spatial coordinates x ϵ 0,1,…,p and y ϵ 0,1,…,q. Binary image have f(x, y) either 0 or 1, while grayscale image have f(x, y) ranges from 0 to 255. Images were captured in the visible bands and thus consisted of red (R), green (G) and blue (B) components but for the higher number of information extractions, images can be captured in multiple bands. For example, multispectral images or hyperspectral images may have multiple bands and thus, information can be extracted separately for each band. In the present study, images were captured in the RGB color space model. RGB color space is additive, and it means all the colors can be generated by the addition of different intensities of different bands[23].

Every image store unique information in the pixels depending on the characteristics of the objects. Thus the information needs to be extracted for the recognition of the objects. The color and texture features of the images were extracted for image analysis and classification. In the past, the

MVA.2017.7986823; Ashok Kumar Patel, Snehamoy Chatterjee, and Amit Kumar Gorai, "Development of an Expert System for Iron Ore Classification," *Arabian Journal of Geosciences* 11, no. 15 (August 27, 2018): 401, https://doi.org/10.1007/s12517-018-3733-x.

[22] Rafael C. Gonzalez and Richard E. Woods, *Digital Image Processing*, 3rd ed. (Prentice Hall, 2008), https://books.google.co.in/books?id=8uGOnjRGEzoC.

[23] Kit L. Yam and Spyridon E. Papadakis, "A Simple Digital Imaging Method for Measuring and Analyzing Color of Food Surfaces," *Journal of Food Engineering* 61, no. 1 (January 2004): 137–42, https://doi.org/10.1016/S0260-8774(03)00195-X.

features used for model development were colo[24], textures[25], morphology[26], hyperspectral-based feature[27].

In the present study, both color and texture-based image features were extracted and subsequently applied the PCA method of feature reduction before model development. The color-based images were derived in terms of R, G, and B from RGB color space, Grey from Grey color image, hue (H), saturation (S), and intensity (I) from HSI color space, cyan (C), magenta (M), yellow (Y), and black (K) from CMYK color space, lightness (L), red/green value (a), and blue/yellow value (b) from Lab color space, tristimulus (x), luminance (y) and z from xyz color images, whereas the texture information was mined in four different frequency transform coefficients viz. Fourier, Cosine, Wavelet, and Gabor filter using the 'I' color component.

---

[24] Snehamoy Chatterjee and Ashis Bhattacherjee, "Genetic Algorithms for Feature Selection of Image Analysis-Based Quality Monitoring Model: An Application to an Iron Mine," *Engineering Applications of Artificial Intelligence* 24, no. 5 (2011): 786–95, https://doi.org/10.1016/j.engappai.2010.11.009; Perez et al., "Ore Grade Estimation by Feature Selection and Voting Using Boundary Detection in Digital Image Analysis"; Tom Horrocks et al., "Classification of Gold-Bearing Particles Using Visual Cues and Cost-Sensitive Machine Learning," *Mathematical Geosciences* 47, no. 5 (2015): 521–45, https://doi.org/10.1007/s11004-015-9597-7; He Zhang and Xiuhua Jiang, "A Method Using Texture and Color Feature for Content-Based Image Retrieval," in *2015 IEEE International Conference on Computer and Communications (ICCC)* (Chengdu, China: IEEE, 2015), 122–27, https://doi.org/10.1109/CompComm.2015.7387552.

[25] Fionn Murtagh and Jean Luc Starck, "Wavelet and Curvelet Moments for Image Classification: Application to Aggregate Mixture Grading," *Pattern Recognition Letters* 29, no. 10 (2008): 1557–64, https://doi.org/10.1016/j.patrec.2008.03.008; Naresh Singh et al., "Textural Identification of Basaltic Rock Mass Using Image Processing and Neural Network," *Computational Geosciences* 14, no. 2 (2010): 301–10, https://doi.org/10.1007/s10596-009-9154-x; Chatterjee and Bhattacherjee, "Genetic Algorithms for Feature Selection of Image Analysis-Based Quality Monitoring Model: An Application to an Iron Mine"; Perez et al., "Ore Grade Estimation by Feature Selection and Voting Using Boundary Detection in Digital Image Analysis"; Zhang and Jiang, "A Method Using Texture and Color Feature for Content-Based Image Retrieval."

[26] Chatterjee and Bhattacherjee, "Genetic Algorithms for Feature Selection of Image Analysis-Based Quality Monitoring Model: An Application to an Iron Mine."

[27] Horrocks et al., "Classification of Gold-Bearing Particles Using Visual Cues and Cost-Sensitive Machine Learning."

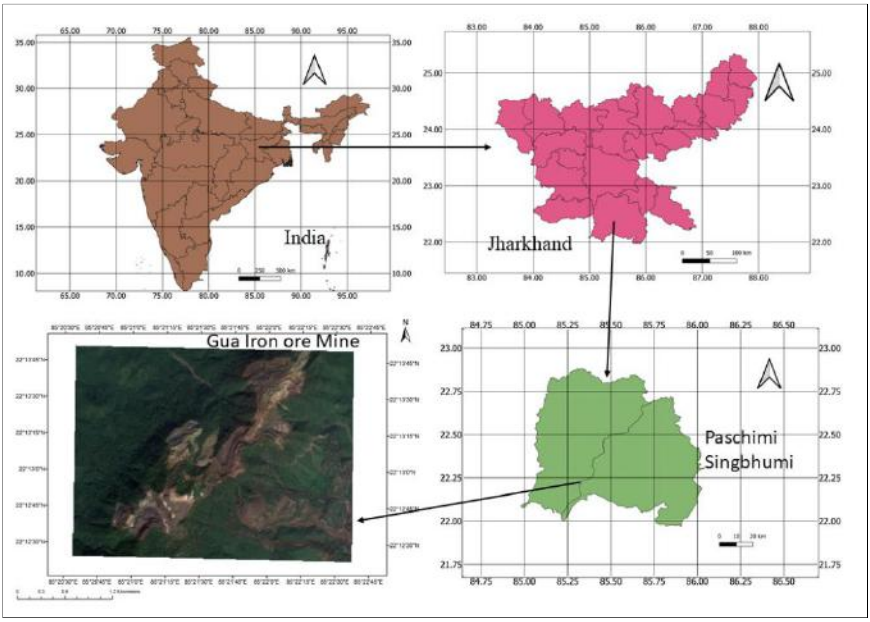


Figure 1. Map for location of sample collection mine.

After deriving the images into defined color components and frequency transform coefficients, image features were extracted in terms of ten statistical parameters viz. mean, minimum, maximum, standard deviation, variance, skewness, kurtosis, 3rd order moments, 4th order modemnts, and 5th order modemnts for PCA analysis. The total number of image features (color and texture) extracted was 280.

Feature Dimension Reduction

Analysis of the multivariate dataset became complex as the variable size increased. It will be simple if the small set of uncorrelated variables can be extracted from the large correlated multivariate dataset[28]. PCA is a

---

[28] Karl Pearson, "LIII. On Lines and Planes of Closest Fit to Systems of Points in Space," *Philosophical Magazine Series 6* 2, no. 11 (November 1901): 559–72, https://doi.org/10.1080/14786440109462720; H. Hotelling, "Analysis of a Complex of

type of statistical method, which converts the large number of correlated features into a smaller set of new uncorrelated features. These features contains retains almost all the original information available in the actual data set. The first component has the highest information by selecting the maximum variation[29].

It is observed that feature extraction produces a high dimension of features (=280) for classification model development. However, this high dimension of features makes the classification model a more complex structure. There may be a possibility that most of the features are interrelated and can be eliminated before model development. It is observed that classification model development with more number of correlated features may have a problem[30]. Thus, the objective to transform the correlated features into a lower dimension of linearly uncorrelated features. Thus, the study used PCA analysis to reduce the feature dimension. PCA determines the eigenvector having the highest eigenvalues of the covariance matrix of the feature set which explains the maximum variance[31].

The input feature vector F^n in a 280-dimensional space (F_1 F_2,…, F_{280}) transformed onto vectors X^P in an P-dimensional space (X_1, X_2,…, X_P) where $P < 280$. Each of the eigenvectors represents a principal component. PCA algorithm starts by determing the mean of the input vectors F^n and represented in 280 dimensions. It was observed that the mean could be the same for different distribution samples. Hence, the variance was calculated by subtracting each element of feature vector with its mean and multiplying these vectors with its transpose and finally averaging with 279 (=280-1).

Statistical Variables into Principal Components.," *Journal of Educational Psychology* 24, no. 6 (1933): 417–41, https://doi.org/10.1037/h0071325.

29 G H Dunteman, *Principal Components Analysis*, ed. Michael S. Lewis-Beck, Quantitative Applications in the Social Sciences (London, UK: Sage, 1991), https://books.google.co.in/books?id=PaDzvwEACAAJ.

30 Snehamoy Chatterjee et al., "Rock-Type Classification of an Iron Ore Deposit Using Digital Image Analysis Technique," *International Journal of Mining and Mineral Engineering* 1, no. 1 (2008): 22, https://doi.org/10.1504/IJMME.2008.020455.

31 Christopher M Bishop, *Neural Networks for Pattern Recognition* (New York, USA: Oxford university press, 1995).

Then, the covariance matrix was calculated between two feature sets. As the dimension of the feature is 280, therefore 280x280 covariance matrix was obtained. The covariance matrix is 280x280 square matrix, and therefore can be used for calculation of eigenvectors and eigenvalues. The eigenvalue has a dimension of 1x280, and the eigenvector has a dimension of 280x280.

The eigenvectors corresponding to the P highest eigenvalues were considered, and the input vectors F^n were projected onto the eigenvectors to obtain the transformed component vector X^P in P-dimensional space. These P-dimensional feature vectors were subsequently used as the input in the SVM classifier model.

Laboratory Analysis

Many techniques were developed for qualitative and quantitative analysis of ores. Any specific techniques that can be chosen for the analysis of ores depend on the characteristics of the ore samples being studied. These are broadly classified into three categories viz. chemical analysis, powder spectroscopy, and in-situ spectroscopy. The wet chemical analysis can be a gravimetric, volumetric, or colorimetric. The spectroscopy can be atomic absorption analysis (AAS), X-ray Fluorescence (XRF), Inductively coupled plasma (ICP), Electron probe micro-analysis (EPMA), X-ray Diffractometers (XRD), Scanning Electron Microscopy (SEM), Mass Spectrometry, and Raman spectroscopy. In the present study, XRD analyses were conducted for estimating the mineral compositions of iron ores.

XRD is an analytical method commonly used for the phase estimation of a crystalline sample[32]. For XRD analysis, samples should be finely grained, homogenized, and represent average bulk composition. In XRD analysis, the target samples are bobarded with accelerated electrons. As a

[32] Andrei A. Bunaciu, Elena gabriela Udriştioiu, and Hassan Y. Aboul-Enein, "X-Ray Diffraction: Instrumentation and Applications," *Critical Reviews in Analytical Chemistry* 45, no. 4 (October 2, 2015): 289–99, https://doi.org/10.1080/10408347.2014.949616.

result, the target material produced different X-ray spectra in different wavelengths depending on the characteristics of the samples. The specific wavelengths represent the characteristic of the samples.

The result obtained using XRD analysis was processed using Philips X'Pert HighScore to match with the library available for different components spectra[33]. Based on the matching of the peak, compositions of the sample can be estimated. The result revealed the presence of five different classes of iron ore in the collected sample named massive iron ore (C1), flaky friable blue dust ore (C2), lateritic iron ore (C3), banded hematite quartz (C4), and shale (C5).. The result also verified by the previous study done on the Gua iron ore mine[34]. This result is used to annotating the image sample into five different classes.

Model Development

The classification process is known as a supervised learning algorithm having input as observed input space and observed annotation space and output a generalized model with the optimum performance measure[35]. In early '80s decision tree and the artificial neural network has proved its effectiveness on the linear classification model development[36]. Later, a

---

[33] Iván Fernando Macías-Quiroga, Gloria Inés Giraldo-Gómez, and Nancy Rocío Sanabria-González, "Characterization of Colombian Clay and Its Potential Use as Adsorbent," *The Scientific World Journal* 2018 (October 24, 2018): 1–11, https://doi.org/10.1155/2018/5969178.

[34] Sumanbabu Patteti, Biswajit Samanta, and Debashis Chakravarty, "Design of a Feature-Tuned ANN Model Based on Bulk Rock-Derived Mineral Spectra for Endmember Classification of a Hyperspectral Image from an Iron Ore Deposit," *International Journal of Remote Sensing* 36, no. 8 (April 18, 2015): 2037–62, https://doi.org/10.1080/ 01431161.2015.1031920.

[35] Rohan Shiloh Shah, "Support Vector Machines for Classification and Regression" (McGill University, 2007).

[36] G. R. Dattatreya and V. V. S. Sarma, "Bayesian and Decision Tree Approaches for Pattern Recognition Including Feature Measurement Costs," *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-3, no. 3 (May 1981): 293–98, https://doi.org/10.1109/TPAMI.1981.4767102; Peter Argentiero, Roland Chin, and Paul Beaudet, "An Automated Approach to the Design of Decision Tree Classifiers," *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-4, no. 1 (January 1982): 51–57, https://doi.org/10.1109/TPAMI.1982.4767195; Y. X. Gu, Q. R. Wang, and C. Y. Suen,

nonlinear classification learning method is developed with the help of learning theory, such as SVM[37]. SVM model uses kernel function and quadratic optimization problem to reach empirical performance. It uses the structural risk minimization principle instead of an empirical risk minimization principle[38].

In the present work, SVM was used for iron ore classification into five different classes based on their component obtained from the XRD analysis. SVM method has proved its effectiveness in every industry and so with the mineral industry[39]. The SVM is developed as a binary classifier and used for categorizing the data into two classes +1 and -1[40]. However, multiclass classification can be achieved through one-versus-one (OVO) or one versus rest (OVR) methods. In the present work, auto-optimization has done by selecting randomly between OVO and OVR. When the OVO method has selected a set of 10 binary classifiers is working internally to find the final result by the highest voting method. If OVR is selected, then

---

"Application of a Multilayer Decision Tree in Computer Recognition of Chinese Characters," *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-5, no. 1 (January 1983): 83–89, https://doi.org/10.1109/TPAMI.1983.4767349; Botha, Barnard, and Casasent, "Optical Neural Networks for Image Analysis: Imaging Spectroscopy and Production Systems," in *IEEE International Conference on Neural Networks*, vol. 1 (San Diego, CA, USA: IEEE, 1988), 541–46, https://doi.org/10.1109/ICNN.1988.23889; Marks et al., "The Effect of Stochastic Interconnects in Artificial Neural Network Classification," in *IEEE International Conference on Neural Networks*, vol. 2 (San Diego, CA, USA: IEEE, 1988), 437–42, https://doi.org/10.1109/ICNN.1988.23957; Specht, "Probabilistic Neural Networks for Classification, Mapping, or Associative Memory," in *IEEE International Conference on Neural Networks*, vol. 1 (San Diego, CA, USA: IEEE, 1988), 525–32, https://doi.org/10.1109/ICNN.1988.23887.

[37] Corinna Cortes and Vladimir Vapnik, "Support-Vector Networks," *Machine Learning* 20, no. 3 (September 1995): 273–97, https://doi.org/10.1007/BF00994018.

[38] Steve Gunn, "Support Vector Machines for Classification and Regression" (Southampton, UK, 1998), http://svms.org/tutorials/Gunn1998.pdf.

[39] Jayson Tessier, Carl Duchesne, and Gianni Bartolacci, "A Machine Vision Approach to On-Line Estimation of Run-of-Mine Ore Composition on Conveyor Belts," *Minerals Engineering* 20, no. 12 (October 2007): 1129–44, https://doi.org/10.1016/j.mineng.2007.04.009; Perez et al., "Ore Grade Estimation by Feature Selection and Voting Using Boundary Detection in Digital Image Analysis"; Chatterjee, "Vision-Based Rock-Type Classification of Limestone Using Multi-Class Support Vector Machine"; Patel, Chatterjee, and Gorai, "Development of a Machine Vision System Using the Support Vector Machine Regression (SVR) Algorithm for the Online Prediction of Iron Ore Grades."

[40] Vapnik, "The Nature of Statistical Learning Theory."

five binary classifiers are working internally to obtain the end classification result by the highest voting method.

In the present study, the SVM classification algorithm is designed using the the iron ore image feature dataset as input space with dimension Dataset (D) × Feature (M) (= 812 × 31). The feature dimension (M) was reduced using PCA technique with the maximum retaining information, and annotation space as {1,2,3,4,5} for representing a number of classes T (=5). The datasets (D) were divided in the proportion of 70:30, respectively for training and testing of the model. The PCA components explaining the maximum variance of the data were used as input in the SVM model for classification of the ore samples.

Cortes and Vapnik (1995) have effectively illustrated the working principle and formulation of the SVM model[41]. The primary aim of the study was to obtain an optimal hyperplane that isolates the data into two classes. An optimal hyperplane can be defined as a linear function with the maximum margin between the lower and upper vectors represented for two classes. The optimal hyperplane and margin of the SVM model are shown below in Figure 2. The margin can be constructed with the help of selected data (known as support vector), as shown in a square box in the Figure 2.

The SVM model was designed as a binary classifier and classified all samples into two classes +1 and –1. First, it was considered that data used for training could be separated without error. If the training dataset contains tr samples and represented by $(x_1, y_1), (x_2, y_2), \ldots(x_{tr}, y_{tr})$, where $x \in \mathbb{R}^M$ represents the input feature with dimension d (=280) and $y \in \{-1,+1\}$ represent the output with two classes. Then, the SVM hyperplane is given by

$$w \cdot x_i + b = 0, \qquad for \quad i \in \{1, 2, \ldots, tr\}$$

and, the margin is given by

[41] Cortes and Vapnik, "Support-Vector Networks."

$$\begin{cases} w \cdot x_i + b \geq +1, & for \quad i \in \{1, 2, ..., tr\}, \quad y_i = +1 \\ w \cdot x_i + b \leq -1, & for \quad i \in \{1, 2, ..., tr\}, \quad y_i = -1 \end{cases}$$

Therefore, the inequalities can be derived as

$$y_i(w \cdot x_i + b) \geq 0, \qquad for \quad i \in \{1, 2, ..., tr\}, \quad y_i \in \{-1, +1\}$$

In the equalities, sample point, x_i lies on the margin. The hyperplane and margin hyperplanes are normal to the origin, and therefore the distance between hyperplanes and origin is given by |b|/||w||. The margin of the hyperplane and origin is given by |+1-b|/||w|| and |-1-b|/||w|| for class +1 and class -1 respectively. The hyperplane and margin hyperplanes are parallel, and therefore the distance between margin is given by 2/||w||. The goal of SVM is to maximize the margin 2/||w||, which is equivalent to minimize $||w||^2/2$

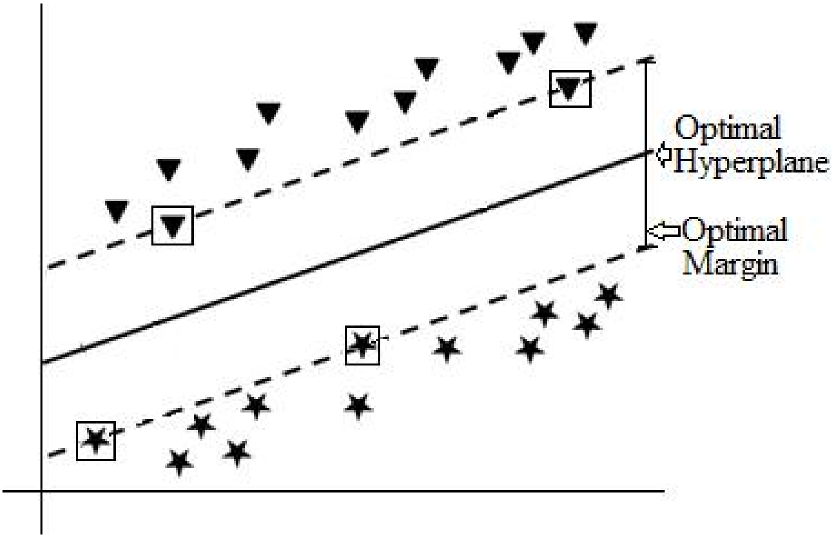


Figure 2. Representation of SVM classification for errorless classified data.

Therefore, the SVM model can be represented by

$$\text{minimize} \quad \frac{1}{2}\|w\|^2$$

$$\text{subject to}: \begin{cases} w \cdot x_i + b \geq 1, & \forall y_i = +1 \\ w \cdot x_i + b \leq 1, & \forall y_i = -1 \end{cases}$$

The SVM function is a quadratic optimization problem and can be solved using the Lagrange multipliers. The optimization function using Lagrange multipliers is given by

$$L(w,b,\alpha) = \frac{1}{2}w^2 - \sum_{i=1}^{tr} \alpha_i \left[y_i \left(w \cdot x_i + b \right) - 1 \right]$$

At the saddle point, the Lagrange multiplier is minimum (Kuhn-Tucker conditions) with respect to a primal variable, w and can be obtained by the partial differentiation with respect to it. The partial differentiation with respect to w gives the following equation

$$\frac{\partial L(w,b,\alpha)}{\partial w} = 0 \quad \Rightarrow \quad w = \sum_{i=1}^{tr} \alpha_i y_i x_i$$

Similarly, at the saddle point, the Lagrange multiplier is minimum (Kuhn-Tucker conditions) with respect to a primal variable, b (equality constraint). The partial differentiation of the objective function with respect to b gives the following equation.

$$\frac{\partial L(w,b,\alpha)}{\partial b} = 0 \quad \Rightarrow \quad \sum_{i=1}^{tr} \alpha_i y_i = 0$$

At the saddle point, the Lagrange multiplier is maximized with respect to the dual variable, α (positivity constraint) and is given by

$$\alpha_i \geq 0, \qquad i = 1, 2, ..., tr$$

Putting the value of w at equation (4.7) in equation (4.6), we get

$$L(\alpha) = \sum_{i=1}^{tr} \alpha_{\text{i}} - \frac{1}{2}\sum_{i,j=1}^{tr} \alpha_i \alpha_j y_i y_j x_i \cdot x_j,$$

The function is maximized with respect to the α subject to equality constraint and positivity constraint given in the last two equations. The optimal solution $\alpha^0 = (\alpha_1^0, \alpha_2^0, \ldots, \alpha_{tr}^0)$ specifies the coefficients for the optimal hyperplane and given by

$$w_0 = \sum_{i=1}^{tr} \alpha_i^0 y_i x_i,$$

Therefore, the optimal hyperplane is given by

$$\sum_{i=1}^{tr} \alpha_i^0 y_i x_i \cdot x + b_0 = 0,$$

The optimal solution satisfied Karush-Kuhn-Tucker condition is given by

$$\alpha_i^0 [y_i (w_0 \cdot x + b_0) - 1] = 0,$$

The above condition tells that if the expansion of vector w_0 uses vector x_i with non-zero weight α_i^0, then the following equality can be maintained (by the support vector xi)

$$y_i (w_0 \cdot x_i + b_0) = 1,$$

Now it is considered that the data cannot be separated without error. Then it was observed that few points are non-separable, and therefore, allowed to deviate the margin rules. A slack variable ξi has been introduced to measure this deviation and shown below in Figure 3.

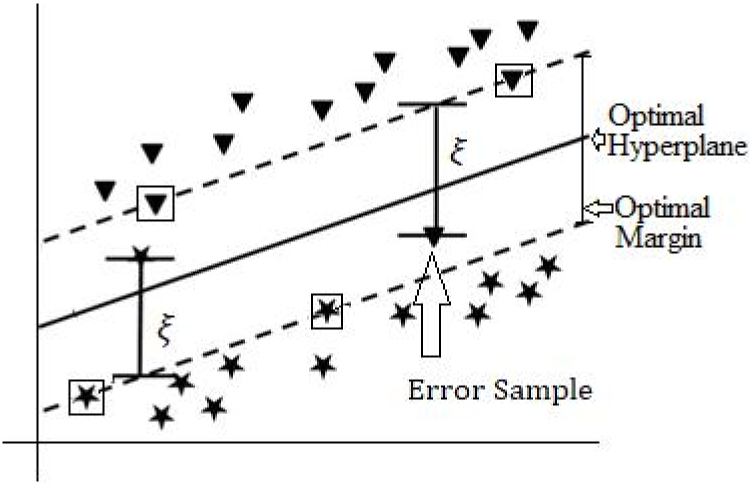


Figure 3. Representation of SVM classification for erroneous classified data.

So, the revised SVM model can be represented by

$$\textit{minimize} \ \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{tr}\xi_i$$

$$\textit{subject to}: \begin{cases} y_i(w \cdot x_i + b) \geq 1 - \xi_i, & \forall x_i \\ \xi_i \geq 0, & i = 1, 2, ..., tr \end{cases}$$

In the above model, C represents the cost parameters and can be optimised by trade-off analysis between model complexity and performance error. The nature of the above problem is quadratic optimization problem and thus Lagrange multipliers method can be used to solve the problem. Applying the Lagrange multipliers, a new objective function is given by

$$L(w,b,\alpha) = \frac{1}{2}w^2 - \sum_{i=1}^{tr} \alpha_i \left[y_i (w \cdot x_i + b) - 1 + \xi_i \right] + C \sum_{i=1}^{tr} \xi_i - \sum_{i=1}^{tr} v_i \xi_i$$

The function needs to maximize with respect to the Lagrange multiplier αi ≥ 0 and vi ≥ 0 and minimize with respect to w, b, and ξ. The minimization with respect to w and b produces the same result given by equation for saddle point previously. The minimization with respect to ξ gives

$$\alpha_i + v_i = C$$

It is known that $v_i \geq 0$, so the equation can be written in the form

$$0 \leq \alpha_i \leq C$$

By putting this equation obtained for the saddle point into the Lagrange equation, the equation becomes

$$L(\alpha) = \sum_{i=1}^{tr} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{tr} \alpha_i \alpha_j y_i y_j x_i \cdot x_j ,$$

with the constraint

$$\begin{cases} 0 \leq \alpha_i \leq C \\ \sum_{i=1}^{tr} \alpha_i y_i = 0 \end{cases}$$

It may be possible that some data are not linearly separable, and therefore the feature space of input vector, x, was mapped to higher dimensional feature space ϕ(x) in order to make them linearly separable using the optimal hyperplane. In the SVM model, the kernel function was introduced for mapping into higher dimensional space as shown in Figure 4.

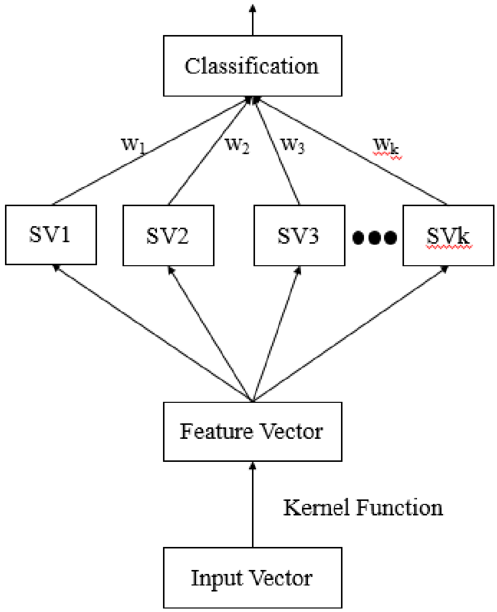


Figure 4. Proceeding of input data into SVM classification.

The kernel function is given by

$$k(x \cdot x_i) = \phi(x) \cdot \phi(x_i)$$

In the previous studies, many types of kernal functions like polynomial function, radial basis function (RBF), Gaussian kernel function, etc. were used in the SVM model development[42]. Among all the kernel functions, RBF is widely used in a different types of pattern classifications. The current study also used an RBF as a kernel function. It is given by:

$$K(x, x_i) = \exp\left(\frac{\|x - x_i\|^2}{\sigma^2}\right)$$

[42] Vapnik, "The Nature of Statistical Learning Theory."

where σ represents the kernel function's bandwidth, and x and x_i represents the two feature vectors. The kernel feature vector with respect to x is determined from these two feature vectors. The function for classification of data is given by:

$$y_i = \begin{cases} +1, & if \quad w \cdot k(x, x_i) + b \geq 1 - \xi_i \\ -1, & if \quad w \cdot k(x, x_i) + b < -1 + \xi_i \end{cases}$$

The number of image samples (= 812) was partitioned into training and testing in the ratio of 7:3 (=569:243). The SVM model was trained in the Matlab platform using auto-optimization active. The default iteration 30 is used for stopping the criteria of training the model along with objective function. The model has chosen the one-vs-one method for enabling multiclass classification by SVM. The optimum result is found on box constrains at 388.2 and kernel scale at 0.0011. The progress of the objective function toward the goal during the training process can be seen in Figure 5.

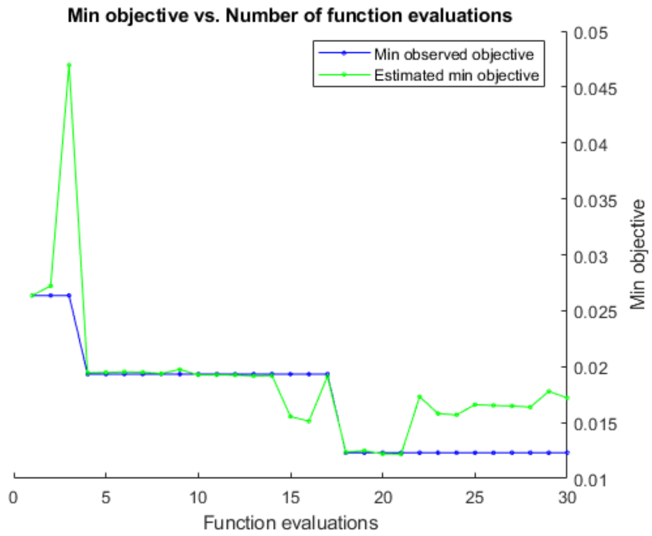


Figure 5. Progress of objective function toward the goal during the training process.

Performance Evaluation

The performance of the proposed PCA-based SVM model was examined based on the testing data set (=243) using four indicators (sensitivity, specificity, misclassification, and accuracy). These were estimated using confusion matrix parameters. The model performnace indicators were determined by combining the results of the individual class. The samples were classified into the actual class or true class (TC) with the integration of true positive (TP) and true negative (TN). On the other hand, samples classified into other than the actual classes are called false classification (FC) with the integration of false-negative (FN) and false positive (FP). The testing sample (=243) were used for performance analysis of the developed model. The results of TP, TN, FP and FN for different classes are shown in Table 1.

Table 1. Confusion matrix for the SVM model using PC data

		Predicted				
		C1	**C2**	**C3**	**C4**	**C5**
Actual	**C1**	69	1	0	0	0
	C2	0	62	0	0	0
	C3	0	0	40	0	0
	C4	0	0	0	18	0
	C5	0	0	0	2	51

The model performance indicators were determined from TP, TN, FP, and FN using the following relationships. The sensitivity of any individual class, T, is the ratio of the number of samples classified as TP and the total number of samples classified as TP and FN. Mathematically, it can be written as:

$$Sensitivity_T = \frac{TP}{TP + FN}$$

The specificity of any individual class, T, is the ratio of the number of samples classified as TN and the total number of samples classified as FP and TN. Mathematically, it can be represented as:

$$Specificity_T = \frac{TN}{FP + TN}$$

The misclassification of any individual class, T, is the ratio of the number of samples classified as FC and the total number of samples classified as TC and FC. Mathematically, it can be represented as:

$$Missclassification_T = \frac{FC}{TC + FC} = \frac{FP + FN}{TP + TN + FP + FN}$$

Accuracy is the ratio of the number of samples classified as TC to the total number of samples classified as TC and FC. This can be represented for class T as:

$$Accuracy_T = \frac{TC}{TC + FC} = \frac{TP + TN}{TP + TN + FP + FN}$$

The average of each class corresponding to each performance indices (sensitivity, specificity, misclassification, and accuracy) represents the model value. The result obtained for the classification of the testing sample is summarized in Table 2.

Table 2. Confusion matrix indices for SVM performance using PC data

	Performance Indices			
	Sensitivity	**Specificity**	**Misclassification**	**Accuracy**
C1	0.9857	1.0000	0.0041	0.9959
C2	1.0000	0.9945	0.0041	0.9959
C3	1.0000	1.0000	0.0000	1.0000
C4	1.0000	0.9911	0.0082	0.9918
C5	0.9623	1.0000	0.0082	0.9918
Model	0.9896	0.9971	0.0049	0.9951

Table 3. Comparison of confusion matrix indices among SVM, kNN and DT performance using PC data

		Performance Indices			
		Sensitivity	**Specificity**	**Misclassification**	**Accuracy**
C1	SVM	0.9857	1.0000	0.0041	0.9959
	kNN	0.8286	0.9942	0.0535	0.9465
	DT	0.8714	0.9827	0.0494	0.9506
C2	SVM	1.0000	0.9945	0.0041	0.9959
	kNN	0.9677	0.9392	0.0535	0.9465
	DT	0.9194	0.9613	0.0494	0.9506
C3	SVM	1.0000	1.0000	0.0000	1.0000
	kNN	0.9750	0.9507	0.0453	0.9547
	DT	1.0000	0.9754	0.0206	0.9794
C4	SVM	1.0000	0.9911	0.0082	0.9918
	kNN	0.9444	0.9956	0.0082	0.9918
	DT	0.8333	0.9644	0.0453	0.9547
C5	SVM	0.9623	1.0000	0.0082	0.9918
	kNN	0.8491	0.9947	0.0370	0.9630
	DT	0.8679	0.9947	0.0329	0.9671
Model	SVM	0.9896	0.9971	0.0049	0.9951
	kNN	0.9130	0.9749	0.0395	0.9605
	DT	0.8984	0.9757	0.0395	0.9605

The performance of the proposed PCA-based SVM model was also compared with the performance of other classification algorithms like k-nearest neighbor (kNN) and decision tree (DT). The result indicates that the PCA-based SVM model performs relatively better than the kNN and DT classification algorithms. The results are summarized in Table 3.

The performance of the proposed PCA-based SVM model was also compared with the previous study results of Patel et al.[43]. Patel et al. conducted a study for the classification of the same iron ore samples using the SFFS-based SVM model. The comparative results (shown in Table 4) reveals that the PCA-based SVM model performs better than the SFFS-based SVM model. This may be due to the retaining of higher variance input data in the PCA-based SVM model.

[43] Patel, Chatterjee, and Gorai, "Development of an Expert System for Iron Ore Classification."

Table 4. Comparison of SVM-PCA model (present study) with the SVM-SFFS model (previous study by Patel et al.)

		Performance Indices			
		Sensitivity	**Specificity**	**Misclassification**	**Accuracy**
Model	PCA-based SVM	0.9896	0.9971	0.0049	0.9951
	SFFS-based SVM	0.9792	0.9949	0.0082	0.9918

CONCLUSION

In the present study, SVM is used for the iron ore classification into five classes based on the principal components obtained from the extracted image features. The iron ore sample is collected from Gua iron ore mine and run on the conveyor system fabricated in the laboratory to capture images. A set of 812 images have captured from different ore samples and preprocessed. These images have used for feature extraction and a set of 280 features extracted for each image resulting in a feature dataset of size 812×280. The data were partitioned into training and testing of the model in the ratio of 569 to 243. This huge dimension of feature is reduced by using principal component analysis and observed that the first 31 principal components explain 95% of the variance and thus used for SVM model development. The SVM model was auto optimized and found the best performance with parameter box constrains at 811.39 and kernel scale at 0.0019 using the one-vs-one method. The model performance is evaluated using the testing sample and obtained performance indices sensitivity, specificity and accuracy near to 1 and misclassification near to 0. The result revealed that the model could be used efficiently for the prediction of the class of an iron ore sample. The model performance was also compared with the other classifiers like kNN and DT. The comparative results indicated that PCA-based SVM performs relatively better than the others.

References

Argentiero, Peter, Roland Chin, Paul Beaudet. "An Automated Approach to the Design of Decision Tree Classifiers." *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-4, no. 1, (January 1982), 51–57. https://doi.org/10.1109/TPAMI.1982.4767195.

Barbosa, Rommel M; Eloisa Silva de Paula, Ana Carolina Paulelli, Anthony F. Moore, Juliana Maria Oliveira Souza, Bruno Lemos Batista, Andres D. Campiglia, Fernando Barbosa. "Recognition of Organic Rice Samples Based on Trace Elements and Support Vector Machines." *Journal of Food Composition and Analysis*, 45, (2016), 95–100. https://doi.org/10.1016/j.jfca.2015.09.010.

Belkin, Mikhail, Partha Niyogi. "Laplacian Eigenmaps for Dimensionality Reduction and Data Representation." *Neural Computation*, 15, no. 6, (June 2003), 1373–96. https://doi.org/10.1162/089976603321780317.

Bishop, Christopher M. *Neural Networks for Pattern Recognition*. New York, USA: Oxford university press, 1995.

Boser, Bernhard E; Isabelle M. Guyon, Vladimir N. Vapnik. "A Training Algorithm for Optimal Margin Classifiers." In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory - COLT '92*, 144–52. New York, New York, USA: ACM Press, 1992. https://doi.org/10.1145/130385.130401.

Botha, Barnard, Casasent. "Optical Neural Networks for Image Analysis: Imaging Spectroscopy and Production Systems." In *IEEE International Conference on Neural Networks*, 1, 541–46. San Diego, CA, USA: IEEE, 1988. https://doi.org/10.1109/ICNN.1988.23889.

Bunaciu, Andrei A; Elena gabriela Udriştioiu, Hassan Y. Aboul-Enein. "X-Ray Diffraction: Instrumentation and Applications." *Critical Reviews in Analytical Chemistry*, 45, no. 4, (October 2, 2015), 289–99. https://doi.org/10.1080/10408347.2014.949616.

Chatterjee, Snehamoy. "Vision-Based Rock-Type Classification of Limestone Using Multi-Class Support Vector Machine." *Applied*

Intelligence, 39, no. 1, (July 12, 2013), 14–27. https://doi.org/ 10.1007/s10489-012-0391-7.

Chatterjee, Snehamoy, Ashis Bhattacherjee. "Genetic Algorithms for Feature Selection of Image Analysis-Based Quality Monitoring Model: An Application to an Iron Mine." *Engineering Applications of Artificial Intelligence*, 24, no. 5, (2011), 786–95. https://doi.org/ 10.1016/j.engappai.2010.11.009.

Chatterjee, Snehamoy, Ashis Bhattacherjee, Biswajit Samanta, Samir Kumar Pal. "Rock-Type Classification of an Iron Ore Deposit Using Digital Image Analysis Technique." *International Journal of Mining and Mineral Engineering*, 1, no. 1, (2008), 22. https://doi.org/ 10.1504/IJMME.2008.020455.

Cortes, Corinna, Vladimir Vapnik. "Support-Vector Networks." *Machine Learning*, 20, no. 3, (September 1995), 273–97. https://doi.org/ 10.1007/BF00994018.

Dattatreya, GR; Sarma, VVS. "Bayesian and Decision Tree Approaches for Pattern Recognition Including Feature Measurement Costs." *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-3, no. 3, (May 1981), 293–98. https://doi.org/10.1109/ TPAMI.1981. 4767102.

Dunteman, GH. *Principal Components Analysis*. Edited by Michael S. Lewis-Beck. Quantitative Applications in the Social Sciences. London, UK: Sage, 1991. https://books.google.co.in/ books?id= PaDzvwEACAAJ.

Galdames, Francisco J; Claudio A. Perez, Pablo A. Estévez, Martin Adams. "Classification of Rock Lithology by Laser Range 3D and Color Images." *International Journal of Mineral Processing*, 160, (March 2017), 47–57. https://doi.org/10.1016/j.minpro.2017.01.008.

Galdames, Francisco J; Claudio A Perez, Pablo A Estévez, Martin Adams, Martin Adams. "Rock Lithological Classification by Hyperspectral, Range 3D and Color Images." *Chemometrics and Intelligent Laboratory Systems*, 2019. https://doi.org/10.1016/ j.chemolab. 2019.04.006.

Gonzalez, Rafael C; Richard E. Woods. *Digital Image Processing*. 3rd ed. Prentice Hall, 2008. https://books.google.co.in/books?id=8uGOnjRGEzoC.

Gu, YX; Wang, QR; Suen, CY. “Application of a Multilayer Decision Tree in Computer Recognition of Chinese Characters.” *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-5, no. 1, (January 1983), 83–89. https://doi.org/10.1109/TPAMI.1983.4767349.

Gunn, Steve. *Support Vector Machines for Classification and Regression*. Southampton, UK, 1998. http://svms.org/tutorials/ Gunn1998.pdf.

Horrocks, Tom, Daniel Wedge, Eun Jung Holden, Peter Kovesi, Nick Clarke, John Vann. “Classification of Gold-Bearing Particles Using Visual Cues and Cost-Sensitive Machine Learning.” *Mathematical Geosciences*, 47, no. 5, (2015), 521–45. https://doi.org/10.1007/s11004-015-9597-7.

Hotelling, H. “Analysis of a Complex of Statistical Variables into Principal Components.” *Journal of Educational Psychology*, 24, no. 6, (1933), 417–41. https://doi.org/10.1037/h0071325.

Hsu, Chih Wei, Chih Jen Lin. “A Comparison of Methods for Multiclass Support Vector Machines.” *IEEE Transactions on Neural Networks*, 13, no. 2, (March 2002), 415–25. https://doi.org/10.1109/72.991427.

Jang, JSR. “ANFIS: Adaptive-Network-Based Fuzzy Inference System.” *IEEE Transactions on Systems, Man, and Cybernetics*, 23, no. 3, (1993), 665–85. https://doi.org/10.1109/21.256541.

Jolliffe, IT. *Principal Component Analysis*. Springer Series in Statistics. New York: Springer-Verlag, 2002. https://doi.org/10.1007/b98835.

Kohavi, Ron, George H. John. “Wrappers for Feature Subset Selection.” *Artificial Intelligence*, 97, no. 1–2, (1997), 273–324. https://doi.org/10.1016/S0004-3702(97)00043-X.

Kudo, Mineichi, Jack Sklansky. “Comparison of Algorithms That Select Features for Pattern Classifiers.” *Pattern Recognition*, 33, no. 1, (January 2000), 25–41. https://doi.org/10.1016/S0031-3203(99)00041-2.

Macías-Quiroga, Iván Fernando, Gloria Inés Giraldo-Gómez, Nancy Rocío Sanabria-González. “Characterization of Colombian Clay and Its

Potential Use as Adsorbent." *The Scientific World Journal*, 2018, (October 24, 2018), 1–11. https://doi.org/10.1155/2018/5969178.

Marks, Atlas, Park, Oh Seho. "The Effect of Stochastic Interconnects in Artificial Neural Network Classification." In *IEEE International Conference on Neural Networks*, 2, 437–42. San Diego, CA, USA: IEEE, 1988. https://doi.org/10.1109/ICNN.1988.23957.

Mountrakis, Giorgos, Jungho Im, Caesar Ogole. "Support Vector Machines in Remote Sensing: A Review." *ISPRS Journal of Photogrammetry and Remote Sensing*, 66, no. 3, (May 2011), 247–59. https://doi.org/10.1016/j.isprsjprs.2010.11.001.

Murtagh, Fionn, Jean Luc Starck. "Wavelet and Curvelet Moments for Image Classification: Application to Aggregate Mixture Grading." *Pattern Recognition Letters*, 29, no. 10, (2008), 1557–64. https://doi.org/10.1016/j.patrec.2008.03.008.

Patel, Ashok Kumar, Snehamoy Chatterjee, Amit Kumar Gorai. "Development of a Machine Vision System Using the Support Vector Machine Regression (SVR) Algorithm for the Online Prediction of Iron Ore Grades." *Earth Science Informatics*, November 9, 2018. https://doi.org/10.1007/s12145-018-0370-6.

———. "Development of an Expert System for Iron Ore Classification." *Arabian Journal of Geosciences*, 11, no. 15, (August 27, 2018), 401. https://doi.org/10.1007/s12517-018-3733-x.

———. "Development of Machine Vision-Based Ore Classification Model Using Support Vector Machine (SVM) Algorithm." *Arabian Journal of Geosciences*, 10, no. 5, (March 1, 2017), 107. https://doi.org/10.1007/s12517-017-2909-0.

———. "Development of Online Machine Vision System Using Support Vector Regression (SVR) Algorithm for Grade Prediction of Iron Ores." In *2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA)*, 149–52. IEEE, 2017. https://doi.org/10.23919/MVA.2017.7986823.

Patteti, Sumanbabu, Biswajit Samanta, Debashis Chakravarty. "Design of a Feature-Tuned ANN Model Based on Bulk Rock-Derived Mineral Spectra for Endmember Classification of a Hyperspectral Image from an Iron Ore Deposit." *International Journal of Remote Sensing*, 36, no. 8, (April 18, 2015), 2037–62. https://doi.org/10.1080/01431161.2015.1031920.

Pearl, Judea. "Reverend Bayes on Inference Engines: A Distributed Hierarchical Approach." In *Proceedings of the Second AAAI Conference on Artificial Intelligence*, 133–36. AAAI'82. AAAI Press, 1982. http://dl.acm.org/citation.cfm?id=2876686.2876719.

Pearson, Karl. "LIII. On Lines and Planes of Closest Fit to Systems of Points in Space." *Philosophical Magazine Series 6*, 2, no. 11, (November 1901), 559–72. https://doi.org/10.1080/14786440109462720.

Perez, Claudio A; Pablo A. Estévez, Pablo A. Vera, Luis E. Castillo, Carlos M. Aravena, Daniel A. Schulz, Leonel E. Medina. "Ore Grade Estimation by Feature Selection and Voting Using Boundary Detection in Digital Image Analysis." *International Journal of Mineral Processing*, 101, no. 1–4, (November 2011), 28–36. https://doi.org/10.1016/j.minpro.2011.07.008.

Quinlan, J. Ross. "Learning Efficient Classification Procedures and Their Application to Chess End Games." In *Machine Learning*, 463–82. Elsevier, 1983. https://doi.org/10.1016/B978-0-08-051054-5.50019-4.

Quinlan, JR. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Series in Machine Learning. Elsevier Science, 1993. https://books.google.co.in/books?id=HExncpjbYroC.

Raghavendra. N; Sujay, Paresh Chandra Deka. "Support Vector Machine Applications in the Field of Hydrology: A Review." *Applied Soft Computing*, 19, (2014), 372–86. https://doi.org/10.1016/j.asoc.2014.02.002.

Rosenblatt, F. "The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain 1." *Psychological Review*., Vol. 65, 1958. https://www.ling.upenn.edu/courses/cogs501/Rosenblatt1958.pdf.

Rosenblatt, F. *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*, Washington, DC: Spartan 1962.

Sarma, DD. *Geostatistics with Applications in Earth Sciences*. Dordrecht: Springer Netherlands, 2009. https://doi.org/10.1007/978-1-4020-9380-7.

Sebe, Nicu, Ira Cohen, Ashutosh Garg, Thomas S. Huang. *Machine Learning in Computer Vision*., Vol. *29*. Computational Imaging and Vision. Berlin/Heidelberg: Springer-Verlag, 2005. https://doi.org/10.1007/1-4020-3275-7.

Shah, Rohan Shiloh. *Support Vector Machines for Classification and Regression*. McGill University, 2007.

Singh, Naresh, Singh, T; Avyaktanand Tiwary, Kripa Sarkar. "Textural Identification of Basaltic Rock Mass Using Image Processing and Neural Network." *Computational Geosciences*, 14, no. 2, (2010), 301–10. https://doi.org/10.1007/s10596-009-9154-x.

Specht. "Probabilistic Neural Networks for Classification, Mapping, or Associative Memory." In *IEEE International Conference on Neural Networks*, 1, 525–32. San Diego, CA, USA: IEEE, 1988. https://doi.org/10.1109/ICNN.1988.23887.

Tessier, Jayson, Carl Duchesne, Gianni Bartolacci. "A Machine Vision Approach to On-Line Estimation of Run-of-Mine Ore Composition on Conveyor Belts." *Minerals Engineering*, 20, no. 12, (October 2007), 1129–44. https://doi.org/10.1016/j.mineng.2007.04.009.

Vapnik, Vladimir. "Pattern Recognition Using Generalized Portrait Method." *Automation and Remote Control*, 24, (1963), 774–80.

Vapnik, Vladimir N. *The Nature of Statistical Learning Theory*, 1995.

Vapnik, Vladimir Naumovich, Vlamimir Vapnik. *Statistical Learning Theory*., Vol. 1. Wiley New York, 1998.

Werbos, PJ. *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*. Washington, D.C.: Harvard University, 1975. https://books.google.co.in/books?id=z81XmgEACAAJ.

Widrow, B; Hoff, ME. "Adaptive Switching Circuits." In *1960 IRE WESCON Convention Record*, *Part 4*, 96–104. New York: IRE, 1960. https://books.google.co.in/books?id=Vs4EAAAAIAAJ.

Wu, Yu-Chieh, Yue-Shi Lee, Jie-Chi Yang. "Robust and Efficient Multiclass SVM Models for Phrase Pattern Recognition." *Pattern*

Recognition 41, no. 9, (September 2008), 2874–89. https://doi.org/10.1016/j.patcog.2008.02.010.

Yam, Kit L; Spyridon E. Papadakis. "A Simple Digital Imaging Method for Measuring and Analyzing Color of Food Surfaces." *Journal of Food Engineering*, 61, no. 1, (January 2004), 137–42. https://doi.org/10.1016/S0260-8774(03)00195-X.

Zhang, He, Xiuhua Jiang. "A Method Using Texture and Color Feature for Content-Based Image Retrieval." In *2015 IEEE International Conference on Computer and Communications (ICCC)*, 122–27. Chengdu, China: IEEE, 2015. https://doi.org/10.1109/CompComm.2015.7387552.

BIOGRAPHICAL SKETCHES

Ashok Kumar Patel

Affiliation: Department of Computer Science and Engineering, C.V. Raman Global University, Bhubaneswar, Orissa, India

Education: PhD

Business Address Department of Computer Science and Engineering, C.V. Raman Global University, Bidyanagar, Mahura, Janla, Bhubaneswar, Orissa, 752054, India.

Research and Professional Experience: 7 Years of Teaching and Research Experiences

Professional Appointments: Assistant Professor

Publications from the Last 3 Years: 5 research papers published in International Journals, and 1 peer-reviewed conference papers.

Snehamoy Chatterjee

Affiliation: Department of Geological and Mining Engineering and Sciences, Michigan Technological University, Michigan, US.

Education: PhD

Business Address: Department of Geological and Mining Engineering and Sciences, Michigan Technological University, 1400 Townsend Drive, Houghton 49931, Michigan, US.

Research and Professional Experience: 12 Years of Teaching and Research Experiences

Professional Appointments: Assistant Professor

Honors: Editor's Best Reviewers Award 2014, Mathematical Geosciences Journal; Young OR (Operations Research) Professional Award by 37th International Symposium on Application of Computers and Operations Research in the Mineral Industry (APCOM 2015)

Publications from the Last 3 Years: 19 research papers published in International Journals, and 6 peer-reviewed conference papers.

Amit Kumar Gorai

Affiliation: Department of Mining Engineering, NIT Rourkela, Odisha, India

Education: PhD

Business Address: Department of Mining Engineering, NIT Rourkela, Odisha - 769008, India.

Research and Professional Experience: 12 Years of Teaching and Research Experiences

Professional Appointments: Associate Professor

Honors: Raman Postdoctoral Fellowship awarded by UGC, New Delhi, Endeavour Executive Fellowship awarded by Australian Department of Higher Education.

Publications from the Last 3 Years: 18 research papers published in International Journals

In: Support-Vector Machines
Editor: Pooja Saigal
ISBN: 978-1-53618-757-1

Chapter 7

MULTI-CATEGORY CLASSIFICATION

Pooja Saigal, PhD*
Vivekananda School of Information Technology,
Vivekananda Institute of Professional Studies, New Delhi, India

Abstract

The conventional Support vector machines (SVMs) are designed for binary classification. Researchers from Machine learning community have been actively trying to extend the binary classifiers for multi-category classification. Various approaches have been proposed which typically involves construction of a multi-category classifier by combining several binary classifiers. Few methods are proposed that consider all classes at once. Such problems are computationally very expensive, so the comparisons of these methods using large-scale problems have not been conducted seriously. Solving a multi-category SVM in one step results in a very large optimization problem, which is restricted to be used with small data sets only. In this chapter, the implementations for "single optimization problem" and methods based on a combination of binary classifications like one-against-all, one-against-one and directed acyclic graph SVM are discussed.

Keywords: binary classifiers, multi-category classification, one-against-all, one-against-one, directed acyclic graph, ternary decision structure

*Corresponding Author's Email: pooja.saigal@vips.edu, saigal.pooja.in@gmail.com.

1. INTRODUCTION

The last two decades have witnessed the evolution of Support Vector Machines (SVMs) as a powerful paradigm for pattern classification and regression [1],[2]. SVMs emerged from research in statistical learning theory on how to regulate the trade-off between structural complexity and empirical risk. One of the most popular SVM classifiers is the "maximum margin" one that attempts to reduce generalization error by maximizing the margin between two disjoint half planes [1],[2],[3]. The resulting optimization task involves the minimization of a convex quadratic function subject to linear inequality constraints.

Support vector machines (SVMs) [4, 5] were originally proposed for binary classification. Real life problems deal with more than two classes. Therefore, researchers have been trying to extend the binary classifiers to multi-category framework. To do so, there are two types of approaches. One way is to construct and combine several binary classifiers while the other is by directly considering all data in one large optimization problem. The formulation of a single optimization problem, to handle multi-category classification data, has variables proportional to the number of classes. Hence, it is computationally more expensive to solve a multi-category problem than a combination of several binary problems, with the same number of data patterns. Up to now experiments are limited to small data sets. In this chapter, the implementations for "single optimization problem" and methods based on a combination of binary classifications like one-against-all (OAA), one-against-one (OAO) and directed acyclic graph (DAG) SVM are discussed [6].

In this chapter, various multi-category classification algorithms are discussed. They are One-against-one, One-against-all, Directed Acyclic Graph, Multi-SVM optimization problem with all data, Ternary decision structure and few other popular approaches.

Symbols and Notations

For a binary classification problem, let the training set has m data patterns, given as $\{(x_i, y_i), \quad (i = 1, 2, ..., m)\}$. Here, $x_i \quad (i = 1, 2, ..., m)$ is a row vector in n-dimensional real space $\mathbb{R}^n$. Let matrix X of dimensions $m \times n$, represents all the training patterns. The labels $y_i \in \{+1, -1\}$ for positive and negative classes are given by $+1$ and -1 respectively. When working with a pair of optimization problems, the patterns belonging to positive and negative classes will be represented by matrices A and B respectively. The number of patterns in these classes be given by m_1 and m_2 ($m = m_1 + m_2$); therefore, the order of matrices A and B are $(m_1 \times n)$ and $(m_2 \times n)$ respectively. Here, n is the dimension of feature space and A_i $(i = 1, 2, ..., m_1)$ is a row vector in n-dimensional real space $\mathbb{R}^n$, that represents feature vector of a data sample. In this chapter, 'positive class' and 'Class $+1$' are used interchangeably; similarly 'negative class' and 'Class -1' would refer to set of negative patterns.

2. Multi-Category Algorithms for SVM

In this section, few popular approaches for multi-category extension of SVM are discussed.

2.1. One-Against-All

A very popular implementation for SVM multi-category classification is probably the one-against-all method [7]. One-against-all constructs K SVM classifiers where K is the number of classes.

The i^{th} SVM classifier, where $i = 1, ..., K$, is trained with patterns of the i^{th} class with positive labels, and all other patterns with negative labels. Thus, given m training patterns $(x_1, y_1), ..., (x_m, y_m)$, where $x_j \in \mathbb{R}^n,\ j = 1, ..., m$ and $y_j \in \{1, ..., K\}$ is the class of x_j. The i^{th} SVM solves an optimization problem to obtain i^{th} hyperplane given as

$$(w^i)^T x + (b^i) = 0, \ \ i = 1, ..., K. \tag{1}$$

The optimization problem is given as

$$\begin{aligned} \min_{w^i, b^i, \xi^i} \quad & \frac{1}{2}\|w^i\| + C\sum_{j=1}^{m} \xi_j^i (w^i)^T \\ \text{subject to} \quad & (w^i)^T \phi(x_j) + b^i \geq 1 - \xi_j^i, \ \ if\ y_j = i \\ & (w^i)^T \phi(x_j) + b^i \leq -1 + \xi_j^i, \ \ if\ y_j \neq i \\ & \xi_j^i \geq 0, \ \ j = 1, ..., m. \end{aligned} \tag{2}$$

Here, the training patterns x_j are mapped to a higher dimensional space by the function ϕ and C is the penalty parameter. Minimizing $\frac{1}{2}\|w^i\|$ would maximize the margin between two groups of data. As a soft-margin implementation, there is a penalty term $C\sum_{j=1}^{m}\xi_j^i$ which can reduce the amount of training error. This optimization problem tries to maintain a trade-off between the regularization term $\frac{1}{2}\|w^i\|$ and the training errors. Here, $\|.\|$ represents L_2-norm.

After solving (2), K decision hyperplanes (1) are obtained. Any new test pattern x is in the class which has the largest value of the decision function

$$\text{class of x} \quad = \quad arg \max_{i=1,\ldots,K}((w^i)^T\phi(x)+b^i). \tag{3}$$

Practically, we solve the dual problem of (2) where the number of constraints is the same as the number of patterns. OAA-SVM implements a series of binary classifiers where each classifier separates one class from the remaining classes, but this would lead to class imbalance problem, due to huge difference in the number of samples. For a K-class classification problem, K m-variable quadratic programming problems are solved by OAA-SVM.

2.2. One-Against-One

Another popular implementation for SVM multi-category classification is the one-against-one (OAO) method. It was introduced in [8], and this approach was first used for extension of SVM in [9]. It constructs $K(K-1)/2$ pairwise binary classifiers, where each one is trained on data from two classes. For training data from the i^{th} and the j^{th} classes, we solve the following binary classification problem:

$$\begin{aligned}
\min_{w^{i,j},b^{i,j},\xi^{i,j}} \quad & \frac{1}{2}\|w^{ij}\| + C\sum_t \xi_t^{ij}(w^{ij})^T \\
\text{subject to} \quad & (w^{ij})^T\phi(x_t) + b^{ij} \geq 1-\xi_t^{ij}, \;\; if\ y_t = i \\
& (w^{ij})^T\phi(x_t) + b^{ij} \leq -1+\xi_t^{ij}, \;\; if\ y_t = j \\
& \xi_t^{ij} \geq 0.
\end{aligned} \tag{4}$$

There are different methods for doing the future testing after all $K(K-1)/2$ pairwise classifiers are constructed. Most commonly voting strategy is used [10]. The voting rule means that if $sign((w^{ij})^T\phi(x)+b^{ij})$ decides that x is in i^{th} class, then one vote is added in the i^{th} class. Otherwise, the votes in j^{th} class are incremented by one. The final class label for the test pattern x is predicted based on the largest number of votes in a class. This voting approach is also called as the Max Wins strategy. In case of tie between two or more classes, the simplest strategy is to choose any one class at random. Another approach is to choose the class with the smallest index.

Practically we solve the dual of (4) whose number of variables is the same as the number of patterns in two classes. Hence if in average each class has m/K data points, we have to solve $K(K-1)/2$ quadratic programming problems where each of them has about $2 * m/K$ variables.

2.3. Directed Acyclic Graph

Directed acyclic graph SVM (DAGSVM) was proposed by Platt et al. [11] in 1999. Its training phase is the same as that of OAO approach of solving $K(K-1)/2$ pair-wise binary SVMs. However, in its testing phase, DAGSVM uses a rooted binary directed acyclic graph which has $K(K-1)/2$ internal nodes and K leaves. Each node is a binary SVM of i^{th} and j^{th} classes. For a given test pattern x, starting at the root node, the binary decision function is evaluated. Then it moves to either left or right subtree, depending on the output value. Therefore, it traverses a path from root to a leaf (terminating) node, which indicates the predicted class. An advantage of using a DAG is that some analysis of generalization can be established. There are still no similar theoretical results for one-against-all and one-against-one methods yet. In addition, its testing time is less than the one-against-one method.

2.4. Multi-SVM Optimization Problem with All Data

Vapnik [4] and Weston [12] proposed an an approach for multi-category problems by solving one single optimization problem was proposed. It constructs K pair-wise class rules where the i^{th} function $w_j^T\phi(x) + b$ separates training vectors of i^{th} class from the other vectors. Hence, there are K decision functions which are simultaneously obtained by solving only one optimization problem. The formulation is as follows:

$$\begin{aligned} \min_{w,b,\xi} \quad & \frac{1}{2}\sum_{i=1}^{K}\|w_i\| + C\sum_{j=1}^{m}\sum_{i\neq y_j}\xi_j^i w_{y_j}^T\phi(x_j) \\ & +b_{y_j} \geq w_i^T\phi(x_j) + b_i + 2 - \xi_j^i \\ & \xi_j^i \geq 0, \;\; j = 1, ..., m, \;\; i \in \{1, ..., K\}. \end{aligned} \tag{5}$$

The decision function is given as

$$\arg\max_{i=1,\ldots,K} \quad (w_i^T \phi(x) + b_i). \tag{6}$$

Crammer and Singer [13] proposed another approach for multi-category problems by solving a single optimization problem, given as

$$\begin{aligned} \min_{w_i,\xi_j} \quad & \frac{1}{2}\sum_{i=1}^{K}\|w_i\| + C\sum_{j=1}^{m}\xi_j w_{y_j}^T\phi(x_j) \\ & w_i^T\phi(x_j) \geq e_j^i - \xi_j, \;\; j = 1,...,m, \end{aligned} \tag{7}$$

where $e_j^i \equiv 1 - \delta_{y_i,m}$ and

$$\delta_{y_i,m} \equiv \begin{Bmatrix} 1 & if\ y_j = i \\ 0 & if\ y_j \neq i \end{Bmatrix}. \tag{8}$$

The decision function is given as

$$\arg\max_{i=1,\ldots,K} \quad w_i^T \phi(x). \tag{9}$$

3. Multi-Category Algorithms for Non-Parallel Hyperplane Classifiers

Non-parallel hyperplane classifiers (NPHCs) are variants of SVM, which obtain a pair of non-parallel hyperplanes instead of a single separating hyperplane. Some of these classifiers are Twin Support Vector Machines (TWSVM) [14], Generalised Eigenvalue Proximal SVM [15], Angle-based Twin SVM [16] and many others. In this section, few popular approaches for multi-category extension of NPHCs are discussed.

3.1. Ternary Decision Structure

The speed while learning a model is a major challenge for multi-class classification problems in SVMs. Twin support Vector Machine (TWSVM) (discussed in Chapter-2) is four times faster than SVM, while learning a model, as it solves two smaller QPPs. Further TWSVM overcomes the unbalance problem in two classes by choosing two different penalty variables for different classes. Because of the strength of TWSVM, it is advantageous to extend it to multi-category scenario. In this chapter, multi-category approaches for non-parallel hyperplane classifiers (NPHCs) are discussed: Ternary Decision Structure (TDS) and Binary Tree (BT) for multi-category classification.

Taking motivation from [17], Ternary Decision Structure based Multi-category Twin Support Vector Machine (TDS-TWSVM) classifier is proposed [18]. TDS-TWSVM determines a decision structure of TWSVM classifiers using training data. Each decision node is split into three decision nodes labeled as $(+1, 0, -1)$, where $+1$, and -1 represent focused groups of classes and 0 represents ambiguous group of classes. Ambiguous group consists of training samples with low confidence. At each level of the decision structure, we partition K-class problem into three $K/3$-class problems, until all samples belong to only one class. TDS-TWSVM requires $\lceil log_3 K \rceil$ tests, on an average, for evaluation of test sample. The strength of this method is its divide-and-conquer approach. This formulation reduces testing time by decreasing the number of evaluations required to derive the conclusion.

TDS is a generic approach to extend any classifier to multi-category scenario. For this chapter, TWSVM is extended using TDS and is termed as TDS-TWSVM. It evaluates all the training points into an 'i-versus-j-versus-k' structure.

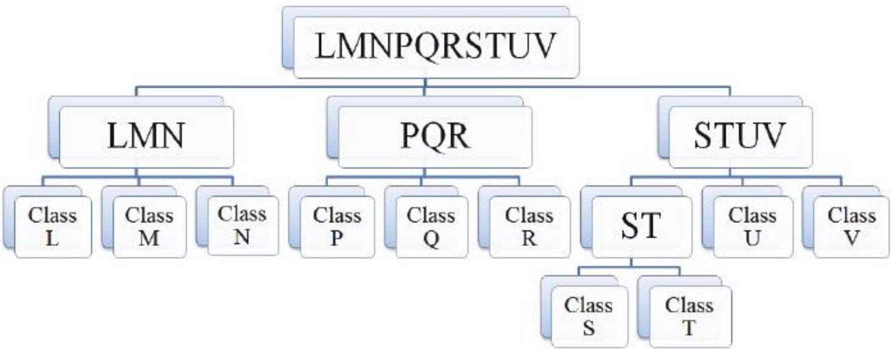


Figure 1. Ternary Decision Structure of classifiers with 10 classes.

TDS evaluates all the training points into an 'i-versus-j-versus-k' structure. During the training phase, TDS recursively divides the training data into three groups by applying k-means (k=2) clustering [19] and creates a ternary decision structure of classifiers, as shown in Fig.1. The training set is first partitioned into two clusters which leads to identification of two focused groups of classes and an ambiguous group of classes. The focused class is one where most of the samples belong to a single cluster whereas the samples of an ambiguous group are scattered in both the clusters. Therefore, TDS assigns ternary outputs $(+1, 0, -1)$ to the samples. TDS partitions each node of the decision structure into at most three groups, as shown in Fig.2. The cluster labels $(+1, 0, -1)$ are assigned to training data and three hyperplanes are determined using one-against-all approach. This in turn creates a decision structure with height $\lceil log_3 K \rceil$.

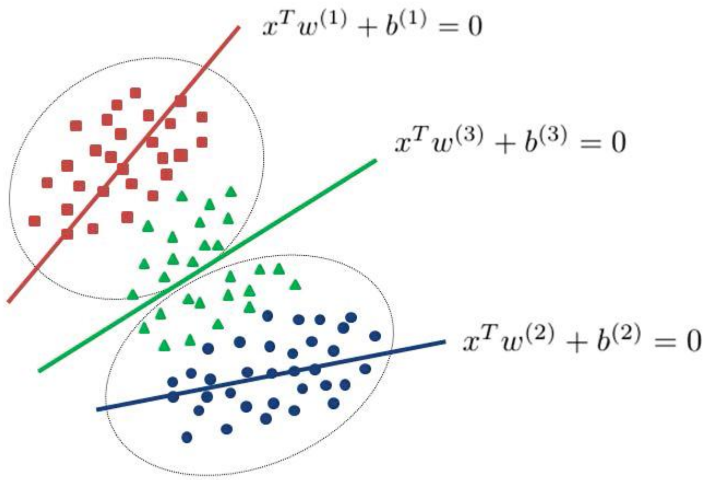


Figure 2. TDS partitions data into at most three groups.

The hyperplanes, thus obtained, are represented by nodes of the ternary decision structure and the K non-divisible nodes represent K-classes. This dynamic arrangement of classifiers significantly reduces the number of tests required in testing phase. The proposed TDS-TWSVM algorithm determines the classifier model, which is efficient in terms of accuracy and requires fewer tests for a K-class classification problem. The process of finding TDS-TWSVM classifier is explained below in the algorithm.

Input : Feature matrix F, of size $N \times n$, for the training data. Here, N and n refer to the number of training patterns and length of feature vector respectively.
Output: Labels for all test patterns.
Process:
1. Select the parameters- kernel type and kernel parameter.
2. Use K-means clustering to partition the training data into two sets. Identify two focused groups of classes with labels '+1' and '-1' respectively, and one ambiguous group of classes represented with label '0'. Here, $K = 2$ and at most three groups are obtained by partitioning the training data.
3. Take training samples of '+1', '-1' and '0' groups as class representatives and find three hyperplanes $(w^{(1)}, b^{(1)})$, $(w^{(2)}, b^{(2)})$ and $(w^{(3)}, b^{(3)})$, by applying one-against-all approach and solving the equations for TWSVM.
4. Recursively partition the data-sets and obtain TWSVM classifiers until each group represents a unique class and hence cannot be further partitioned.

Algorithm 1: Ternary Decision Structure for Multi-category Twin Support Vector Machine (TDS-TWSVM).

To test a new pattern, we find its distance from all the K hyperplanes corresponding to positive classes and the actual class label of nearest hyperplane is assigned to the test pattern i.e., the test pattern $x \in R^n$ is assigned to class r $(r = 1 \; to \; K)$, depending on which of the K hyperplanes given by (10) it lies closer to, i.e.,

$$x^T w^{(r)} + b^{(r)} = \min_{l=1:K} \frac{|x^T w^{(l)} + b^{(l)}|}{\|w^{(l)}\|_2}, \tag{10}$$

where $|.|$ is the absolute distance of point x from the plane $x^T w^{(l)} + b^{(l)} = 0$. Once we have built the classifier model, we can test a new pattern by selecting the nearest hyperplane at each level of the decision node, as in (10) and traverse through the decision structure until we reach a terminating node. The class label of terminating node is then assigned to the test pattern. With a balanced ternary structure, a K-class problem would require only $\lceil log_3 K \rceil$ tests. Also, at each level, the number of training samples used by TDS diminishes with the expansion of decision structure. Hence the order of QPP reduces as the height of the structure increases.

3.2. Binary Tree for Multi-Category Classification

Binary Tree (BT) builds the multi-category classifier model by recursively dividing the training data into two groups and creates a binary tree of classifiers [18]. At each level of the binary tree, training data is partitioned into two groups by applying K-means ($K = 2$) clustering [19] and the hyperplanes are determined for the two groups using any non-parallel hyperplane classifiers like TWSVM. This process is repeated until further partitioning is not possible. The BT classifier model thus obtained can be used to assign the label to the test pattern. The distance of the new pattern is calculated form both the hyperplanes, at each level and the group with nearer hyperplane is selected, as given in (10). We repeat until a leaf node is reached and assign the label of leaf node to the test pattern. BT determines $(K-1)$ NHCA classifiers for a K-class problem, but the size of the problem diminishes as we traverse down the binary tree. For testing, BT requires at most $\lceil log_2 K \rceil$ binary evaluations.

3.3. One-Against-One Twin Support Vector Machine

Let a K-class dataset consists of m patterns, represented by $X \in \mathbb{R}^{m \times n}$ and each pattern is associated with a label $y \in \{1, 2, ..., K\}$. We define $i \in \{1, 2, ..., K\}$ and $m = m_1 + m_2 + ... + m_K$. One-Against-One TWSVM (OAA-TWSVM) [20] solves K binary TWSVM problems, where the i^{th} problem presumes i^{th} class as positive and remaining all patterns as negative class. Let the data for i^{th} class be represented by X_i and the remaining data points are given by $\widehat{X}_i$, where $X_i \in \mathbb{R}^{m_i \times n}$. Here, m_i represents the number of patterns in i^{th} class. OAA-TWSVM formulates K binary TWSVM problems to obtain K positive hyperplanes, given by

$$x^T w_i + b_i = 0, \quad i \in \{1, 2, ..., K\}. \tag{11}$$

For each class i ($i = 1, 2, ..., K$), the positive hyperplane is generated by solving the optimization problem for TWSVM, with positive class $A = X_i$ and negative class $B = \widehat{X}_i$. The i^{th} hyperplane thus obtained would be proximal to the data points of class i. The constraints require that the hyperplane should be at least unit away from the patterns of other $(K-1)$ classes. The class imbalance problem is taken care by choosing the proper penalty variable c_i for the i^{th} class.

A test pattern x is assigned label r $(r = 1, 2, ..., K)$, based on minimum distance from the hyperplanes given by Eq.(11), i.e.,

$$x^T w^{(r)} + b^{(r)} = \min_{l=1:K} \frac{|x^T w^{(l)} + b^{(l)}|}{\|w^{(l)}\|_2}, \tag{12}$$

where $|.|$ is the absolute distance of point x from the l^{th} hyperplane. OAA-TWSVM is computationally very expensive as it solves K QPPs each of order $O((\frac{K-1}{K})m)^3$.

3.4. Twin-KSVC

Angulo et al. [21] proposed a multi-category classification algorithm, called support vector classification regression machine for K-class classification (K-SVCR) which evaluates all the training points into 'one-versus-one-versus-rest' structure. Working on the lines of K-SVCR, Xu et al. [17] proposed Twin-KSVC for multi-category classification. Twin-KSVC presented a TWSVM like binary classifier, which is extended using One-Against-One (OAO) multi-category approach. Twin-KSVC selects two focused classes $(A \in \mathbb{R}^{m_1 \times n}, B \in \mathbb{R}^{m_2 \times n})$ from K classes and constructs two nonparallel hyperplanes. The patterns of the remaining $(K - 2)$ classes (represented by $C \in \mathbb{R}^{(m-m_1-m_2)\times n}$) are mapped into a region between these two hyperplanes. Here, m, m_1 and m_2 represent the total number of patterns in the dataset, number of patterns in positive and negative focused classes respectively. The positive hyperplane $(w_1,\ b_1)$ is obtained by solving the following problem:

$$\begin{aligned}
\min_{w_1, b_1, \xi_1, \eta_1} \quad & \frac{1}{2}\|Aw_1 + e_1 b_1\|_2^2 + c_1 e_2^T \xi_1 + c_2 e_3^T \eta_1 \\
\text{subject to} \quad & -(Bw_1 + e_2 b_1) + \xi_1 \geq e_2, \\
& -(Cw_1 + e_3 b_1) + \eta_1 \geq e_3(1 - \epsilon), \\
& \xi_1 \geq 0, \quad \eta_1 \geq 0.
\end{aligned} \tag{13}$$

The constraints require that the positive hyperplane should be at least unit distance away from the patterns of negative class and $(1 - \epsilon)$ distance away from rest of the patterns. The second and third terms of the objective function try to minimize the error due to misclassification of patterns belonging to B and C, represented by ξ_1 and η_1 respectively. The negative hyperplane is obtained by solving the following problem:

$$\begin{aligned} \min_{w_2,b_2,\xi_2,\eta_2} \quad & \frac{1}{2}\|Bw_2 + e_2 b_2\|_2^2 + c_3 e_1^T \xi_2 + c_4 e_3^T \eta_2 \\ \text{subject to} \quad & (Aw_2 + e_1 b_2) + \xi_2 \geq e_1, \\ & (Cw_2 + e_3 b_2) + \eta_2 \geq e_3(1-\epsilon), \\ & \xi_2 \geq 0, \quad \eta_2 \geq 0. \end{aligned} \tag{14}$$

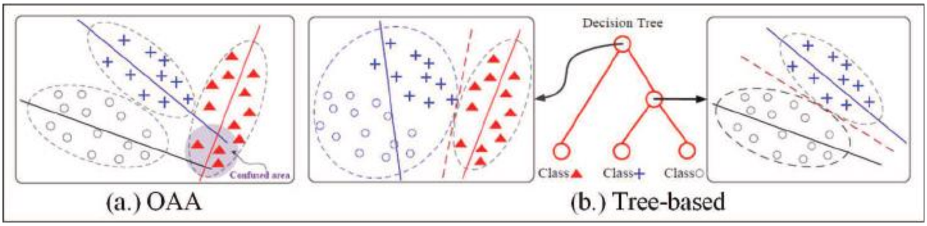


Figure 3. TDS versus OAA for non-parallel hyperplane classifiers with 3 classes.

3.5. Few Other Approaches for Multi-Category Classification

OAA-SVM classification using decision tree is proposed by Kumar et al. in [22]. Chen et al. proposed multiclass support vector classification via coding and regression [23]. Jayadeva et al. proposed fuzzy linear proximal support vector machines for multi-category data classification [24]. Lei et al. propose Half-Against-Half (HAH) multiclass-SVM [25]. HAH is built via recursively dividing the training dataset of K classes into two subsets of classes. Shao et al. propose a decision tree TWSVM (DTTSVM) for multi-class classification [26], by constructing a binary tree based on the best separating principle, which maximizes the distance between the classes. Xie et al. have extended TWSVM for multi-class classification [20] using OAA approach.

CONCLUSION

The tree-based multi-category approaches are more efficient than OAA, regarding classification accuracy as well as learning and testing time. TDS requires

$\lceil log_3 K \rceil$ comparisons for evaluating test data as compared to $\lceil log_2 K \rceil$ comparisons required by BT and K comparisons required by OAA approaches, as shown in Figure 3. Thus, TDS requires minimum testing time.

REFERENCES

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

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

[3] Cherkassky Vladimir and Mulier Filip M., *Learning from data: concepts, theory, and methods*. John Wiley & Sons, 2007.

[4] Vapnik Vladimir N. An overview of statistical learning theory. *IEEE Transactions on Neural Networks,*, 10(5):988–999, 1999.

[5] Vapnik Vladimir, *The nature of statistical learning theory*. Springer Science & Business Media, 2000.

[6] Hsu Chih-Wei and Lin Chih-Jen, A comparison of methods for multi-class support vector machines. *IEEE Transactions on Neural Networks,*, 13(2):415–425, 2002.

[7] Bottou Léon, Cortes Corinna, Denker John S., Drucker Harris, Guyon Isabelle, Jackel Larry D., LeCun Yann, Muller Urs A., Sackinger Edward, Simard Patrice, et al. Comparison of classifier methods: a case study in handwritten digit recognition. In *Proceedings of the 12th IAPR International Conference on Pattern Recognition, Vol. 3-Conference C: Signal Processing (Cat. No. 94CH3440-5)*, volume 2, pages 77–82. IEEE, 1994.

[8] Knerr Stefan, Personnaz Léon, and Dreyfus Gérard, Single-layer learning revisited: a stepwise procedure for building and training a neural network. In *Neurocomputing*, pages 41–50. Springer, 1990.

[9] Kreßel Ulrich H.-G. Pairwise classification and support vector machines. In *Advances in kernel methods*, pages 255–268. MIT Press, 1999.

[10] Hastie Trevor and Tibshirani Robert, Classification by pairwise coupling. In *Advances in neural information processing systems*, pages 507–513, 1998.

[11] Platt John C., Cristianini Nello, and Shawe-Taylor John, Large margin dags for multiclass classification. In *nips*, volume 12, pages 547–553, 1999.

[12] Weston Jason and Watkins Chris, Multi-class support vector machines. Technical report, Citeseer, 1998.

[13] Crammer Koby and Singer Yoram, On the learnability and design of output codes for multiclass problems. *Machine learning*, 47(2-3):201–233, 2002.

[14] Jayadeva, Khemchandani R., and Chandra Suresh, Twin support vector machines for pattern classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(5):905–910, 2007.

[15] Mangasarian Olvi L. and Wild Edward W., Multisurface proximal support vector machine classification via generalized eigenvalues. *IEEE Transactions on Pattern Analysis and Machine Intelligence,*, 28(1):69–74, 2006.

[16] Khemchandani Reshma, Saigal Pooja, and Chandra Suresh, Angle-based twin support vector machine. *Annals of Operations Research*, 269(1-2):387–417, 2018.

[17] Xu Yitian, Guo Rui, and Wang Laisheng, A twin multi-class classification support vector machine. *Cognitive computation*, 5(4):580–588, 2013.

[18] Khemchandani Reshma and Saigal Pooja, Color image classification and retrieval through ternary decision structure based multi-category twsvm. *Neurocomputing*, 165:444–455, 2015.

[19] Hartigan John A. and Wong Manchek A., Algorithm as 136: A k-means clustering algorithm. *Applied Statistics*, pages 100–108, 1979.

[20] Xie Juanying, Hone Kate, Xie Weixin, Gao Xinbo, Shi Yong, and Liu Xiaohui, Extending twin support vector machine classifier for multi-category classification problems. *Intelligent Data Analysis*, 17(4):649–664, 2013.

[21] Angulo Cecilio, Parra Xavier, and Catala Andreu, K-svcr. a support vector machine for multi-class classification. *Neurocomputing*, 55(1):57–77, 2003.

[22] Arun Kumar M. and Gopal Madan, Fast multiclass svm classification using decision tree based one-against-all method. *Neural processing letters*, 32(3):311–323, 2010.

[23] Chen Pei-Chun, Lee Kuang-Yao, Lee Tsung-Ju, Lee Yuh-Jye, and Huang Su-Yun, Multiclass support vector classification via coding and regression. *Neurocomputing*, 73(7):1501–1512, 2010.

[24] Khemchandani Reshma, Chandra Suresh, et al., Fuzzy linear proximal support vector machines for multi-category data classification. *Neurocomputing*, 67:426–435, 2005.

[25] Lei Hansheng and Govindaraju Venu, Half-against-half multi-class support vector machines. In *Multiple Classifier Systems*, pages 156–164. Springer, 2005.

[26] Shao Yuan-Hai, Chen Wei-Jie, Huang Wen-Biao, Yang Zhi-Min, and Deng Nai-Yang, The best separating decision tree twin support vector machine for multi-class classification. *Procedia Computer Science*, 17:1032–1038, 2013.

In: Support-Vector Machines
Editor: Pooja Saigal
ISBN: 978-1-53618-757-1

Chapter 8

SIMULTANEOUS PREDICTION OF THE DENSITY AND VISCOSITY FOR THE TERNARY SYSTEM WATER-ETHANOL–ETHYLENE GLYCOL IONIC LIQUIDS USING SUPPORT VECTOR MACHINE

***Ehsan Kianfar*[1], *H. Mazaheria*[2,*] *and Reza Azimikia*[2]**

[1]Department of Chemical Engineering, Arak Branch, Islamic Azad University, Arak, Iran; Young Researchers and Elite Club, Gachsaran Branch, Islamic Azad University, Gachsaran, Iran

[2]Department of Chemical Engineering, Arak Branch, Islamic Azad University, Arak, Iran

* Corresponding Author's Email: h-mazaheri@iau-arak.ac.ir.

ABSTRACT

In this chapter, to predict the density and viscosity of water-ethanol-ethylene glycol as a function of temperature ternary mixtures of these compounds, a model of support vector machine, are provided. Ionic liquids are a new class of compounds that in recent years much attention has been paid. These compounds as green solvents in many chemical reactions as well as advanced materials used in the structure. Density and viscosity data of the ternary system of water-ethanol-ethylene glycol solutions were obtained from the literature and an SVM (Support vector machine)the model was used to predict the density and viscosity of this system. The determination coefficient for density and viscosity of the ternary system of water ethanol-ethylene glycol 0.9854 and 0.9892, and mean square error of its density and viscosity is obtained 4.6572×10^{-4} and 3.4920×10^{-4} μmole /lit, respectively. The results confirmed that the SVM(Support vector machines) method can predict the density and viscosity of the ternary system of water- ethanol-ethylene glycol as a function of temperature, using an SVM (Support vector machines) is done well.

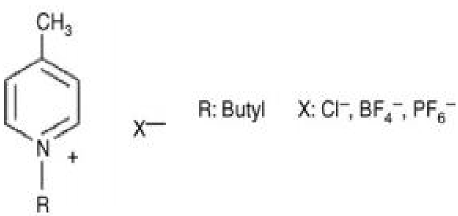
Keywords: water, ethanol, ethylene glycol, SVM (support vector machines), density, viscosity

INTRODUCTION

Ethylene glycol is an organic solvent used in extractive distillation to separate water and ethanol mixtures [1-2]. Ethylene glycol is an organic compound of the family of two-factor alcohols. It was first prepared in 1959 by the French scientist Chaser Adolf Wertz. The abbreviation is ethylene glycol MEG [3]. The most important use of ethylene glycol is antifreeze [4]. In this paper, density Laboratory fluid and dynamic viscosity of pure ethylene glycol, as well as a triple system of water and ethanol and ethylene glycol in a wide temperature range and at atmospheric pressure [5]. To design or stimulate this process, sound knowledge is important from the physical and transitive properties of this triple mixture [6]. Recently, ionic liquids have been considered by researchers due to their unique environmental characteristics [7].

Determination of thermodynamic properties for use in the industry [8]. Access to this information must be costly and time-consuming. In spite of the vast advances in technology, most analytical devices cannot directly analyze small amounts of analytic in real samples (complex tissues) [9]. However, to solve this problem, the sampling steps are used before analysis. In addition to separating the species from its complex matrix, it can simultaneously provide analytic concentration [10]. As a result, it is possible to measure analytic at very low concentrations. Liquid discharge is a classical method of preparation, most of which are standard methods [11]. Despite its widespread use and popularity, disadvantages such as emulsion formation are time-consuming and boring and are classified as multi-step preparation methods that require high volumes of expensive and toxic organic solvents [12]. It can cause serious damage to human health and the environment [13] another classic method of sample preparation is solid-phase extraction. Although this method requires less organic solvent than liquid extraction, it is also time-consuming, high cost, low repeatability, and high consumption. An organic solvent is suffering [14]. The search for new preparation methods has never been interrupted. Therefore, extensive research into separation science is associated with the development of new sample preparation techniques that have the benefits of fast, simple and high performance. And it’s good to overcome the problem of using organic solvents. Ionic fluids are compounds that have revolutionized in the research and chemical industries in recent years [15]. These compounds, which are green except for halogen, play a very important role in reducing the use of harmful, toxic and harmful compounds in the environment. Ionic data can replace many common solvents in the pharmaceutical industry [16]. Today, ionic liquids refer to organic compounds that are liquid ions formed at 100 °C. One of the reasons Today’s research on ionic liquids has intensified that scientists are looking for a position the yen is suitable for organic solvent escaping in industries. Volatile organic solvents are the most important source of environmental pollution in the chemical industry and arbitration [17]. This does not mean, however, that ionic fluids are entirely green solvents, even some they are highly toxic. There are various types of ionic liquids, such as ionic liquids at room temperature, chiral ion fluids, water ion liquids, and so on [18]. Ionic ions at room temperature are fluid ions at room temperature and have a very wide application in the chemistry of ionic liquids. The purpose of this paper is to use generalized linear modeling called "backup vector machine" to simultaneously predict the density and viscosity of a triple system of water, ethanol and ethylene glycol [19]. The structures of both groups are shown in Figure 1. Diverse Cations and Anions are used to prepare ionic liquids, making ionic liquids with dedicated uses or strengthened physicochemical properties. The conventional Anions include BF^4, BF^6, Br^-, Cl^-, etc., [20]. In this chapter, based on the linear combination, the decisions on the category and regression have been adopted. The supported vector machine used in this work belongs to the Grendel method, which has systems for the efficient execution of linear learning machines in the kernel space so that they maintain the properties of optimality and generalizability. The variables studied in the chapter the thermodynamic properties of triple mixtures are the same dependent variables, and the viscosity density variables are considered as independent variables and, on the other hand, the variables of the backup vector machine model should be optimized.

(a) Imidazolium



(b) Pyridinium

Figure 1. Ionic liquids Cation with (a) Imidazolium (b)) Pyridinium [20].

EXPERIMENTAL

Methodology Validation

The required data was found from the chapter with temperature-based density, viscosity amounts to Table 1 [21]. A support vector machine is another simple algorithm that every machine-learning expert should have in his/her arsenal. The support vector machine is highly preferred by many as it produces significant accuracy with less computation power. Support Vector Machine, abbreviated as SVM can be used for both regression and classification tasks. However, it is widely used in classification objectives. The objective of the support vector machine algorithm is to find a hyperplane in N-dimensional space (N — the number of features) that distinctly classifies the data points. To separate the two classes of data points, there are many possible hyperplanes that could be chosen. Our objective is to find a plane that has the maximum margin, i.e the maximum distance between data points of both classes. Maximizing the margin distance provides some reinforcement so that future data points can be classified with more confidence. Figure 2 shows the test design flowchart/ algorithm.

Table 1. Density and viscosity values for water and ethanol and ethylene glycol in terms of temperature [21]

temperature	**water**	**ethanol**	**Ethylene glycol**	**density**	**viscosity**
298.15	0.4024	0.2725	0.3251	0.98438	4.578
308.15	0.4024	0.2725	0.3251	00.97668	3.383
318.15	0.4024	0.2725	0.3251	0.96882	2.566
328.15	0.4024	0.2725	0.3251	0.96078	2.225
298.15	0.8369	0.1135	0.0496	0.97758	2.375
308.15	0.8369	0.1135	0.0496	0.97123	1.731
318.15	0.8369	0.1135	0.0496	0.96445	1.317
328.15	0.8369	0.1135	0.0496	0.95729	1.030
298.15	0.7015	0.2350	0.0635	0.94283	2.755
308.15	0.7015	0.2350	0.0635	0.93501	2.006

Table 1. (Continued)

temperature	water	ethanol	Ethylene glycol	density	viscosity
318.15	0.7015	0.2350	0.0635	0.92695	1.514
328.15	0.7015	0.2350	0.0635	0.91861	1.192
298.15	0.7496	0.0969	0.1535	1.01146	3.165
308.15	0.7496	0.0969	0.1535	1.00478	2.322
318.15	0.7496	0.0969	0.1535	0.99779	1.769
328.15	0.7496	0.0969	0.1535	0.99051	1.387
298.15	0.6972	0.1921	0.1107	0.96895	3.056
308.15	0.6972	0.1921	0.1107	0.96144	3.239
318.15	0.6972	0.1921	0.1107	0.95368	1.690
328.15	0.6972	0.1921	0.1107	0.94565	1.324
298.15	0.6064	0.2391	0.1545	0.96407	3.334
308.15	0.6064	0.2391	0.1545	0.95631	2.464
318.15	0.6064	0.2391	0.1545	0.94834	1.872
328.15	0.6064	0.2391	0.1545	0.94014	1.472
298.15	0.3099	0.2050	0.4851	1.02378	6.530
308.15	0.3099	0.2050	0.4851	1.01634	4.662
318.15	0.3099	0.2050	0.4851	1.00876	3.495
328.15	0.3099	0.2050	0.4851	1.00104	2.700
298.15	0.5988	0.2058	0.1954	0.98327	3.691
308.15	0.5988	0.2058	0.1954	0.9757	2.708
318.15	0.5988	0.2058	0.1954	0.96791	2.058
328.15	0.5988	0.2058	0.1954	0.9599	1.612
298.15	0.5101	0.2408	0.2491	0.98186	4.055
308.15	0.5101	0.2408	0.2491	0.97418	2.987
318.15	0.5101	0.2408	0.2491	0.96632	2.273
328.15	0.5101	0.2408	0.2491	0.95828	1.779
298.15	0.6944	0.1130	0.1926	1.01822	3.584
308.15	0.6944	0.1130	0.1926	1.01133	2.627
318.15	0.6944	0.1130	0.1926	1.00419	1.999
328.15	0.6944	0.1130	0.1926	0.99679	1.565
298.15	0.6172	0.1056	0.2772	1.03042	4.458
308.15	0.6172	0.1056	0.2772	1.01605	3.247
318.15	0.6172	0.1056	0.2772	1.00854	2.463
328.15	0.6172	0.1056	0.2772	0.94708	1.913

temperature	**water**	**ethanol**	**Ethylene glycol**	**density**	**viscosity**
298.15	0.7674	0.1960	0.0366	0.94708	2.562
308.15	0.7674	0.1960	0.0366	0.93955	1.854
318.15	0.7674	0.1960	0.0366	0.93173	1.398
328.15	0.7674	0.1960	0.0366	0.92362	1.090
298.15	0.2892	0.2887	0.4221	0.99282	5.307
308.15	0.2892	0.2887	0.4221	0.98518	3.902
318.15	0.2892	0.2887	0.4221	0.97739	2.956
328.15	0.2892	0.2887	0.4221	0.96945	2.299
298.15	0.8011	0.1000	0.0989	0.99703	2.688
308.15	0.8011	0.1000	0.0989	0.99059	1.974
318.15	0.8011	0.1000	0.0989	0.98379	1.504
328.15	0.8011	0.1000	0.0989	0.97664	1.184
298.15	0.1074	0.4050	0.4876	0.97171	5.121
308.15	0.1074	0.4050	0.4876	0.96396	3.778
318.15	0.1074	0.4050	0.4876	0.9561	2.892
298.15	0.2766	0.4220	0.3014	0.94218	3.726
298.15	0.1097	0.5862	0.3041	0.90768	3.033
298.15	0.4213	0.4769	0.1018	0.89154	2.460
298.15	0.2107	0.6868	0.1025	0.85146	1.945
298.15	0.3126	0.4726	0.2148	0.91549	3.035
298.15	0.5023	0.2981	0.1996	0.95601	3.533
298.15	0.5076	0.3919	0.1005	0.90958	2.659
298.15	0.1130	0.6858	0.2012	0.87165	2.257
298.15	0.7919	0.0490	0.1591	1.03009	2.897
298.15	0.1998	0.5973	0.2029	0.88887	2.553
298.15	0.3940	0.4086	0.1974	0.92694	3.166
298.15	0.1025	0.4945	0.4030	0.94117	3.925
298.15	0.2997	0.5938	0.1065	0.86859	2.196
298.15	0.1177	0.7891	0.0932	0.83374	1.677
298.15	0.0010	0.9071	0.0919	0.81674	1.432
298.15	0.0010	0.7826	0.2164	0.85881	2.017
298.15	0.00090	0.6667	0.3324	0.89753	2.789
298.15	0.00090	0.5990	0.4001	0.91981	3.333

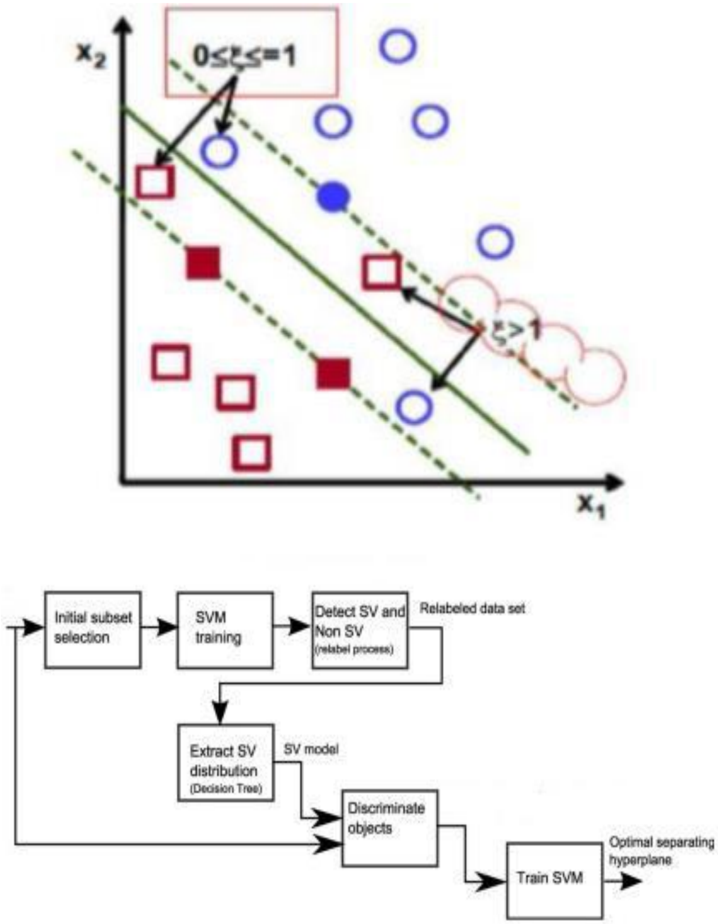


Figure 2. Design flowchart/ algorithm support vector machine.

RESULTS AND DISCUSSION

In this chapter, the backup machine model was used to create the model in Assam, which is given in Tables 2 and 3 for the value of R^2, MSE, sigma, gamma for density and viscosity in the training category and the test category. 78 data as inputs for this paper, 62 data were used as training groups and 16 data were used as test classes. Data input to the backup machine and computation by it were performed 15 times with the results obtained for the density and the viscosity is shown in Tables 2 and 3. Given the obtained values, R^2, MSE, sigma, gamma, where the lowest MSE and the highest R^2 coefficient were obtained, as the optimum values in the calculation of the ninth stage of software implementation for the density and the fifth stage of software implementation for viscosity. The optimal R^2 values for the training and testing arm in both cases show that there is a fairly good correlation between the predictive and experimental values. Therefore, the proposed model can be a successful model for

Table 2. The data obtained by the rectifying machine vector in the training and test category for the density of the triple system of water and ethanol and ethylene glycol

Steps to run the software	***gamma***	***Sigma2***	**Train**		**test**	
			R^2	*MSE*	R^2	*MSE*
1	247.6291	0.2101193	1.0000	$5.1328*10^{-7}$	0.8972	0.0026
2	81.8542	0.1505280	1.0000	$2.7551*10^{-6}$	0.8120	0.0043
3	200.3167	0.2001038	1.0000	$6.9913 *10^{-7}$	0.8858	0.0029
4	329.5379	0.1716492	1.0000	$2.3653*10^{-7}$	0.8477	0.0036
5	108.0751	0.1671832	1.0000	$1.7645*10^{-6}$	0.8399	0.0038
6	206.4259	0.1982543	1.0000	$5.5601*10^{-7}$	0.8836	0.0029
7	277.3569	0.2117261	1.0000	$4.2543*10^{-7}$	0.8990	0.0026
8	438.0806	0.3766458	1.0000	$5.2785*10^{-7}$	0.9819	$5.7082*10^{-3}$
9	452.842	0.3980327	1.0000	$5.5770*10^{-7}$	0.9854	$4.6572*10^{-4}$
10	225.5518	0.2065906	1.0000	$5.9090*10^{-7}$	0.8933	0.0027
11	177.3426	0.1934183	1.0000	$8.3294*10^{-6}$	0.8775	0.0030
12	300.861	0.2093218	1.0000	$3.6289*10^{-7}$	0.8964	0.0026
13	410.7357	0.3413949	1.0000	$4.8611*10^{-7}$	0.9740	$7.9614*10^{-4}$
14	361.4749	0.2482923	1.0000	$3.4360*10^{-7}$	0.9314	0.0019
15	388.7204	0.2525492	1.0000	$3.1245*10^{-7}$	0.9344	0.0018

Table 3. The data obtained by the rectifying machine vector in the training and test category for the viscosity of the triple system of water and ethanol and ethylene glycol

Steps to run the software	***gamma***	***Sigma2***	**Train**		**test**	
1	200.3167	0.2001038	1.0000	6.9913*10-7	0.8858	0.0029
2	231.7204	0.2025492	1.0000	2.7551*10-6	0.8887	0.0028
3	302.4856	0.2632014	1.0000	5.0897*10-7	0.9414	0.0016
4	279.5518	0.2265906	1.0000	4.0555*10-7	0.8934	0.0027
5	488.1297	0.4280164	1.00005. 7	7609*10-	0.989230-4	4920*1
6	445.4749	0.4133784	1.0000	6.1395*10-7	0.9875	4.0275*10-4
7	227.7399	0.1918428	1.0000	5.3016*10-7	0.8756	0.0031
8	390.1423	0.2462117	1.0000	2.9747*10-7	0.9299	0.0019
9	457.3605	0.3797392	1.0000	5.0260*10-7	0.9824	5.5380*10-4
10	376.3066	0.2994503	1.0000	4.4222*10-7	0.9599	0.0012
11	420.8256	0.3692034	1.0000	5.4027*10-6	0.9805	6.1293*10-4
12	248.242	0.208641	1.0000	5.0629*10-7	0.8956	0.0027
13	296.7531	0.2439018	1.0000	4.6624*10-7	0.9281	0.0019
14	436.7166	0.3976163	1.0000	5.8695*10-7	0.9853	4.6806*10-4
15	355.0319	0.2709318	1.0000	4.0961*10-7	0.9460	0.0015

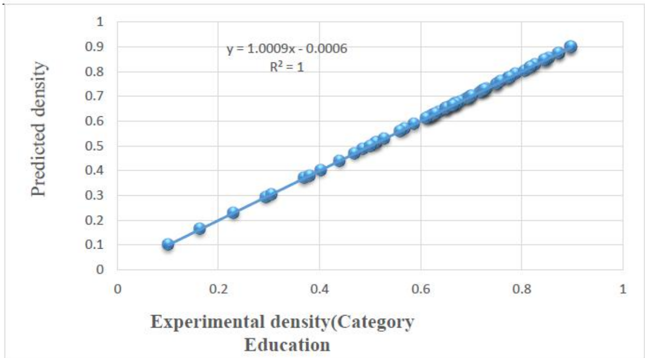


Figure 3. The relationship between experimental density and density predicted by the proposed model in the training category.

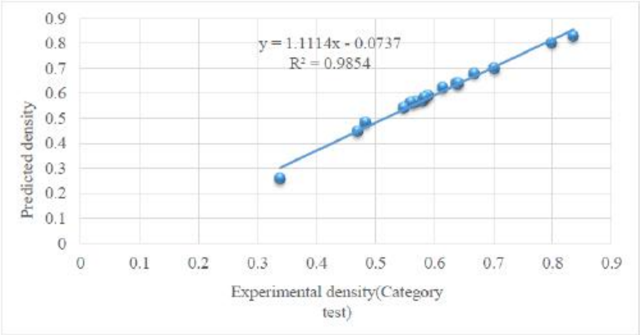


Figure 4. The relationship between experimental density and predicted density by the proposed model in the test category.

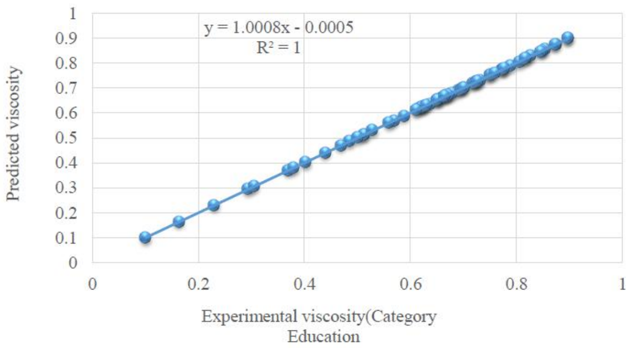


Figure 5. The relationship between experimental viscosity and viscosity predicted by the proposed model in the training category.

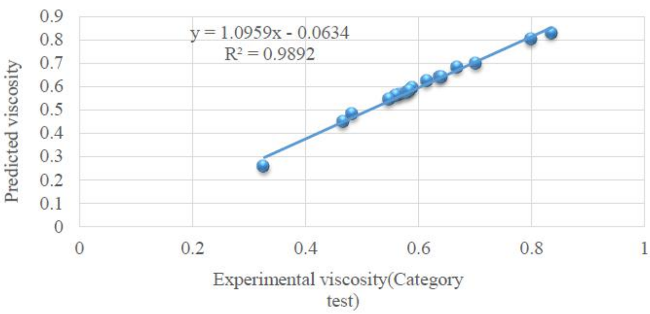


Figure 6. The relationship between experimental viscosity and predicted viscosity by the proposed model in the test category.

predicting the density and viscosity of the triple system of water, ethanol and ethylene glycol at the same time. Figure 3 shows the correlation between the experimental data and the predicted values of the trivalent density of water, ethanol and ethylene glycol by the proposed model. The determination coefficient for the density of this system was obtained in the 1.0000 instruction class, as shown in Figure 4. The coefficient of determination for the density of this system was obtained in the test category 0.9854. Figure 5 shows the correlation between the experimental data and the predicted values of the trivalent density of water, ethanol and ethylene glycol by the proposed model. The determination coefficient for the density of this system is obtained in the 1.0000 instruction class, as shown in Figure 6. The coefficient of determination for the density of this system was obtained in the test category 0.9892.

CONCLUSION

In this chapter, it is possible to simultaneously predict the density and viscosity of the triple system of water and ethanol and ethylene glycol using Support vector machines. The determination of thermodynamic properties for use in industry is important. But access to this information is costly and time-consuming. Ionic data is usually composed of asymmetric organic cations and organic or mineral anions. Anionic ion is a salt in which ions are weakly interconnected, asymmetry, the energy of the grid and hence the melting point of the ionized medium decreases. The less one is the one these solutions are unaware that all materials must be soluble to absorb the body so that they can pass through the cell membrane. Also, the nature of our surroundings is based on the dissolution and insolubility of materials. The thermodynamics of the process of fuzzy separation of several mixtures from key topics to it is a matter of great research because of the wide scope and importance of the topic. Since the retest requires a lot of time and money, and on the other hand, the number of parameters is high and the temperature is considered for each of the parameters separately. This method, which predicts all conditions simultaneously, is of particular importance. In other words, this method allows researchers in the future to achieve optimal results without re-testing and a waste of time by referring to modeling. The supported vector machine method confirmed that the prediction simultaneity Density, viscosity water, ethanol and of butylene glycol as a function of temperature shows lower errors and higher precision compared to statistical method while the use of supported vector machine is easier and quicker than statistical methods.

ACKNOWLEDGMENT

The authors gratefully a Department of Chemical Engineering, Arak Branch, Islamic Azad University, Arak, Iran and Young Researchers and Elite Club, Gachsaran Branch, Islamic Azad University, Gachsaran

REFERENCES

[1] Abbaspour. A., Baramakeh. L., 2005, Simultaneous determination of antimony and bismuth by beta-correction spectrophotometry and an artificial neural network algorithm, *Talanta*, 65, 692-699. https://www.ncbi.nlm.nih.gov/pubmed/18969854.

[2] Agrawal. V.K., Louis. B., Khadikar. P.V., 2010, Prediction of intrinsic solubility of generic drugs using MLR, ANN and SVM analyses, *European Journal of Medicinal Chemistry*, 45, 4018-4025. https://www.ncbi.nlm.nih.gov/pubmed/20584562.

[3] Akbari. E., Buntat. Z., Enzevaee. A., Ebrahimi. M., Yazdavar. A. H., Yusof. R., 2014, Analytical modeling and simulation of I–V characteristics in carbon nanotube-based gas sensors using ANN and SVR methods, *Chemometrics and Intelligent Laboratory*, 137, 173-180.

https://www.sciencedirect.com/science/article/abs/pii/S0169743914001488.

[4] Balaban. R.M., Lomakina. E.I., Safieva. R.Z., 2011, Neural network (ANN) approach to biodiesel analysis: Analysis of biodiesel density, kinematic viscosity, methanol and water contents using near infrared (NIR) spectroscopy, *Fuel*, 90, 2007-2015. https://www.sciencedirect.com/science/article/pii/S0016236110006538.

[5] Beebe, K.R., Kowalski, B.R., 1987, Binding of tissue plasminogen activator to human umbilical vein endothelial cells. *Anal. Chem.*, 705, 241-254. https://www.ncbi.nlm.nih.gov/pubmed/3111002.

[6] Bordbar. M.M., Khajehsharifi. H., Solhjoo. A., 2015. PC-ANN assisted to the determination of Vanadium (IV) ion using an optical sensor based on immobilization of Eriochorome Cyanine R on a triacetylcellulose film, Spectrochimica Acta Part A: *Molecular and Biomolecular Spectroscopy*, 151, 225-231. https://www.ncbi.nlm.nih.gov/pubmed/26142655

[7] Keskin, s., Kayrak, D., Akman, U., Hortac., 2007, "A review of ionic liquids towards supercritical fluid applications," *The Journal of Supercritical Fluids*. 43, 150-180. https://www.sciencedirect.com/science/article/pii/S0896844607002033.

[8] Khajehsharifi. H., Bordbar. M.M., 2015, A highly selective chemosensor for detection and determination ofcyanide by using an indicator displacement assay and PC-ANN and itslogic gate behavior, *Sensors and Actuators B*, 209, 1015-1022. https://www.sciencedirect.com/science/article/abs/pii/S0925400514012647.

[9] Khanchi. A.R., Khayatzadeh Mahani. M., Hajihosseini. M., Ghanadi. M., Chaloosi. M., Bani. F., 2007, Simultaneous spectrophotometric determination of caffeine and theobromine in Iranian tea by artificial neural networks and its comparison with PLS, *Food Chemistry*, 103, 1062-1068.

https://www.sciencedirect.com/science/article/pii/S0308814606006625.

[10] Liu, Q.S., Li, P.P., Welz-Biermann, U., Chen. J., Liu. X.X., 2013, "Density, dynamic viscosity, and electrical conductivity of pyridinium-based hydrophobic ionic liquids", *J. Chem. Thermodynamics*, 66, 88-94. https://www.sciencedirect.com/science/article/pii/S0021961413002279.

[11] Ni. Y., Xia. Zh., Kokot. S., 2011, A kinetic spectrophotometric method for simultaneous determination of phenol and its three derivatives with the aid of artificial neural network, *Journal of Hazardous Materials*, 192, 722-729. https://www.ncbi.nlm.nih.gov/pubmed/21719195.

[12] Raimundo Jr. I., Narayanaswamy. R., 2001, Simultaneous determination of relative humidity and ammonia in air employing an optical fibre sensor and artificial neural network, *Sensors and Actuators B,* 74, 60-68. https://www.sciencedirect.com/science/article/abs/pii/S0925400500007127.

[13] Rezaei. B., Ensafi. A.A., Shandizi. F., 2001, Simultaneous determination of cobalt and nickel by spectrophotometric method and artificial neural network, *Microchemical Journal*, 70, 35-40. https://www.sciencedirect.com/science/article/abs/pii/S0026265X01000947.

[14] Safavi. A., Abdollahi. H., Hormozi Nezhad. M.R., 2003, Artificial neural networks for simultaneous spectrophotometric differential kinetic determination of Co(II) and V(IV), *Talanta*, 45, 515-523. https://www.ncbi.nlm.nih.gov/pubmed/18968936.

[15] Smiglak, M., Hough, W.L., Rodriquez, M.D., Davis Jr., H.J, Rogers, R.D., 2007, "The third evolution of ionic liquids: active pharmaceutical ingredients," *New Journal of Chemistry*, 31, 1429-1436. https://pubs.rsc.org/en/content/articlelanding/2007/nj/b706677p#!divAbstract.

[16] Tang. L., Zeng. G.M., Shen. G.L., Zhang. Y., Huang. G., Li. J.B., 2006, Simultaneous amperometric determination of lignin peroxidase and manganese peroxidase activities in compost bioremediation using artificial neural networks, *Anal. Chem. Acta*, 579, 109-116. https://www.ncbi.nlm.nih.gov/pubmed/17723735.

[17] Tenorio-Borroto. E., Peñuelas Rivas. C.G., Vásquez Chagoyán. J.C., Castañedo. N., Prado-Prado. F.J., García-Mera. X., González-Díaz. H., 2012, ANN multiplexing model of drugs effect on macrophages; theoretical and flow cytometry study on the cytotoxicity of the anti-microbial drug G1 in spleen, *Bioorganic & Medicinal Chemistry*, 20, 6181-6194.
https://www.ncbi.nlm.nih.gov/pubmed/22981917.

[18] Zamaniyan. A., Joda. F., Behroozsarand. A., Ebrahimi. H., 2013, Application of artificial neural networks (ANN) for modeling of industrial hydrogen plant, *International Journal of Hydrogen Energy*, 38, 6289-6297.
https://www.sciencedirect.com/science/article/abs/pii/S0360319913005703.

[19] Zhao, H., Xia, S., Ma, P., 2005, "Use of ionic liquids as 'green' solvents for extractions," *Journal of Chemical Technology and Biotechnology*, 80, 1089-1096.
https://onlinelibrary.wiley.com/doi/10.1002/jctb.1333.

[20] kianfar E, Shirshahi M, Kianfar F, kianfar F. 2018. "Simultaneous prediction of the density, viscosity and electrical conductivity of pyridinium-based hydrophobic ionic liquids using artificial neural network." *Silicon*,10, 2617–2625.
https://link.springer.com/article/10.1007/s12633-018-9798-z.

[21] Quijada-Maldonado.E., Meindersma. G.W., Haan. A.B. 2013. "Viscosity and density data for the ternary system water(1)–ethanol(2)–ethylene glycol(3) between 298.15 K and 328.15 K." *J. Chem. Thermodynamics* 57: 500-505.
https://www.sciencedirect.com/science/article/pii/S0021961412003382.

About the Editor

Pooja Saigal, PhD
Associate Professor
Vivekananda School of Information Technology,
Vivekananda Institute of Professional Studies,
(Affiliated to Guru Gobind Singh Indraprastha University),
New Delhi, India
Email: pooja.saigal@vips.edu, saigal.pooja.in@gmail.com

Education: PhD (Computer Science)

Business Address: Vivekananda School of Information Technology, A-Block, Technical Campus, Vivekananda Institute of Professional Studies, AU-Block, Pitampura, New Delhi, India

Research and Professional Experience: More than 16 years of teaching experience at post-graduate and under-graduate levels

Professional Appointments: Associate Professor, Reviewer for various SCI Journals, Member of TPC of International Conferences

Honors: Awarded Gold Medal by the President of India for securing first rank in University, Research Scholarship at South Asian University (by SAARC)

Publications from the Last 3 Years:

1. "Semi-supervised Weighted Ternary Decision Structure for Multi-category Classification", Saigal P., Chandra S and Rastogi R, *Neural Processing Letters* DOI: 10.1007/s11063-020-10323-7, (Accepted on 21 July 2020).
2. "Multi-category news classification using Support Vector Machine based classifiers" Saigal, P., and Khanna, V. *SN Applied Sciences*, Springer 2(3), 1-12 (2020), Published: 20 February 2020, Electronic ISSN: 2523-3971.
3. "Multi-category ternion support vector machine." Saigal P., Chandra S and Rastogi R. *Engineering Applications of Artificial Intelligence* 85 (2019): 229-242. ISSN: 0952-1976.
4. "Angle-based twin parametric-margin support vector machine for pattern classification", Rastogi R., Saigal P., Chandra S., published in *Knowledge-Based Systems*, Elsevier 2018. ISSN: 0950-7051, Jan 2018.
5. "Angle-based twin support vector machine", Khemchandani R., Saigal P., Chandra S., *Annals of Operations Research*, Springer 2017. ISSN: 0254-5330 (Print) 1572-9338 (Online)
6. "Divide and conquer approach for semi-supervised multi-category classification through localized kernel spectral clustering", Saigal P., Khanna V. and Rastogi R., published in *Neurocomputing*, Elsevier ISSN: 0925-2312.
7. "Tree-based localized fuzzy twin support vector clustering with square loss function", Rastogi R. and Saigal P. published in *Applied Intelligence*, Springer Feb 2017, Vol. 47, Issue 1, page 96–113. ISSN: 0924-669X (Print) 1573-7497 (Online)

Dr. Pooja Saigal received her PhD degree in Computer Science from South Asian University (established by SAARC), New Delhi, India. Her research interests include Machine Learning, Optimization and Image Processing. She has proposed various supervised, unsupervised and semi-supervised machine learning algorithms and applied them on image processing problems like content-based image retrieval, segmentation etc.

She is UGC-NET qualified in Computer Science. She holds Master's Degree (2004) and Bachelor's Degree (2001) in Computer Applications. She is University Topper of MCA 2004 batch and was awarded Gold Medal by the erstwhile President of India, Dr. APJ Abdul Kalam, for her outstanding performance in Master's program (CPI: 90.24% and Percentage: 88.6%). She earned Distinction in all 32 courses of Master of Computer Applications. She secured First Rank in University in Bachelor of Computer Applications and was awarded by the Chief Minister of Haryana. She has experience of over 16 years in teaching Information Technology and Computer Science subjects at Post-graduate and Under-graduate levels. She is reviewer of various SCI Indexed Journals: *Neural Networks*, Elsevier; *IEEE Transactions on Cybernetics*; *Neurocomputing,* Elsevier; and many more. She has authored 24 research papers in international journals, books and conference proceedings.

INDEX

A

B

C

Q

R

S

T

U

V

W

X

About the Editor

Dr. Pooja Saigal is an Associate Professor in Vivekananda Institute of Professional Studies, affiliated with Guru Gobind Singh Indraprastha University, New Delhi. She received her PhD in Computer Science from South Asian University (established by SAARC), New Delhi in 2018. Her research interests include Machine Learning, Optimization and Image Processing. She has proposed various supervised, unsupervised and semi-supervised machine learning algorithms and applied them on image processing problems like content-based image retrieval, segmentation etc. She is UGC-NET qualified in Computer Science. She holds a Master's Degree (2004) and Bachelor's Degree (2001) in Computer Applications. She has authored more than 20 research papers in international journals, books and conference proceedings.

nova
science publishers
www.novapublishers.com

ISBN 978-1-53618-757-1
9 781536 187571