Comparative analysis of zoning based methods for Gujarati handwritten numeral recognition

Ankit K. Sharma[1], Dipak M. Adhyaru[2], Tanish H. Zaveri[3], *Priyank* B *Thakkar*[4]
[1]Assistant professor, [2] Professor & Head, [3]Associate Professor, [4]Associate Professor
[1,2]Instrumentation and Control Engineering Section, [3]Electronics and Communication Engineering Section, [4]Computer Engineering Section
Institute of Technology, Nirma University, Ahmedabad, Gujarat, India
ankit.sharma@nirmauni.ac.in[1], dipak.adhyaru@nirmauni.ac.in[2], ztanish@nirmauni.ac.in[3], priyank.thakkar@nirmauni.ac.in[4]

***Abstract*— Gujarati is one of the ancient Indian languages spoken widely by the people of Gujarat state. This paper is concerned with the recognition of handwritten Gujarati numerals. For recognition of Gujarati numerals zoning based Feature extraction method is used. Numeral image is divided in 16x16, 8x8, 4x4 and 2x2 Zones. After feature extraction through the zoning method, Naive Bayes classifier and multilayer feed forward neural network classifier are implemented for the classification of numerals. For the database generation, 14,000 samples of each numeral are used. The overall recognition rate of this method used for recognition of Gujarati numeral using 16x16, 8x8, 4x4 and 2x2 zoning with neural network are 93.03%, 95.92%, 91.89% and 61.78% and with Naive Bayes classifier are 75%, 85.60%, 81% and 53.75% respectively.**

***Index Terms*— Gujarati script, Neural networks, Naive Bayes classifier, Zone based feature extraction.**

I. INTRODUCTION

Handwritten numeral Recognition becomes a prime area of research because of its potential that can be used in various applications. Researchers have explored many Indian languages such as Hindi, Marathi, Telugu, Bangla, Gurumukhi, Tamil and Kannada etc., but Gujarati is yet to be researched, explored and recognized to be on part with the other Languages. This paper throws light on the Gujarati numeral recognition. Gujarati script is part of the Brahmic family and it is similar to Devanagari scripts as most of the words are derived from Sanskrit. There is no header line at the top of the letters or words in Gujarati script. Gujarati is the mother tongue of the people of Gujarat state, one of the most spoken native languages and nearly 65 million people converse in Gujarati. Gujarati numerals have various shapes and many numerals have close resemblance which creates confusion and have possibilities of incorrect recognition. This paper talks about Optical character recognition for handwritten Gujarati numerals. Optical character recognition, usually abbreviated as OCR,it translates typewritten, scanned copy of images or documents into machine encoded text. This translated machine encoded text can be easily searched, edited and processed in numbers of ways as per our requirement. Handwritten document recognition is a very challenging area for research and many researchers are putting in efforts to convert handwritten scripts to computer readable format. Handwritten numeral recognition is tedious because handwritten numerals may vary from person to person depending on their writing style, curve, thickness and the size of the numerals. Whereas printed numerals on the other hand, are simpler to identify because of their uniformity.

Numeral recognition can be done in two ways: Offline recognition and online recognition. This paper emphasizes on offline numeral recognition. Application of handwritten numeral recognition includes helping sort out or categorizes postal mails, bank cheque, code reading, postal address reading, form processing, signature verification, etc. Optical character recognition helps to reduce human efforts of manually handling and processing of documents.

II. REVIEW OF RELATED WORK

In comparison to other foreign languages like Chinese, English, and Japanese etc., not much work has been carried out in the area of Gujarati numeral recognition. It is found that Indian languages in comparison with Bangla, Hindi, Marathi and few south Indian languages, the OCR activities related to Gujarati language is very less. Shailesh A. Chaudhari and Ravi M. Gulati have worked on Separation and identification of mixed English – Gujarati printed numerals. Statistical approach is used as feature extraction with KNN classifier. An overall accuracy of 99.23% is obtained with the same and an accuracy of 99.26% for Gujarati and 99.20% for English numerals is obtained using KNN classifier [25]. Baheti M.J and Kale K.V have developed an algorithm to classify handwritten Gujarati numerals using affine invariant moment feature extraction technique. Author have used various classifier to classify Gujarati numerals an highest accuracy of 92.28% is achieved using support vector machine and accuracy of 90.04%,87.2% and 84.1% is obtained using K-Nearest Neighbor, Gaussian distribution function and principal component analysis classifier [26]. In [27] affine invariant moments based feature is used by Mamta maloo and K.V. Kale to classify handwritten Gujarati numerals. Recognition rate of 91% is obtained using support vector machine classifier.To recognize handwritten Gujarati numerals author Avani R. Vasant, Sandeep R.Vasant and Dr. G.R.Kulkarni have used Neural Network classifier. Recognition rate of 87.29%, 88.52% and 88.76% is obtained for 7x5, 14x10 and 16x16 size images respectively [28]. In [8] Minimum hamming distance classifier and k-NN classifier are used for identification of Gujarati characters and overall accuracy achieved was 67%. An algorithm for Gujarati script identification is proposed by S. K. shah and A.Sharma, algorithm uses template matching based approach for character classification [13]. Jignesh Dholakia, Atul Negi and S. Ram Mohan have used GRNN and k-NN classifier for classification of printed Gujarati characters [18].

978-1-4799-9991-0/15/$31.00 ©2015 IEEE

A. Dutta and S. Chaudhuri proposed the use of two stage feed forward neural network, trained by back propagation algorithm is used for recognition of Bangla alphanumeric handwritten characters in [4]. In [6] work presented by L. L. Lee and N. R. Gomes for recognizing handwritten numerals is shown, and it makes use of the structural feature extraction method. Another structural feature based approach for handwritten character recognition is described in [7] which displays intersections between the character and straight lines, number of horizontal and vertical lines, end points, presence of loops, holes position, number of intersections and junctions. U. Bhattacharya, T. K. Das, A. Datta, S. K. Parui and B.Chaudhuri worked on recognition of Bangla handwritten numerals [10] using neural network as a classifier. To classify Oriya handwritten numerals, K. Roy, T. Pal, U. Pal and F. Kimura have used Histograms of direction chain code of the contour points of the numerals as a feature extraction method in [12] and achieved 94.81% accuracy using Neural Network based classifier. V.L. Lajish, T.T.K. Suneesh and N.K. Narayanan worked on Malayalam handwritten character Images and for classification Kolmogrov-Sminov statistical classifier and k-NN classifier was used [11]. For recognition of Off-line Devnagari handwritten characters directional chain code feature extraction method is suggested in [14] 80.36% and 98.86% accuracy is achieved for Devnagari characters and numerals respectively. To classify Kannada and Telugu numerals a zone based and distance metric based feature extraction method is described in [17] a recognition rate of 86% for Telugu and 98% for Kannada numerals is achieved. Work on handwritten Gujarati numerals was presented by Apurva A. Desai. Author has used multi layered feed forward neural network to classify Gujarati handwritten numerals and four different profiles of numerals are used to extract its features. The accuracy achieved in this work is approximately 82% for Gujarati handwritten numerals [19]. A survey on different feature extraction methods for character recognition is described in [24].

III. DATABASE

The database collected consists of handwritten numerals written by persons of various educational background and age groups. Specially designed forms are used for database collection (shown in Figure.1). All the handwritten forms are scanned with the HP flatbed scanner at 300 dpi resolution and saved in JEPG format. In all, 14,000 numeral images are obtained, the number of images for each numeral (0-9) being 1,400.

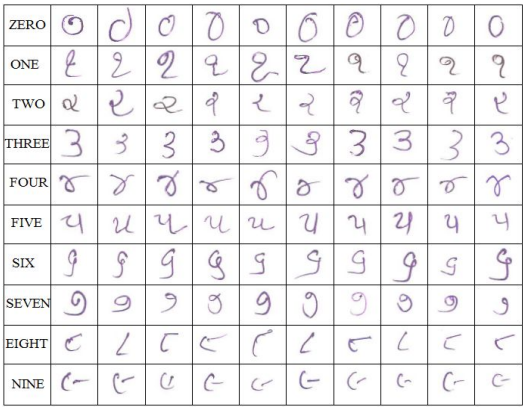

Figure.1 Special form designed for database collection.

IV. PRE-PROCESSING

A set of preprocessing steps are applied to the database images in order to remove noise and for the simplification of feature extraction procedure. Preprocessing steps include Binarization, resizing, median filtering and morphological operation such as Thinning. The input to the preprocessing step is a colored scanned image. This image is then converted to gray scale image. Gray scale image is then converted into black and white format by the process of Binarization. Otsu's Thresholding technique is used for binarization in which optimum threshold value is calculated and all the pixel intensities are converted to 0 and 1. After Binarization, numeral image is segmented from binary image and segmented image is resized to 16x16 pixels. Resizing is required because different people write in different style with different sizes. Hence it is necessary that all the numeral images should be of uniform size. After resizing, morphological thinning operation is performed. Figure. 2 shows the results of different operations adopted for preprocessing on numeral image.

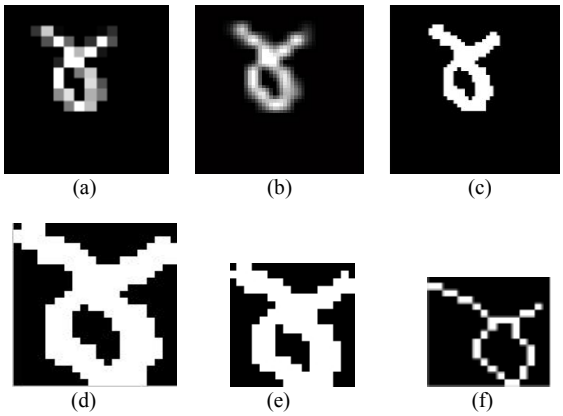

Figure.2 (a) Original numeral image (b) Image in gray scale form (c) Image in binary form (d) Segmented numeral (e) Segmented numeral resized to 16×16 (f) Thinned numeral image

V. FEATURE EXTRACTION

Feature extraction is one of the most important steps in developing an OCR system. It is required to represent the numeral images in a unique way in order to increase the classification accuracy of the system. Zone based feature extraction method is proposed for recognition of Gujarati handwritten numerals. In this method, the numeral image of size 16x16 is divided into 256, 64, 16, 4 uniform zones. Figure.3 shows the different zoning adapted in this work. For computation of feature vector, the summation of all the pixels representing the numeral in each zone is computed. If the image is divided in 16x16 zones then each zone will be consisting of one pixel value and the size of the feature vector will be 256. But, if the image is divided in 8x8 zones then each zone will be consisting of four pixels values and the size of feature vector will be 64. In this way, the feature vector will be consisting of 256, 64, 16 and 4 elements respectively depending on the type of zoning used.

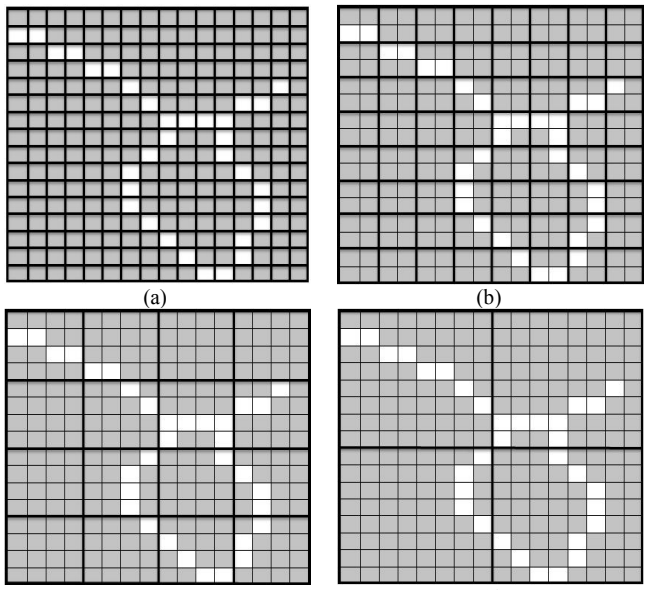

Figure.3 (a) Numeral image divided in 16x16 zones (b) Numeral image divided in 8x8 zones (c) Numeral image divided in 4x4 zones (d) Numeral image divided in 2x2 zones

VI. CLASSIFICATION

Classification is the decision making step of the OCR system. The feature vectors generated by the feature extraction method are fed to the classifier for training and testing purpose. For the proposed algorithm we have used two different classifiers, one being the multilayer feed forward neural network with Back propagation learning and the other being the Naive Bayes classifier. When the input dimensionality is high, the naive bayes classifier technique is used which is based on Bayesian theorem. Though naive Bayes is simple yet it can outperform most sophisticated classification methods. The below equation is represents Naive Bayes rule.

(1)

Based on the observation of evidences (E), the output of event (H) or hypothesis can be predicted, this is the basic idea of Bayes's rule. (1) A priori probability of H or P(H): Probability of an event before the evidence is observed. (2) A posterior probability of H or P(H | E) : Probability of an event after the evidence is observed.

The structure of the neural network classifier is shown in Figure.4. One hidden layer is used consisting of 25 neurons. Number of epochs considered is 5000. These values are decided on the basis of best performance achieved for the classification accuracy with these values in all cases.

Number of neurons in the input layer will depend on the size of the feature vector. A popular algorithm known as an error back algorithm is used for training. The goal of the training is to find the set of weight values that will cause the output from the neural network to match the actual target values as closely as possible.

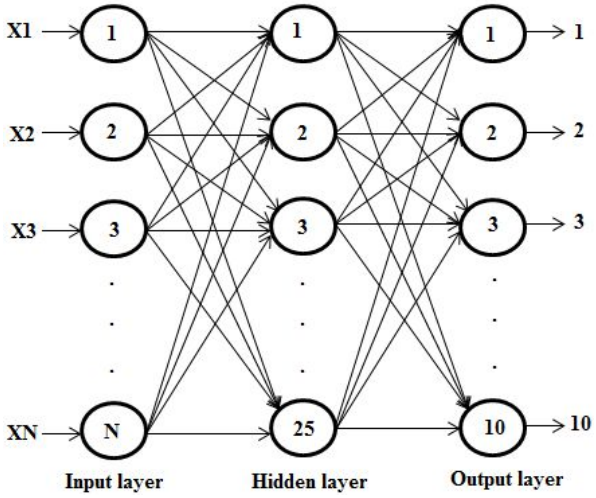

Figure.4 Neural network architecture

VII. EXPERIMENTS AND RESULTS

The present database of Gujarati handwritten numeral is collected from the persons of different age groups and with different educational background. Total database is consist of 14,000 images having 1400 samples of each numeral from zero to nine. Images are devided randomly in the ratio of 4:1 for the training and testing purpose. Out of 14000 sample images 11,200 samples are applied for training and 2800 samples are used for testing the performance of the classifier. The results obtained for the success rate of the different Gujarati numerals with neural network and Bayes classifiers are shown in the tables below. Table 1 (a), 1(c), 1(e), and 1(g) represent the results obtained from neural network classifier with 16x16, 8x8, 4x4 and 2x2 zoning methods. The overall accuracy achieved with different zoning method is 93.03%, 95.92%, 91.89% and 61.78% respectively.

Similarly, Table 1 (b), 1(d), 1(f), and 1(h) represent the results obtained from the Bayes classifier with 16x16, 8x8, 4x4 and 2x2 zoning methods. The overall accuracy achieved with different zoning method is 75%, 85.60%, 81% and 53.75% respectively.

VIII. CONCLUSION

The Goal of the proposed work is to identify handwritten Gujarati numerals. This paper introduces a numeral recognition scheme based on zone based feature extraction method. Numeral image is divided into 16x16, 8x8, 4x4 and 2x2 Zones. The proposed system uses Multi-layer feed forward neural network and Naive Bayes classifier for classification of handwritten Gujarati numerals. The overall accuracy achieved for 16x16, 8x8, 4x4, 2x2 zoning methods are 75%, 85.60%, 81% and 53.75% respectively using Naive Bayes classifier and 93.03%, 95.92%, 91.89% and 61.78% respectively using Multi layer feedforward neural network classifier. It is observed that the success rate of identification of 8x8 zoning is best as compared to other zoning methods.

Digit	০	১	২	৩	৪	৫	৬	৭	৮	৯
০	272	0	0	0	0	0	0	3	3	2
১	0	262	6	0	5	2	1	1	2	1
২	3	3	263	0	1	3	0	1	4	2
৩	0	0	0	265	0	1	8	5	1	0
৪	0	3	1	1	262	3	3	1	6	0
৫	4	1	5	0	4	260	2	0	2	2
৬	0	4	1	11	3	11	243	5	1	1
৭	9	0	0	13	1	0	4	249	3	1
৮	2	1	1	0	2	0	1	0	268	5
৯	2	0	5	0	1	0	3	0	10	259

Table 1 (a) 16x16 zoning method using neural network classifier, The accuracy achieved 93.03%.

Digit	০	১	২	৩	৪	৫	৬	৭	৮	৯
০	263	0	2	1	0	2	0	8	1	3
১	0	259	11	0	3	1	3	2	1	0
২	2	72	193	0	3	2	2	1	1	4
৩	5	0	0	253	0	3	9	10	0	0
৪	0	56	4	1	211	6	1	1	0	0
৫	0	21	6	9	3	233	3	2	0	3
৬	4	0	5	89	8	5	148	13	0	8
৭	89	0	1	47	0	3	22	118	0	0
৮	57	12	0	0	6	0	0	2	201	2
৯	6	1	9	1	4	0	1	10	5	243

Table1 (b) 16x16 zoning method using Bayes classifier, The accuracy achieved 75.00%.

Digit	০	১	২	৩	৪	৫	৬	৭	৮	৯
০	277	0	0	0	0	0	0	1	2	0
১	0	266	7	0	2	4	0	0	1	0
২	0	4	268	0	0	1	0	1	4	2
৩	1	1	0	267	1	0	3	6	0	1
৪	0	1	2	0	271	1	0	2	3	0
৫	2	0	2	0	3	269	1	0	0	3
৬	0	0	4	2	2	10	257	5	0	0
৭	1	0	1	7	1	0	3	266	0	1
৮	1	0	0	0	3	0	0	0	272	4
৯	0	0	1	0	2	1	2	0	3	271

Table 1 (c) 8x8 zoning method using neural network classifier, The accuracy achieved 95.92%.

Digit	০	১	২	৩	৪	৫	৬	৭	৮	৯
০	262	0	2	0	0	1	1	8	3	3
১	0	245	10	0	8	5	6	1	5	0
২	2	28	233	0	1	6	1	0	7	2
৩	0	0	0	231	0	4	22	23	0	0
৪	0	11	2	0	251	12	2	0	1	1
৫	0	11	10	0	7	249	3	0	0	0
৬	0	0	5	35	14	14	176	36	0	0
৭	8	0	0	13	1	2	17	236	0	3
৮	2	4	0	0	4	0	0	0	267	3
৯	5	2	11	2	4	3	1	3	4	245

Table 1 (d) 8x8 zoning method using Bayes classifier, The accuracy achieved 85.60%.

Digit	০	১	২	৩	৪	৫	৬	৭	৮	৯
০	273	0	1	1	0	0	0	2	3	0
১	0	267	6	1	1	2	1	0	1	1
২	0	6	260	2	1	0	2	3	1	5
৩	0	0	0	242	1	0	12	24	0	1
৪	0	3	2	1	265	1	2	0	4	2
৫	0	0	2	1	0	274	1	0	0	2
৬	1	2	2	17	2	7	218	30	0	1
৭	3	0	0	22	2	0	18	235	0	0
৮	1	0	0	0	1	0	0	0	277	1
৯	1	0	2	1	1	0	0	1	3	271

Table 1 (e) 4x4 zoning method using neural network classifier, The accuracy achieved 91.89%.

Digit	০	১	২	৩	৪	৫	৬	৭	৮	৯
০	254	0	1	0	0	0	1	9	2	13
১	0	226	5	0	17	2	13	0	10	7
২	0	32	220	0	3	12	5	0	6	2
৩	0	0	0	205	1	11	28	35	0	0
৪	0	10	1	5	255	4	5	0	0	0
৫	0	7	17	0	4	246	2	0	0	4
৬	0	0	5	69	27	30	99	44	0	6
৭	1	0	0	19	0	2	28	227	0	3
৮	0	0	4	0	3	1	0	0	268	4
৯	2	0	12	0	0	2	0	0	4	260

Table 1 (f) 4x4 zoning method using Bayes classifier, The accuracy achieved 81.00%.

Digit	০	১	২	৩	৪	৫	৬	৭	৮	৯
০	247	3	8	1	9	2	0	3	2	5
১	8	217	9	1	21	2	2	5	4	11
২	21	16	159	6	18	5	4	3	2	46
৩	12	1	7	122	4	14	55	64	0	1
৪	7	10	4	6	168	29	5	19	22	10
৫	2	2	5	4	4	246	11	0	0	6
৬	8	1	3	60	16	66	78	41	0	7
৭	15	2	5	73	9	10	21	145	0	0
৮	1	0	13	0	5	0	0	0	236	25
৯	9	6	27	0	14	2	0	0	32	190

Table 1 (g) 2x2 zoning method using neural network classifier, The accuracy achieved 61.78%.

Digit	০	১	২	৩	৪	৫	৬	৭	৮	৯
০	239	4	9	0	7	1	2	5	3	10
১	7	213	7	1	16	8	4	4	8	12
২	23	29	113	11	15	4	5	19	8	53
৩	8	1	0	118	0	21	29	101	0	2
৪	30	29	3	4	77	50	9	26	42	10
৫	4	8	5	2	8	219	28	3	0	3
৬	11	1	0	74	7	74	60	41	1	11
৭	19	2	0	43	5	23	21	165	0	2
৮	1	2	8	0	5	0	0	0	248	16
৯	7	11	14	0	8	1	0	1	57	181

Table 1 (h) 2x2 zoning method using Bayes classifier,
The accuracy achieved 53.75%

REFERENCES

[1] B. V. Dasarathy, "Nearest neighbor pattern classification techniques", IEEE Computer Society Press, New York, 1991.

[2] C.Y. Suen, C. Nadal, R. Legault, T.A.Mai, L. Lam, "Computer Recognition of unconstrained handwritten numerals", Proc. IEEE, 1992, Vol 80, pp. 1162-1180.

[3] S. Mori, C.Y. Suen, K. Yamamoto, "Historical Review of OCR research and development", Proc. IEEE, 1992, Vol. 80, pp. 1029-1058.

[4] A. Dutta and S. Chaudhury, "Bengali alpha-numeric Character recognition using curvature features", Patter Recognition, Vol. 26(12), pp. 1757-1770, 1993.

[5] Anil.K.Jain and Torfinn Taxt, "Feature Extraction Methods for Character Recognition-A Survey", *Pattern Recognition,* 1996*,* vol. 29, no. 4, pp. 641-662.

[6] L. L. Lee and N. R. Gomes, "Disconnected handwritten numeral image recognition", in the Proceedings of 4th ICDAR, pp. 467-470, 1997.

[7] L. Heutte, T. Paquet, J. V. Moreau, Y. Lecourtier and C. Olivier, "A structural/statistical feature based vector for handwritten character recognition", Pattern Recognition Letters, Vol. 19(7), pp. 629-641, 1998.

[8] Antani S. and Agnihotri L., "Gujarati Character Recognition", In Proc.Of 5th International Conference on Document Analysis and Recognition, IEEE Computer Society Press, pp. 418-421, 1999.

[9] U. Garain, B.B. Chaudhuri, Segmentation of touching characters in printed Devnagari and Bangla Scripts using fuzzy multifactorial analysis, IEEE Transactions on Systems, Man And Cybernetics, Part C 32 (4) (2002) 449–459.

[10] U. Bhattacharya, T. K. Das, A. Datta, S. K. Parui and B. Chaudhuri, "A hybrid scheme for hand printed numeral recognition based on a self-organizing network and MLP classifiers", International Journal on Pattern Recognition and Artificial Intelligence, Vol. 16(7), pp. 845-864, 2002

[11] V.L. Lajish, T.T.K. Suneesh and N.K. Narayanan,"Recognition of Isolated Handwritten Character Images using Kolmogorov-smirnov, Statistical Classifier and K-nearest Neighbour Classifier", Proc. of the International Conference on Cognition and Recognition, pp 526-531, 2005.

[12] K. Roy, T. Pal, U. Pal and F. Kimura, "Oriya handwritten numeral recognition system", Proceedings of ICDAR, pp. 770-774, 2005.

[13] S K Shah and A Sharma "Design and Implementation of Optical Character Recognition System to Recognize Gujarati Script using Template Matching", IE (I) Journal-ET, Vol. 86, pp. 44-49, 2006.

[14] N. Sharma, U. Pal, F. Kimura, and S. Pal P. Kalra and S.Peleg (Eds.), "Recognition of Off-Line Handwritten Devnagari Characters Using Quadratic Classifier", ICVGIP 2006, LNCS 4338, Springer-Verlag Berlin Heidelberg, pp. 805-816,2006 .

[15] A. Majumdar and B.B. Bhaudhuri, "Printed and Handwritten Bangla Numeral Recognition using Multiple Classifier Outputs", *Proceedings of the first IEEE ICSIP06, 2006*, Vol. 1, pp. 190-195.

[16] U. Pal, Sharma. N, Wakabayashi. T, Kimura. F, "Handwritten Numeral Recognition of Six Popular Indian Scripts", *Ninth International conference on Document Analysis and Recognition ICDAR07,* vol. 2,pp.749-753.

[17] Rajashekararadhya, S.V.; Ranjan, P.V., "Neural network based handwritten numeral recognition of Kannada and Telugu scripts", TENCON 2008, IEEE Region 10 Conference, pp .1 –5 Nov. 2008.

[18] Jignesh Dholakia, Atul Negi and S. Ram Mohan "Progress In Gujarati Document Analysis and Character Recognition", Guide to OCR for Indic Scripts: Advance in Pattern Recognition, Vol. 1, pp. 73-95, Springer Link 2010.

[19] Apurva A. Desai, "Gujarati handwritten numeral optical character reorganization through neural network", Pattern Recognition Volume 43, Issue 7, pp. 2582-2589, July 2010.

[20] Apurva A.Desai, "Handwritten Gujarati Numeral Optical Character Recognition using Hybrid Feature Extraction Technique", Proceedings of International Conference on Image processing, computer vision & pattern recognition, IPCV"10. pp. 733-739, 2010.

[21] Desai A. A., (2010) "Gujarati handwritten numeral optical character reorganization through neural network", Pattern Recognition, Vol. 43, pp 2582-2589.

[22] Desai A. A. (2010), Handwritten Gujarati Numeral Optical Character Recognition using Hybrid Feature Extraction Technique. IPCV pp 733-739.

[23] Anilkumar N. Holambe, Dr. Ravinder. C. Thoo l, "Printed and Handwritten Character &Number Recognition of Devanagari Script using SVM and KNN", International Journal of Recent Trends in Engineering and Technology, Vol.3, No.2, pp.163-166, 2010.

[24] D. Impedovo n, G.Pirlo "Zoning methods for hand written character recognition: Asurvey" Pattern Recognition 47(2014)969–981.

[25] Shailesh A. Chaudhari, Ravi M. Gulati, "An OCR for Separation and Identification of Mixed English- Gujarati Digits using k NN Classifier",2013 International Conference on Intelligent Systems and Signal Processing (ISSP).

[26] Baheti M.J, Kale K.V., "Gujarati Numeral Recognition: Affine Invariant Moments Approch",1st International Conference on Recent Trends in Engineering & Technology, Mar-2012, Special Issue of International Journal of electronics, Communication & Soft Computing Science & Engineering, ISSN: 2277-9477.

[27] Mamta maloo, K.V. Kale, "Support vector machine based Gujarati numeral recognition", International Journal on Computer Science and Engineering (IJCSE), ISSN: 0975-3397 Vol. 3 No. 7 July 2011.

[28] Avani R. Vasant, Sandeep R.Vasant, Dr. G.R.Kulkarni, "Performance Evaluation of Different Image Sizes for Recognizing Offline Handwritten Gujarati Digits using Neural Network Approach", 2012 International Conference on Communication Systems and Network Technologies.

Copy Move Forgery Detection using SIFT and GMM

Neetu Yadav and Rupal Kapdi

***Abstract*—Modifying or enhancing an image is ubiquitous but when enhancement tends to change the interpretation of the image they are termed as an attempt of forgery on digital images. Copy move forgery (CMF) is a simple technique and has a number of well built tools in a number of image enhancement software. CMF detection techniques often tend to establish similarity between copied and pasted region on the same image as both are from same original image. Keypoint and block based techniques are used to determine the CMF. SIFT keypoints are combined with different techniques to accurately localize forgery. High dimensionality of feature vector acts as a bottle neck in SIFT based analysis. We propose a method to detect CMF using SIFT descriptors which are clustered using GMM and segment the obtained suspect region speeding up the analysis.**

***Keywords*—Copy Move Forgery, Gaussian Mixture Model, Scale Invariant Feature Transform and Image Forensics.**

I. INTRODUCTION

IN CMF a part of image is taken from the source and pasted in another region on the target image where both the source and target image correspond to same image. It is often used to either hide anomaly in the image or add extra object into the image.

Image forensics encompasses both Image Tamper Detection (ITD) and Image Forgery Detection (IFD) [1]. Establishment of authenticity of the digital image is the only focus of ITD whereas IFD aims towards the localization of the forgery.

Passive techniques are often used to establish both the originality and authenticity of the digital image. As not all images consists authenticating details embedded in them active techniques like digital watermark consistency detection do not offer concrete results and hence when used in combination provide a rough estimate on the existence of a forgery.

Forgery localization can be done with maximum accuracy after establishment of forged image. Passive techniques comprises of pixel, format, geometrical, physical based methodologies that are used in heterogeneous combination to establish both originality and authenticity of the image [2]. The focus of this paper is to perform localization of forgery using pixel based passive forgery detection technique. In this paper we propose to use gaussian mixture model based clustering of the SIFT features.

II. RELATED WORK

Pixel based techniques for CMF detection can be performed using block based or keypoint based techniques or using both techniques in combination. Block based techniques for CMFD divide the image into loosely tiled blocks followed by feature extraction and matching. Often considered to be bulky and time consuming process [3]. Keypoint based techniques aim to locate the unique pixel points having features that are invariant to RST combined transformations. Keypoint are detected using various techniques like the SIFT, SURF, MIFT, ASIFT etc [4].

A variety of methods like the PCA [9], Zernike moment, DWT [11], have been combined with Keypoint methods to detect CMF [5]. In [7], authors have used a two stage process to detect CMF using the transformation matrix and using the keypoint matching techniques to detect similar matches. Cluster matching methodologies are much more advantageous as they simplify and speedup the similarity detection stage. In [8], the authors combined the SIFT features along with BFSN clustering method and the Color Filter Array feature to detect simultaneous multiple CMF regions.

To detect CMF forgeries robust to various post processing methods the authors in [18] used the MIFT features combined with the affine transformation estimation and used the RANSAC method to remove the false matches refining the algorithm to be robust against blurring, scaling , rotation, mirror reflections and deformation operations on the CMF regions.

III. PROPOSED METHOD

The key objective of the proposed method is to locate the suspect region for a CMF in the image and segment that region for further analysis. Since in copy move region same part of image is copied and pasted on the same image at a different location. The similarity present in the regions is exploited to detect this type of forgery [23]. The similarity detection is performed by extracting and matching local features from different regions of image. Figure 1 illustrates the main steps of the proposed approach.

Keypoint based techniques of CMF detection is robust against a variety of post processing operations [9]. Keypoint based CMF detection methods are robust against many post processing operations performed to evade the detection mechanism.

In the first stage the input test image is subjected to grayscale conversion after which the SIFT features are extracted from them. Since the number of SIFT features extracted are dependent on the image content the number of features vary fro few hundreds to thousands. Each keypoint in SIFT consists of 128 feature vectors. Direct matching of these features is computationally expensive due to high dimensionality of the feature vectors. Hence clustering the keypoints using the Gaussian mixture model aids in determining the similar features from the image.

Using Gaussian mixture model to perform soft clustering of keypoints helps in determining the similar keypoints. Once the list of similar keypoints generated then using the G2NN (generalized two nearest neighbor) method helps in determining the exact pairs of similar features. After

978-1-4799-9991-0/15/$31.00 ©2015 IEEE

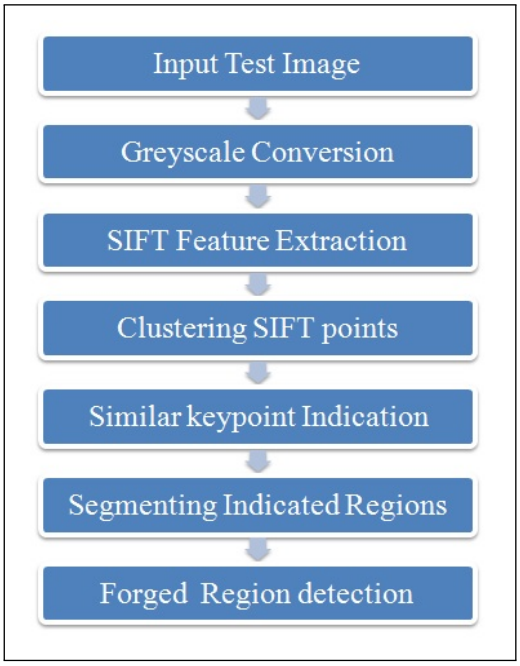

Fig. 1 Main steps of proposed methodology

the keypoint pairs with similar features are determined image segmentation is performed using the first pair of the keypoints.

GMM clustering is soft clustering method. The initial value for number of clusters is set to n, determined empirically. We have used the maximum prior probability of the cluster. After which sort the keypoints based on the covariance values and sort them into similar points behavior. Since covariance is the measure of how two random variables change.

To reduce keypoint set for analysis of similarity first half or an eighth of the keypoint set. Further for similarity analysis from the G2NN method setting the empirical threshold value set to t as we analyze the RST combine forgery. The similar points extracted after performing the G2NN method are plotted and suspect region detection is performed using laplacian filter and boundary trace operations.

A. Scale Invariant Feature Transform (SIFT)

SIFT features are invariant to most post processing operations like translation, rotation, scaling and combinations of other techniques like noise addition, color adjustment, brightness change, contrast adjustment. D. G. Lowe in [17] proposed a method to aid image retrieval application. The provides with a feature matrix with 128 valued vectors for each unique point that is detected along with their coordinates and orientation. The generated features give local image description and also help in visual correspondence between similar images regions. SIFT features are computed using scale space of an image represented as a convolution of the Gaussian and the image functions in the equation below.

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \qquad (1)$$

were scale space function is L(x,y,σ), G(x,y,σ) is variable scale Gaussian, $*$ refers to convolution operator and I(x, y) represents the input image. Further Gaussian function given below is used as scale space kernel because it is efficient to perform computation using smoothed images.

$$G(x, y, \sigma) = 1/(2\pi\sigma^2)e^{[} - (x^2 + y^2)/2\sigma^2] \qquad (2)$$

SIFT features are computed by performing Difference of Gaussian (DoG) to find local extrema so as to produce stable features. DoG is used because it provides close approximation to scale normalized Laplacian of Gaussian (LoG) function also maxima and minima produce stable feature compared to others.

SIFT algorithm has been modeled with a number of variations based on its application. A number of modifications to SIFT exist in which the algorithm is subjected to changes that enable generated features to be invariant to reflections (mirror) or affine changes (shear shifts) [4]. The wide range of usage of SIFT features can be due to their high invariance to scale and rotation combined with robustness to Gaussian noise (white) and other forms of occlusion, affine distortion, illumination and perspective shifts in digital image.

The operation of SIFT method can be accurately put into four small steps. These steps are scale space depiction of the image, keypoint computation with the DoG, contrast based edge filter and keypoint orientation computation before the final SIFT descriptors are formed. The SIFT keypoints can be thought as the points of discontinuity of gradient function taken from digital image regions using the DoG function.

B. GMM soft clustering of SIFT keypoints and Segmentation

Keypoint matching algorithm makes use of ratio of Euclidean distance between the closest neighbours and checking it against a global threshold. This method is incapable of detecting the multiple forgeries and results in high rate of false matches. Using the cluster matching method the false negatives [19]. GMM clustering methods often used in segmentation applications. Gaussian distribution most common and continuous distribution described by two parameters the mean and the variance. Gaussian have same shape wherein the location is controlled by the mean and spread controlled by variance. The probability distribution of d dimensional vector X is a multivariate gaussian that is parameterized by mean and covariance by $\sum$.

$$f_k(x) = \frac{1}{\sqrt{((2)^d|\sum k|)}} exp \frac{(-(-x-\mu_k)^T \sum k(-1)(X-\mu))}{2}$$

Applying GMM provides a collection of k gaussian and each distribution consists of prior probabilities and representing the cluster of data points [21]. In model based clustering, a model is hypothesized for each of clusters and the concept is to find the best fit of that model to each other. Mixture models have well applied statistical inferences and provide flexibility in choosing component distribution also provide density estimation for all clusters and soft classification aid in efficient clustering.

Clustering of SIFT keypoints using GMM helps in reducing the number of keypoints to be analysed for similarity detection that is performed using G2NN method. Generalized

two nearest neighbour (G2NN) method helps in detecting the similar keypoint pairs.G2NN method works by sorting the Euclidean distance between the descriptors and then computing the ratio between distance of closest neighbour to that of the second closest neighbour. After satisfying the threshold constraint similar points are separated out. The points thus detected are utilized in detecting and segmenting the suspect region for existence of copy move forgery (CMF).

IV. EXPERIMENTAL ASSEMBLE AND RESULTS

Implementation of the proposed CMF method is carried out in Matlab and using the open source VL-Feat library for computer vision as it provides necessary implementations for algorithms widely used in image processing and computer vision [19]. We have also used the CoMoFoD dataset consisting of RST combined post-processing to assess our CMF detection method. The CoMoFoD dataset aims towards setting up a benchmark for CMF detection algorithms [20].

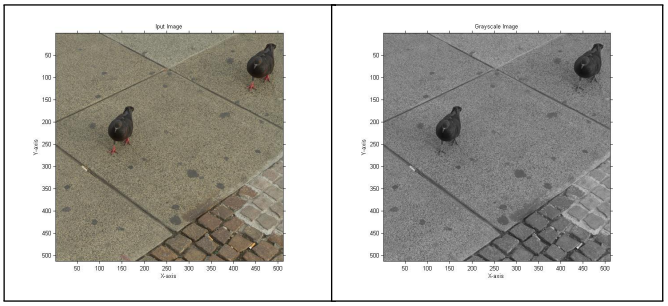

Fig. 2 Test Image from CoMoFoD dataset [20]

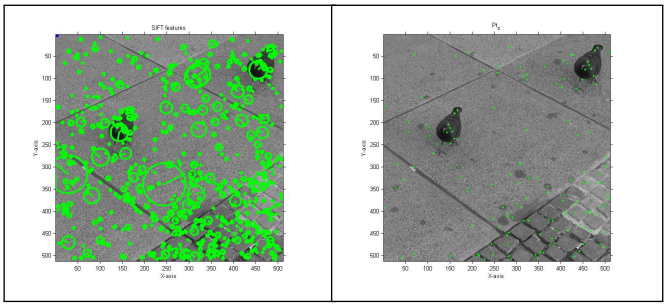

Fig. 3SIFT keypoints and GMM reduced keypoint set(n=6 and t=1.3)

Also we detected forgery our custom created images. The CMF modifications were performed using the GIMP tool. Image forgery detection is the main focus and the test images are confirmed to be tampered with by other algorithms.

The proposed method aims towards detection and localization of CMF suspect region. Some of the incorrect detections are depicted in the following images.

V. CONCLUSION

A progressive method to enable image forgery detection based on SIFT features has been proposed and implemented. Given a suspected test photo, it can reliably perform CMF

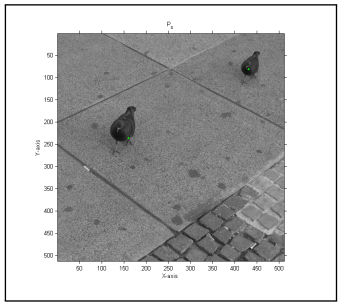

Fig. 4 Similar keypoint pair

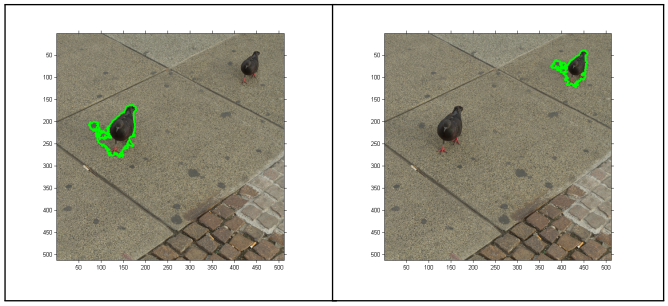

Fig. 5CMF Suspect region detection using our method

detection for duplicate regions. The presented technique can show effectiveness with respect to combined RST modifications to detect copy-move forged region in the image. The cluster formation phase has been extended by using image segmentation method. SIFT based image forgery detection techniques can further be extended to include the detection of other image forgery apart from copy-move. The limitation of the proposed method exists in improper detection of multiple forged regions and flat surface regions.

VI. FUTURE WORK

In future, the segmentation method can be extended using the object recognition operation enhancing the applicability of the algorithm for real time forgery detection. Combining deep learning concept with image forgery detection techniques can help detect improper presence of an object at a particular place in an image. Implementing the machine learning techniques to automate this process further enhances the applicability and deployment of algorithm for real time forgery detection.

REFERENCES

[1] W. Wang, J. Dong, and T. Tan, "A survey of passive image tampering detection," *in Digital Watermarking (A. Ho, Y. Shi, H. Kim, and M.Barni, eds.), vol. 5703 of Lecture Notes in Computer Science, pp. 308-322, Springer Berlin Heidelberg,* Feb. 2009.

[2] Hany Farid, "Image forgery detection- a survey," *in IEEE Signal Processing Magzine,* Feb. 2009.

[3] Kalpana Manudhane and Mr. M.M. Bartere, "Methodology for Evidence Reconstruction in Digital Image Forensics," *Computer Engineering and Intelligent Systems, ISSN 2222-1719 (Paper) ISSN 2222-2863 (Online), Vol.4, No.13,* 2013.

[4] Jian Wu, Zhiming Cui, Victor S. Sheng, Pengpeng Zhao, Dongliang Su and Shengrong Gong, "A Comparative Study of SIFT and its Variants," *Measurement Science Review, Volume 13, No. 3,* 2013.

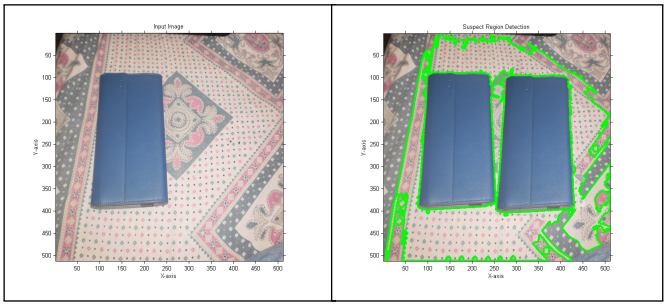

Fig. 6CMF suspect region detection (original and forged pair from custom images).

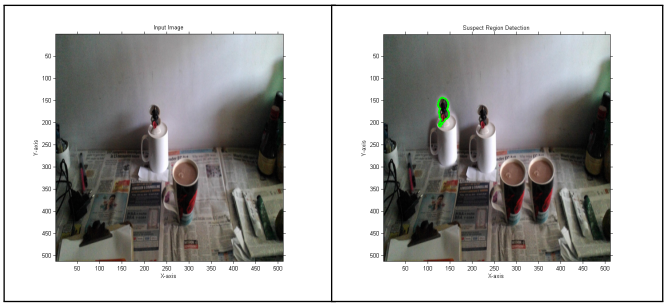

Fig. 7 Improper CMF detection (original and forged pair form custom images).

[5] B. L. Shivakumar and S. S. Baboo, "Detecting copy-move forgery in digital images: A survey and analysis of current methods," *Global Journal of Computer Science and Technology, vol. 10, no. 7, pp. 61-65,* 2011.

[6] Wiem Taktak,Jean-Luc Dugelay and Judith A. Redi, "Digital image forensics: A booklet for beginners," *in Eurecom, France*, 2009.

[7] Jian Li, Xiaolong Li, Bin Yang, and Xingming Sun, "Segmentation-Based Image Copy-Move Forgery Detection Scheme" *in IEEE Transactions On Information Forensics And Security,* Vol. 10, No. 3, March 2015.

[8] Lu Liu, Rongrong Ni, Yao Zhao and Siran Li, "Improved SIFT-based Copy-move Detection Using BFSN Clustering and CFA Features", *in 10th International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, IEEE,2014.

[9] Takwa Chihaoui, Sami Bourouis, and Kamel Hamrouni, "Copy-Move Image Forgery Detection Based On Sift Descriptors And Svd-Matching," *in 1st International Conference on Advanced Technologies*, March,2014.

[10] Ramesh Chand Pandey, Sanjay Kumar Singh, K. K. Shukla and Rishabh Agrawal, "Fast and Robust Passive Copy-Move Forgery Detection Using SURF and SIFT Image Features," *in Department of Computer Science Engineering Indian Institute of Technology (BHU)* , IEEE, 2014.

[11] Mohammad Farukh Hashmi, Aaditya R. Hambarde and Avinash G. Keskar, "Copy Move Forgery Detection using DWT and SIFT Features," *in Department of Electronics Engineering, Visvesvaraya National Institute of Technology* , Nagpur, India, IEEE,2013.

[12] Bo Liu and Chi-Man Pun, "A Sift And Local Features Based Integrated Method For Copy-Move Attack Detection In Digital Image" *IEEE international conference on Information and automation* ,August,2013.

[13] Irene Amerini, Lamberto Ballan, Roberto Caldelli, Alberto Del Bimbo, and Giuseppe Serra, "A Sift-Based Forensic Method For CopyMove Attack Detection And Transformation Recovery" *in IEEE Transactions On Information Forensics And Security*, Vol. 6, No. 3, September 2011.

[14] Xunyu Pan and Siwei Lyu, "Region Duplication Detection -Using Image Feature Matching" *in IEEE Transactions On Information Forensics And Security*, Vol. 5, No. 4, December 2010.

[15] M. Jaberi, G. Bebis, M. Hussain, and G. Muhammad, "Improving the detection and localization of duplicated regions in copy-move image forgery" *in Digital Signal Processing (DSP), 18th International Conference on, pp. 1-6,* , July 2013.

[16] V. Christlein, J. Jordan, C. Riess, and E. Angelopoulou, "An evaluation of popular copy-move forgery detection approaches" *Information Forensics and Security, IEEE Transactions on, vol. 7, pp. 1841-1854,* ,Dec 2012.

[17] David G.Lowe, "Distinctive image features from scale-invariant keypoints," Journal of Computer Vision, vol. 60, no. 2, pp. 91-110,, 2004.

[18] E. Ardizzone,A.Bruno and G. Mazzola, "Detecting Multiple Copies In Tampered Image" *IEEE 17th International Conference on Image Processing,* , 2010.

[19] A.Vedaladi and B.Fulkerson, "An open and portable library of computer vision algorithm," *http://www.vlfeat.org/api/sift.html,*2007.

[20] D. Tralic, I. Zupancic, S. Grgic, and M. Grgic, "Comofod x2014; new database for copy-move forgery detection," *in ELMAR, 2013 55th International Symposium, pp. 49-54,*,Sept 2013.

[21] Douglas Reynolds, "Gaussian Mixture Models," *MIT Lincoln Laboratory, 244 Wood St., Lexington, MA 02140, USA.*

[22] Ju zhang, Qiuqi ruan and Yi jin, "Combined Sift And Bi-Coherence Features To Detect Image Forgery," *in ICSP Proceedings,*IEEE ,2014.

[23] Rupal Amit Kapdi and Neetu Yadav, "Copy Move Forgery Detection Using SIFT Features- An Analysis," *Nirma University Journal of Engineering and Technology, North America,vol-4,*, Aug-2015.

Surveying Stock Market Portfolio Optimization Techniques

Mukesh Kumar Pareek
Computer Science and Engineering,
Inst. of Technology, Nirma University,
Ahmedabad, India
Email: 13mcec26@nirmauni.ac.in

Priyank Thakkar
Computer Science and Engineering,
Inst. of Technology, Nirma University,
Ahmedabad, India
Email: priyank.thakkar@nirmauni.ac.in

***Abstract*—Optimizing a stock market portfolio requires decision making at two distinct stages, first is to select the stocks and second is to assign distribution of investment amount among these selected stocks. Given historical data of stocks, role of optimization models is to select stocks and assign portfolio proportion to the selected stocks. Selection and weight assignment to stocks are co-occurring activities. Investors prime motive is to maximize the return and minimize the risk of portfolio. Stock market is uncertain and volatile and therefore Artificial Intelligence, Machine Learning and Soft Computing techniques are viable candidates which can help in optimization and making decisions using such data. This paper surveys research carried out in the domain of stock market portfolio optimization. Paper compares research efforts in the domain on the basis of techniques used, risk models and stock markets considered. It is observed from the surveyed papers that Artificial Intelligence, Machine Learning and Soft Computing techniques are widely accepted for studying and evaluating stock market behavior and optimizing portfolios.**

***Keywords*—Stock Market, Stock Market Portfolio Optimization, Risk Models, Stock Market Portfolio Optimization Techniques**

I. Introduction

Achieving high returns while limiting the risk to minimum possible value are the prime objectives of any investor when investing capital in the stock market. A portfolio is grouping of stocks in which the capital among stocks is invested in such a proportion that profit is maximum and risk is minimum. Markowitz [1] had proposed a mean variance model for optimizing a portfolio in 1952. This model can be used by investors to achieve desired returns from portfolio with minimum possible risk. If fact, this theory has wide spread acceptance and has been used as a practical tool for portfolio optimization.

However, certain characteristics of the problem, such as its size, requirements of the practical-world, investor's desire of certain constraints, very limited computation time etc. may make analytical method unsuitable. This forces researchers and practitioners to explore various heuristic and machine learning techniques that can deal with portfolio optimization problem with these requirements and constraints.

These techniques have been used extensively with different ways of risk modelling. Research efforts are not confined to few stock markets, rather, researchers have applied their knowledge to a large number of stock markets across the globe. This paper surveys on these different dimensions and presents a comparative study.

II. Portfolio Optimization Models

Markowitz [1] had proposed a mean-variance model for portfolio optimization in which weighted mean returns of the stocks in portfolio were considered as a return of the portfolio and variance of these stocks from mean return was considered as a risk. Markowitz model can be described using Equation 1, Equation 2, Equation 3 and Equation 4.

$$Minimize \quad \sum_{i=1}^{n}\sum_{j=1}^{n} w_i w_j \sigma_{ij} \tag{1}$$

$$such\ that \quad \sum_{i=1}^{n} w_i \mu_i >= \mu_p \tag{2}$$

$$\sum_{i=1}^{n} w_i = 1 \tag{3}$$

$$0 \leq w_i \leq 1; i = 1...n \tag{4}$$

where,
n = number of stocks in dataset,
μ_i = the expected return of the asset i,
σ_{ij} = the co-variance between asset i and j,
w_i = proportion of capital invested in asset i,
μ_p = desired return from portfolio

Equation 1 represents the objective function of the optimization problem which aims to minimize risk of portfolio while Equation 2 enforces the portfolio to achieve desired return μ_p. Equation 3 and Equation 4 are constraints on portfolio, assuring that 100% of the investors' capital is invested and no short selling is performed, respectively.

Markowitz model has been extensively used for portfolio optimization. This model uses historical mean return and co-variance of stocks to optimize a portfolio. Markowitz model selects stocks which have minimum co-variance between them to ensure diversified risk, i.e., to minimize chances of loss. It is easy to understand from statistics that low co-variance stocks do not move together, so if some stocks in portfolio are not

978-1-4799-9991-0/15/$31.00

performing well, then other stocks (having low co-variance with poorly performing stocks) in portfolio can cover the loss. Adding cardinality constraint to Markowitz model turns the model form a QP problem to a MIQP (Mixed Integer Quadratic Programming) problem which is an NP-Hard problem. Other constraints like, sector capitalization, minimum transaction lots, etc. make the problem even harder to solve.

MAD (Mean Absolute Deviation) model proposed by Konno and Yamazaki [2] is another popular model which is frequently used to solve portfolio selection problem. It solves the problem through linear programming. It is to be noticed that mean in MAD refers to the mean return of the assets in portfolio.

Some of the other models used include Exponential Decay Model [3], Extended Markowitz Model [4], [5], Mean-Variance-Skewness Model [6] and Robust MAD Model [7].

III. Portfolio Optimization Problem as a Quadratic Programming Problem and Heuristics for Portfolio Optimization

Basic Markowitz model can easily be implemented as a Quadratic Programming problem. Equation 5, Equation 6 and Equation 7 represents Markowitz model as a QP problem.

$$Minimize \qquad \frac{1}{2} w^T H w - m^T w \tag{5}$$

$$subject\ to \qquad e^T w = 1 \tag{6}$$

$$and \qquad 0 \leq w \leq 1 \tag{7}$$

where,
w = weight vector of all stocks,
H = co-variance matrix of mean return of all stocks,
e = vector of ones,
m = mean return vector of all stocks.

Above formulation restricts short selling, however, portfolio optimization problem can be formulated considering short selling as well. Further, if the cardinality constraint is added to the (quadratic programming) model presented above then the problem becomes **Mixed Integer Quadratic Programming** problem, which is NP-Hard and considerably more difficult to solve than the original problem. Instead of solving NP-Hard optimization problem, researchers have proposed various heuristics approaches to get the near optimal results for portfolio optimization. These heuristic approaches are implemented through various Artificial Intelligence and Soft Computing techniques. Different techniques that are widely used by researchers are mentioned in the next section.

IV. A Comparative Study

This section compares literature in the domain of stock market portfolio optimization across three dimensions: techniques used, risk models and stock markets considered.

A. Techniques Used

Survey of the existing literature reveals that Genetic Algorithm, Fuzzy Theory and Particle Swarm Optimization are extensively used techniques for portfolio optimization. Other techniques that are used frequently for optimizing the portfolio include multi-objective evolutionary algorithms (MOEA). NSGA-II (Nondominated Sorting Genetic Algorithm II), SPEA-2 (Strength Pareto Evolutionary Algorithm-2), PESA-II (Pareto Envelope Based Selection-II) and PAES (Pareto Archived Evolution Strategy) are some of the multi-objective evolutionary algorithms that have been used. Table I summarizes various techniques used for portfolio optimization problem by the researchers around the globe.

Abbreviations used in Table I are enlisted below:

AES = Adaptive Exponential Smoothing,

AHP = Analytical Hierarchy Process,

ARIMA = Autoregressive Integrated Moving Average,

ARM = Association Rule Mining,

ARMS = Autoregressive Markov-Switching Model,

ARX = Autoregressive Exogenous,

EA = Evolutionary Algorithm,

ES = Exponential Smoothing,

FT = Fuzzy Theory,

GA = Genetic Algorithm,

GBM = Geometric Brownian Motion,

GMM = Generalized Method of Moments,

MOEA = Multi-Objective Evolutionary Algorithm,

MPM = Minmax Probability Machine,

NCP = Nadir Compromising Programming,

NSGA-II = Nondominated Sorting Genetic Algorithm II,

PESA-II = Pareto Envelope Based Selection,

PQP = Parametric Quadratic Programming,

PSO = Particle Swarm Optimization,

RBF = Radial Basis Function,

RS = Rough Set,

SA = Simulated Annealing,

TABLE I. TECHNIQUES USED FOR PORTFOLIO OPTIMIZATION BY DIFFERENT RESEARCHERS

Paper	GA	FT	PSO	Others
[4]	Y	-	-	TS, SA
[5]	Y	-	-	ANN, TS, SA
[6]	-	-	-	Lagrange multiplier theory in optimization, RBF ANN, ARIMA, AES
[8]	-	-	-	ARX
[9]	-	-	-	ARX, RS, GS
[10]	Y	-	-	-
[11]	Y	-	-	-
[12]	Y	-	-	-
[13]	Y	-	-	-
[14]	-	-	Y	-
[15]	Y	-	-	SMO
[16]	-	Y	-	RS, GS
[17]	-	Y	-	-
[18]	-	-	-	EA
[19]	-	-	-	ARX
[20]	-	Y	-	AHP
[21]	-	-	Y	-
[22]	-	-	-	NCP
[23]	-	-	Y	-
[24]	-	-	Y	-
[25]	-	-	-	Kernel Method
[26]	-	Y	-	-
[27]	-	Y	Y	Monte Carlo
[28]	-	Y	-	-
[29]	Y	Y	-	-
[30]	Y	Y	-	-
[31]	Y	-	-	-
[32]	-	Y	-	-
[33]	-	Y	-	Clustering, SOM
[34]	Y	-	-	-
[35]	-	Y	-	ARM
[36]	Y	-	-	SVR
[37]	Y	-	-	-
[38]	-	Y	-	-
[39]	Y	-	-	Hybrid of GA, SA
[40]	-	-	-	DEA
[41]	-	Y	-	-
[42]	Y	Y	-	-
[43]	-	-	Y	-
[44]	Y	-	-	-
[45]	Y	-	-	-
[46]	-	Y	-	MOEA
[47]	Y	-	-	-
[48]	Y	-	-	SA
[49]	Y	-	-	-
[50]	Y	-	-	-
[51]	-	-	-	MPM, SVM
[52]	-	Y	-	-
[53]	Y	-	-	-
[54]	-	-	-	XCS
[55]	-	-	-	ARM, K-means clustering
[56]	-	-	-	ARM
[57]	-	-	-	EA
[58]	-	-	-	ARM, K-means clustering
[59]	-	-	-	eTrend, XCS
[60]	- -	- -	- -	Bayesian forecasting Latent Threshold Dynamic Models
[61]	-	-	-	GMM
[62]	-	-	-	PQP
[63]	-	-		NSGA-II, PESA, SPEA-2
[64]	-	-	-	ARIMA, ES, TSD
[65]	-	-	-	ARMS, GBM
[66]	-	-	-	ANN

SMO = Sequential Minimal Optimization,

SOM = Self Organising Maps,

SPEA-2 = Strength Pareto Evolutionary Algorithm,

SVM = Support Vector Machine,

SVR = Support Vector Regression,

TS = Tabu Search,

TSD = Time Series Decomposition,

XCS = Extended Classifier System,

Y = Yes.

B. Stock Markets Considered

Table II reflects the fact that portfolio optimization problem is attempted on the stocks belonging to various stock markets and efforts are not limited to only few stock markets.

TABLE II. DIFFRENT STOCK MARKETS ON WHICH STOCK MARKET PORTFOLIO OPTIMIZATION IS STUDIED

Stock Market	Paper
Korea Stock Exchange	[49], [50]
New York Stock Exchange	[53], [30]
Taiwan Stock Exchange	[55], [56], [58], [3], [66], [17], [47]
Shanghai Stock Exchange	[58], [59], [29], [38], [39], [41]
Shenzhen Stock Exchange	[58]
Hongkong Stock Exchange	[58]
Istanbul Stock Exchange	[64], [26]
Brazilian Stock Exchange	[8]
Tokyo Stock Exchange	[45], [13]
National Stock Exchange, India	[20], [35]
Iran Stock Exchange	[22]
Tehran Stock Exchange	[32]
Bombay Stock Exchange	[33], [35], [45]
Spanish Stock Exchange	[42], [46]

This clearly indicates the interest of the researchers across the globe to address the problem.

C. Risk Models

Table III depicts different parameters which are considered to model risk.

TABLE III. DIFFRENT RISK MODELS CONSIDERED

Risk Models	Paper
Variance	[10], [12], [67], [14], [68], [24], [32], [33] [37], [48], [60], [4], [5], [69], [6], [63]
Semi-variance	[12]
Mean absolute deviation	[12], [17], [30]
Variance with skewness	[12]
Microeconomic risk	[20]
Possibilistic absolute deviation	[41]
Possibilistic mean variance	[16]

It is evident that most of the researchers have used variance to model the risk, however, there are a few attempts where other measures have also been used to model the risk.

V. Conclusion

This paper has provided a brief account of the literature present in the domain of stock market portfolio optimization. Existing literature has been compared on the basis of optimization techniques used, risk models considered and stock markets on which this kind of study is undertaken. Existing literature points towards many possible future directions.

One of the possibility is to consider fundamental and technical indicators systematically. One can work on identifying most suitable fusion of these fundamental and technical indicators. Impact of extra economical aspects such as risk aversion or transaction costs can also be studied. It is also found that many approaches which have been proposed involve lots of computational effort. There is always a scope to improve on this aspect to make the proposals practical and easily usable.

It has already been established that growth stocks exhibit an under-reacting phenomenon while value stocks exhibit a significantly overreacting phenomenon [3]. One can study the strength and weakness of these phenomena:
for emerging or developed markets,
in Bullish and/or Bearish market,
in small-size and/or large-size companies,
on other value factor, e.g., earnings-to-price ratio, and other growth factor, e.g., return on asset.

References

[1] H. Markowitz, "Portfolio selection," *The Journal of Finance*, vol. 7, no. 1, pp. 77–91, 1952.

[2] H. Konno and H. Yamazaki, "Mean-absolute deviation portfolio optimization model and its applications to tokyo stock market," *Management Science*, vol. 37, no. 5, pp. 519–531, 1991.

[3] I.-C. Yeh and T.-K. Hsu, "Exploring the dynamic model of the returns from value stocks and growth stocks using time series mining," *Expert Systems with Applications*, vol. 41, no. 17, pp. 7730 – 7743, 2014.

[4] T.-J. Chang, N. Meade, J. Beasley, and Y. Sharaiha, "Heuristics for cardinality constrained portfolio optimisation," *Computers & Operations Research*, vol. 27, no. 13, pp. 1271 – 1302, 2000.

[5] A. Fernandez and S. Gomez, "Portfolio selection using neural networks," *Computers & Operations Research*, vol. 34, no. 4, pp. 1177 – 1191, 2007.

[6] L. Yu, S. Wang, and K. K. Lai, "Neural network-based mean-variance-skewness model for portfolio selection," *Computers & Operations Research*, vol. 35, no. 1, pp. 34 – 46, 2008. Part Special Issue: Applications of {OR} in Finance.

[7] Y. Moon and T. Yao, "A robust mean absolute deviation model for portfolio optimization," *Computers & Operations Research*, vol. 38, no. 9, pp. 1251 – 1258, 2011.

[8] D. Pinto, J. Monteiro, and E. Nakao, "An approach to portfolio selection using an arx predictor for securities risk and return," *Expert Systems with Applications*, vol. 38, no. 12, pp. 15009 – 15013, 2011.

[9] K. Y. Huang and C.-J. Jane, "A hybrid model for stock market forecasting and portfolio selection based on arx, grey system and rs theories," *Expert Systems with Applications*, vol. 36, no. 3, Part 1, pp. 5387 – 5392, 2009.

[10] H. Soleimani, H. R. Golmakani, and M. H. Salimi, "Markowitz-based portfolio selection with minimum transaction lots, cardinality constraints and regarding sector capitalization using genetic algorithm," *Expert Systems with Applications*, vol. 36, no. 3, Part 1, pp. 5058 – 5063, 2009.

[11] J.-S. Chen, J.-L. Hou, S.-M. Wu, and Y.-W. Chang-Chien, "Constructing investment strategy portfolios by combination genetic algorithms," *Expert Systems with Applications*, vol. 36, no. 2, Part 2, pp. 3824 – 3828, 2009.

[12] T.-J. Chang, S.-C. Yang, and K.-J. Chang, "Portfolio optimization problems in different risk measures using genetic algorithm," *Expert Systems with Applications*, vol. 36, no. 7, pp. 10529 – 10537, 2009.

[13] Y. Chen, S. Mabu, and K. Hirasawa, "Genetic relation algorithm with guided mutation for the large-scale portfolio optimization," *Expert Systems with Applications*, vol. 38, no. 4, pp. 3353 – 3363, 2011.

[14] G.-F. Deng, W.-T. Lin, and C.-C. Lo, "Markowitz-based portfolio selection with cardinality constraints using improved particle swarm optimization," *Expert Systems with Applications*, vol. 39, no. 4, pp. 4558 – 4566, 2012.

[15] W.-C. Tsai and A.-P. Chen, "Using the xcs classifier system for portfolio allocation of msci index component stocks," *Expert Systems with Applications*, vol. 38, no. 1, pp. 151 – 154, 2011.

[16] X. Zhang, W.-G. Zhang, and W.-J. Xu, "An optimization model of the portfolio adjusting problem with fuzzy return and a smo algorithm," *Expert Systems with Applications*, vol. 38, no. 4, pp. 3069 – 3074, 2011.

[17] S.-T. Liu, "A fuzzy modeling for fuzzy portfolio optimization," *Expert Systems with Applications*, vol. 38, no. 11, pp. 13803 – 13809, 2011.

[18] S. Garcia, D. Quintana, I. M. Galvan, and P. Isasi, "Time-stamped resampling for robust evolutionary portfolio optimization," *Expert Systems with Applications*, vol. 39, no. 12, pp. 10722 – 10730, 2012.

[19] K. Y. Huang, "Application of vprs model with enhanced threshold parameter selection mechanism to automatic stock market forecasting and portfolio selection," *Expert Systems with Applications*, vol. 36, no. 9, pp. 11652 – 11661, 2009.

[20] P. Gupta, M. Inuiguchi, and M. K. Mehlawat, "A hybrid approach for constructing suitable and optimal portfolios," *Expert Systems with Applications*, vol. 38, no. 5, pp. 5620 – 5632, 2011.

[21] J. Sun, W. Fang, X. Wu, C.-H. Lai, and W. Xu, "Solving the multi-stage portfolio optimization problem with a novel particle swarm optimization," *Expert Systems with Applications*, vol. 38, no. 6, pp. 6727 – 6735, 2011.

[22] M. Amiri, M. Ekhtiari, and M. Yazdani, "Nadir compromise programming: A model for optimization of multi-objective portfolio problem," *Expert Systems with Applications*, vol. 38, no. 6, pp. 7222 – 7226, 2011.

[23] H. R. Golmakani and M. Fazel, "Constrained portfolio selection using particle swarm optimization," *Expert Systems with Applications*, vol. 38, no. 7, pp. 8327 – 8335, 2011.

[24] H. Zhu, Y. Wang, K. Wang, and Y. Chen, "Particle swarm optimization (pso) for the constrained portfolio optimization problem," *Expert Systems with Applications*, vol. 38, no. 8, pp. 10161 – 10169, 2011.

[25] Y. Takano and J. ya Gotoh, "Multi-period portfolio selection using kernel-based control policy with dimensionality reduction," *Expert Systems with Applications*, vol. 41, no. 8, pp. 3901 – 3914, 2014.

[26] M. G. Yunusoglu and H. Selim, "A fuzzy rule based expert system for stock evaluation and portfolio construction: An application to istanbul stock exchange," *Expert Systems with Applications*, vol. 40, no. 3, pp. 908 – 920, 2013. FUZZYSS11: 2nd International Fuzzy Systems Symposium 17-18 November 2011, Ankara, Turkey.

[27] Y. Liu, X. Wu, and F. Hao, "A new chance-variance optimization criterion for portfolio selection in uncertain decision systems," *Expert Systems with Applications*, vol. 39, no. 7, pp. 6514 – 6526, 2012.

[28] W. Chen, S. Tan, and D. Yang, "Worst-case var and robust portfolio optimization with interval random uncertainty set," *Expert Systems with Applications*, vol. 38, no. 1, pp. 64 – 70, 2011.

[29] T. Magoc and F. Modave, "The optimality of non-additive approaches for portfolio selection," *Expert Systems with Applications*, vol. 38, no. 10, pp. 12967 – 12973, 2011.

[30] H. Dastkhan, N. S. Gharneh, and H. Golmakani, "A linguistic-based portfolio selection model using weighted max-min operator and hybrid genetic algorithm," *Expert Systems with Applications*, vol. 38, no. 9, pp. 11735 – 11743, 2011.

[31] Y. Chen, E. Ohkawa, S. Mabu, K. Shimada, and K. Hirasawa, "A portfolio optimization model using genetic network programming with control nodes," *Expert Systems with Applications*, vol. 36, no. 7, pp. 10735 – 10745, 2009.

[32] M. Fasanghari and G. A. Montazer, "Design and implementation of fuzzy expert system for tehran stock exchange portfolio recommendation," *Expert Systems with Applications*, vol. 37, no. 9, pp. 6138 – 6147, 2010.

[33] S. Nanda, B. Mahanty, and M. Tiwari, "Clustering indian stock market data for portfolio management," *Expert Systems with Applications*, vol. 37, no. 12, pp. 8793 – 8798, 2010.

[34] A. Gorgulho, R. Neves, and N. Horta, "Applying a ga kernel on optimizing technical analysis rules for stock picking and portfolio composition," *Expert Systems with Applications*, vol. 38, no. 11, pp. 14072 – 14085, 2011.

[35] P. Paranjape-Voditel and U. Deshpande, "A stock market portfolio recommender system based on association rule mining," *Applied Soft Computing*, vol. 13, no. 2, pp. 1055 – 1063, 2013.

[36] C.-F. Huang, "A hybrid stock selection model using genetic algorithms and support vector regression," *Applied Soft Computing*, vol. 12, no. 2, pp. 807 – 818, 2012.

[37] K. Lwin, R. Qu, and G. Kendall, "A learning-guided multi-objective evolutionary algorithm for constrained portfolio optimization," *Applied Soft Computing*, vol. 24, no. 0, pp. 757 – 772, 2014.

[38] X. Li, B. Shou, and Z. Qin, "An expected regret minimization portfolio selection model," *European Journal of Operational Research*, vol. 218, no. 2, pp. 484 – 492, 2012.

[39] W.-G. Zhang, Y.-J. Liu, and W.-J. Xu, "A possibilistic mean-semivariance-entropy model for multi-period portfolio selection with transaction costs," *European Journal of Operational Research*, vol. 222, no. 2, pp. 341 – 349, 2012.

[40] S. Lim, K. W. Oh, and J. Zhu, "Use of dea cross-efficiency evaluation in portfolio selection: An application to korean stock market," *European Journal of Operational Research*, vol. 236, no. 1, pp. 361 – 368, 2014.

[41] P. Zhang and W.-G. Zhang, "Multiperiod mean absolute deviation fuzzy portfolio selection model with risk control and cardinality constraints," *Fuzzy Sets and Systems*, vol. 255, no. 0, pp. 74 – 91, 2014. Theme: Decision and Optimisation.

[42] J. Bermudez, J. Segura, and E. Vercher, "A multi-objective genetic algorithm for cardinality constrained fuzzy portfolio selection," *Fuzzy Sets and Systems*, vol. 188, no. 1, pp. 16 – 26, 2012. Theme: Decision and Optimisation.

[43] J.-F. Chang and P. Shi, "Using investment satisfaction capability index based particle swarm optimization to construct a stock portfolio," *Information Sciences*, vol. 181, no. 14, pp. 2989 – 2999, 2011.

[44] Y. Chen, S. Mabu, and K. Hirasawa, "A model of portfolio optimization using time adapting genetic network programming," *Computers & Operations Research*, vol. 37, no. 10, pp. 1697 – 1707, 2010.

[45] G. Vijayalakshmi Pai and T. Michel, "Evolutionary optimization of constrained k -means clustered assets for diversification in small portfolios," *Evolutionary Computation, IEEE Transactions on*, vol. 13, pp. 1030–1053, Oct 2009.

[46] E. Vercher and J. Bermudez, "A possibilistic mean-downside risk-skewness model for efficient portfolio selection," *Fuzzy Systems, IEEE Transactions on*, vol. 21, pp. 585–595, June 2013.

[47] P.-C. Lin and P.-C. Ko, "Portfolio value-at-risk forecasting with ga-based extreme value theory," *Expert Systems with Applications*, vol. 36, no. 2, Part 1, pp. 2503 – 2512, 2009.

[48] M. Jahan and M. Akbarzadeh-Totonchi, "From local search to global conclusions: Migrating spin glass-based distributed portfolio selection," *Evolutionary Computation, IEEE Transactions on*, vol. 14, pp. 591–601, Aug 2010.

[49] K. J. Oh, T. Y. Kim, and S. Min, "Using genetic algorithm to support portfolio optimization for index fund management," *Expert Systems with Applications*, vol. 28, no. 2, pp. 371 – 379, 2005.

[50] K. J. Oh, T. Y. Kim, S.-H. Min, and H. Y. Lee, "Portfolio algorithm based on portfolio beta using genetic algorithm," *Expert Systems with Applications*, vol. 30, no. 3, pp. 527 – 534, 2006. Intelligent Information Systems for Financial Engineering.

[51] H. Ince and T. B. Trafalis, "Kernel methods for short-term portfolio management," *Expert Systems with Applications*, vol. 30, no. 3, pp. 535 – 542, 2006. Intelligent Information Systems for Financial Engineering.

[52] H. Katagiri, T. Uno, K. Kato, H. Tsuda, and H. Tsubaki, "Random fuzzy multi-objective linear programming: Optimization of possibilistic value at risk (pvar)," *Expert Systems with Applications*, vol. 40, no. 2, pp. 563 – 574, 2013.

[53] J.-S. Chen, C.-L. Chang, J.-L. Hou, and Y.-T. Lin, "Dynamic proportion portfolio insurance using genetic programming with principal component analysis," *Expert Systems with Applications*, vol. 35, pp. 273 – 278, 2008.

[54] W.-C. Tsai and A.-P. Chen, "Strategy of global asset allocation using extended classifier system," *Expert Systems with Applications*, vol. 37, no. 9, pp. 6611 – 6617, 2010.

[55] S.-H. Liao, H. hui Ho, and H. wen Lin, "Mining stock category association and cluster on taiwan stock market," *Expert Systems with Applications*, vol. 35, pp. 19 – 29, 2008.

[56] S. hsien Liao, P. hui Chu, and Y. lu You, "Mining the co-movement between foreign exchange rates and category stock indexes in the taiwan financial capital market," *Expert Systems with Applications*, vol. 38, no. 4, pp. 4608 – 4617, 2011.

[57] T.-Y. Yu, H.-C. Huang, C.-L. Chen, and Q.-T. Lin, "Generating effective defined-contribution pension plan using simulation optimization approach," *Expert Systems with Applications*, vol. 39, no. 3, pp. 2684 – 2689, 2012.

[58] S.-H. Liao and S.-Y. Chou, "Data mining investigation of co-movements on the taiwan and china stock markets for future investment portfolio," *Expert Systems with Applications*, vol. 40, no. 5, pp. 1542 – 1554, 2013.

[59] Y. Hu, B. Feng, X. Zhang, E. Ngai, and M. Liu, "Stock trading rule discovery with an evolutionary trend following model," *Expert Systems with Applications*, vol. 42, no. 1, pp. 212 – 222, 2015.

[60] X. Zhou, J. Nakajima, and M. West, "Bayesian forecasting and portfolio decisions using dynamic dependent sparse factor models," *International Journal of Forecasting*, vol. 30, no. 4, pp. 963 – 980, 2014.

[61] M. A. Ayadi and L. Kryzanowski, "Portfolio performance sensitivity for various asset-pricing kernels," *Computers & Operations Research*, vol. 35, no. 1, pp. 171 – 185, 2008. Part Special Issue: Applications of {OR} in Finance.

[62] M. Stein, J. Branke, and H. Schmeck, "Efficient implementation of an active set algorithm for large-scale portfolio selection," *Computers & Operations Research*, vol. 35, no. 12, pp. 3945 – 3961, 2008. Part Special Issue: Telecommunications Network Engineering.

[63] K. Anagnostopoulos and G. Mamanis, "A portfolio optimization model with three objectives and discrete variables," *Computers & Operations Research*, vol. 37, no. 7, pp. 1285 – 1297, 2010. Algorithmic and Computational Methods in Retrial Queues.

[64] O. Ustun and R. Kasimbeyli, "Combined forecasts in portfolio optimization: A generalized approach," *Computers & Operations Research*, vol. 39, no. 4, pp. 805 – 819, 2012. Special Issue on Operational Research in Risk Management.

[65] P. Aigner, G. Beyschlag, T. Friederich, M. Kalepky, and R. Zagst, "Modeling and managing portfolios including listed private equity," *Computers & Operations Research*, vol. 39, no. 4, pp. 753 – 764, 2012. Special Issue on Operational Research in Risk Management.

[66] P.-C. Ko and P.-C. Lin, "Resource allocation neural network in portfolio selection," *Expert Systems with Applications*, vol. 35, no. 1-2, pp. 330 – 337, 2008.

[67] K. Anagnostopoulos and G. Mamanis, "The mean-variance cardinality constrained portfolio optimization problem: An experimental evaluation of five multiobjective evolutionary algorithms," *Expert Systems with Applications*, vol. 38, no. 11, pp. 14208 – 14217, 2011.

[68] X. Huang, "Mean-variance models for portfolio selection subject to experts estimations," *Expert Systems with Applications*, vol. 39, no. 5, pp. 5887 – 5893, 2012.

[69] D. Bertsimas and D. Pachamanova, "Robust multiperiod portfolio management in the presence of transaction costs," *Computers & Operations Research*, vol. 35, no. 1, pp. 3 – 17, 2008. Part Special Issue: Applications of {OR} in Finance.

Hierarchical Clustering Technique for Word Sense Disambiguation using Hindi WordNet

Nirali Patel, Bhargesh Patel, Rajvi Parikh
IT Department
GCET
Vidhyanagar, Anand
niralipatel119211@gmail.com, bhargeshpatel@gcet.ac.in,
rajviparikh@gcet.ac.in

Dr. Brijesh Bhatt
Computer Department
DDIT
Nadiad, India
brij.s.bhatt@gmail.com

Abstract— Word Sense Disambiguation (WSD) is crucial and its significance is prominent in every application of computational linguistics. WSD is a challenging problem of Natural Language Processing (NLP). Though there are lots of algorithms for WSD available, still little work is carried out for choosing optimal algorithm for that. Three approaches are available for WSD, namely, Knowledge-based approach, Supervised approach and Unsupervised approach. Also, one can use the combination of given approaches. Supervised approach needs large amounts of manually created sense-annotated corpus which takes computationally more amount of time and effort. Knowledge-based approach requires machine readable dictionaries, sense inventories, thesauri, etc, which are dependent on own interpretation about word's sense; Whereas unsupervised approach uses sense-unannotated corpus and it is based on the phenomenon of working that words that co-occur have similarity. This research is for Hindi language which uses Hierarchical clustering algorithm with different similarity measures which are cosine, Jaccard and dice, the result of clusters is overlapped with Hindi WordNet a product of IIT Bombay which improves result of word sense disambiguation as clustering does grouping of words which are similar.

***Keywords*—Word Sense Disambiguation; Natural Language Processing; Hindi WordNet; Hierarchical Clustering; Similarity Measure**

I. INTRODUCTION

Natural Language Processing (NLP) is getting more interest as part of Information Technology [2]. There is a necessity to measure performance of Natural Language Processing to computer processing, such that computer can understand human language and human can interact with computer in their own language, in addition to English. There are many areas to research in NLP, like Part-of-Speech tagger, Stemming, Information Retrieval, Word Sense Disambiguation, Machine Translation, Question-answering etc. In a Natural Language Processing, the task of Word Sense Disambiguation (WSD) is assigning appropriate meaning or we can say sense to given word in text [3]. Word Sense Disambiguation is mainly concerned with giving meaning to words according to their context in which they occur. For example, the sentence in Hindi सुन्दर का घर सुन्दर है in which the word सुन्दर occurs two times and the meaning for each is different. First one is a name of a boy; whereas second one describes how beautiful the house of that boy is. Humans can easily recognize this but computer cannot. So, WSD can help here to understand the context of word.

Word Sense Disambiguation has mainly three approaches, namely, Knowledge-based, Supervised approach and Unsupervised approach and also one can do the combination of these approaches [3]. In every approach, there are many techniques available. Knowledge-based is based on external sources like machine readable dictionaries, like WordNet, Thesauri, etc. Supervised approach is based on annotated corpora for sense disambiguation, whereas, Unsupervised approach uses un-annotated corpora and can be sometimes used with machine readable dictionaries. Unsupervised approach has the ability to overwhelm knowledge acquisition problems. Clustering word occurrences of unsupervised methods are capable to find word senses [3].

II. REALATED WORK

Ref. [1] Hindi Word Sense Disambiguation, Authors have proposed first WSD approach for Hindi language. They have used Hindi WordNet for finding sense of the given word in a context. They have first created context of word by considering its following and preceding sentences. In addition, they have built context of that word from Hindi WordNet relations. They have included Synonyms, Hypernyms, Hyponyms, Meronyms and glosses and examples of all them. Then, both contexts have been intersected and the highest overlapping context of WordNet will be winner sense. They have tested this approach on different domain corpus of Hindi.

Ref. [4] An Unsupervised Approach to Hindi Word Sense Disambiguation was for Hindi language and authors have used unsupervised approach for word sense disambiguation. The authors have created a decision list using un-tagged instances from which some are manually provided. First stemming was

978-1-4799-9991-0/15/$31.00 ©2015 IEEE

performed to get the root word, then stop words were removed from the context text. Decision list is further used for annotating those words which are ambiguous with its right meaning in the given context. 20 ambiguous words were taken for experiment and evaluation was done on those 20 words with its multiple senses from Hindi WordNet. Training set was of 1856 words and testing data was 1641 words. Accuracy was the measure of performance. The research investigation proves that stop word removal improves the performance of given approach. Stemming also does the same thing.

Ref. [5] Measuring Context-Meaning for Open Class Words in Hindi Language, authors have discussed in this paper, the word sense disambiguation for Hindi language with use of graph connectivity measures and Hindi WordNet. For graph connectivity measure based on graph clustering, they have considered denseness, graph randomness, edge density. Method was tested on 500 Hindi sentences of Hindi WordNet. They have proposed node neighbors and graph clustering for sense disambiguation from which node neighbors was giving better performance.

Ref. [10] Performance Comparison of Word Sense Disambiguation Algorithm on Hindi Language Supporting Search Engines, Hindi language search engines also have the problem of word sense disambiguation. Highest sense count is the method used by authors for sense disambiguation. Their approach works better with Google. The main goal of this research was comparative analysis of word sense disambiguation algorithms on different Hindi search engines which are Guruji, Raftaar and Google. They have calculated Sense Count (SC) to detect ambiguity. In addition, calculated Phrase Frequency that is occurrence count of query phrase in documents also from test corpus it counts gloss and hypernym. For which value of sense count is highest decides the context of that document. Experiment result gives better result in Google.

Ref. [11] Mining Association Rules Based Approach to Word Sense Disambiguation for Hindi Language, Authors have applied the Data Mining concept here by using Association Rule Mining method to find sense of ambiguous word with the help of Hindi WordNet of IIT Bombay. They have experimented this on one sentence containing total 6 nouns. This experiment was only for nouns. They got 72% average precision for given sentence.

Ref. [12] Evaluating Effect 0f Context Window Size, Stemming and Stop World Removal on Hindi Word Sense Disambiguation, Aim of this paper is to analyses how stop word removal, stemming and context window size affect the result of sense disambiguation in Hindi. They have manually created sense-tagged corpus which contains only nouns of Hindi words. They have used Hindi WordNet prepared by IIT Bombay for sense definition . They have first created context vector using four cases. First case was without stemming and stop word removal, second was with stemming, third was with stop word removal and fourth one was with stemming and stop word removal. Then they have created sense vector from dictionary. Then they have found similarity between context vector and sense vector then counted sense score. One with highest sense score gives sense for ambiguous word.

III. PROPOSED WORK

From literature review it has been observed that techniques do not provide effective result of sense disambiguation. Other parts-of-speech except nouns are still lacking in research. Indian language is also lacking in research work. So, the main concern of this proposed work is Hindi which is an Indian language and to improve the performance by combining the Unsupervised approach with Knowledge-based, which is Hindi WordNet product of IIT Bombay. It has also been observed that only intersecting context with knowledge-based dictionary does not give accurate result as context may not contain necessary information for sense disambiguation.
So, clustering techniques from unsupervised approach is used here with Hindi WordNet, as cluster output contains similar group of words so it can provide better result than only considering context. Following flow diagram will explain how clustering techniques will work with Hindi WordNet to disambiguate senses.

Flow of Proposed Work
Following steps are carried out in the present work:

A. Pre-Processing

Pre-Processing includes Tokenization, POS tagging, Stemming and Stop Word Removal. POS tagging will give Part-of-Speech tagging, means whether it is a noun, an adjective, etc of word in sentence. Then stemming is applied which convert word into its root form by removing suffix from the word. Next stop words are removed from the text as stop words have no meaning for word sense disambiguation.

B. Context Vector Creation

Context vector of word is 2 words left from the word and 2 words right from the word which is standard for creating this vector,their part-of-speech and co-occurrence of those words with a word for which context vector is created.

C. Co-occurrence Matrix Creation

First, the total number of words in the text is counted. Suppose total numbers of words are N, then size of matrix will be N × N, and entries of matrix are co-occurrence to that pair.

D. Pointwise Mutual Information (PMI) weight

PMI is Oldest and most used in computational linguistics. In information theory, PMI quantifies extra-information (in bits) about possible occurrence of w_2 when we know that first word is w_1 [16]. The formula:

$$PMI(w_1, w_2) = log2 \frac{P(w_1, w_2)}{P(w_1)P(w_2)} \quad (1)$$

It is logarithm ratio of empirically estimated probability of bigram and theoretical probability under independence logarithm of ratio of P(w2|w1) which is probabilities of appearing second word if the first word has appeared to P(w2) which is probability of second word independently of context [16]. Apply usual maximum likelihood estimates C() is a

counting function; which count how many times w_1, w_2 occurs together.):

$$\frac{P(w_1,w_2)}{P(w_1)P(w_2)} = \frac{\frac{C(w_1,w_2)}{N}}{\frac{C(w_1)}{N}\frac{C(w_2)}{N}} = \frac{C(w_1,w_2)}{N} \times \frac{N^2}{C(w_1)C(w_2)}$$

We need to take logarithm of

$$\frac{C(w_1,w_2)N}{C(W_1)C(w_2)}$$

Given

$$\log A \times B = \log A + \log B$$

$$\log \frac{A}{B} = \log A - \log B$$

We derive

$$PMI(w_1,w_2) = log_2(C(w_1,w_2)) + log_2(N) - log_2(C(w_1)) - log_2(C(w_2)) \quad (2)$$

As shown from equation we will count PMI measure for every pair words in context vector.

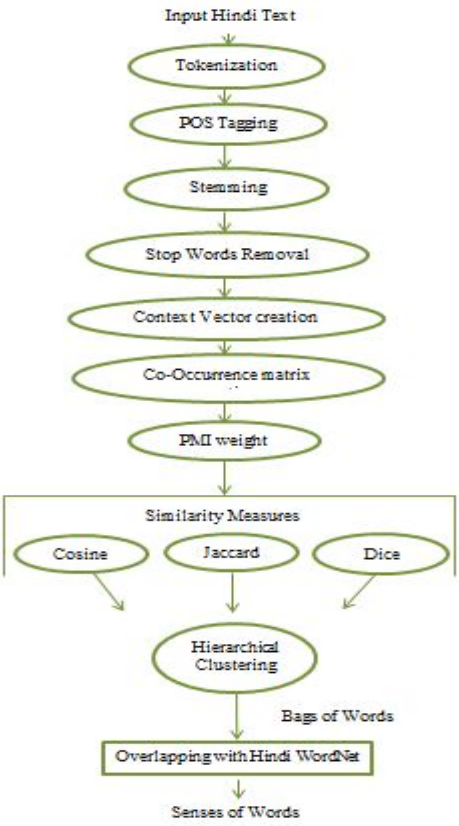

Fig 1. Flow Diagram of Proposed Work

E. Similarity Measure

Similarity measures create matrix for all similarity measures. Calculation of similarity measures is described as follow.

1. Cosine Similarity

The cosine similarity is mostly used to fine similarity between two vectors. It calculates cosine angle between vectors. Result of this similarity is the matrix form which tells how one word is similar to other word [14].

$$\vec{a} \cdot \vec{b} = \|\vec{a}\| \|\vec{b}\| cos\theta \quad (3)$$

$$cos\theta = \frac{\vec{a} \cdot \vec{b}}{\|\vec{a}\| \|\vec{b}\|} \quad (4)$$

Above formula is used to calculate cosine similarity. Here, a and b are word vector, which are created from co-occurrence matrix.

2. Jaccard Similarity

To find similarity among documents, Jaccard Similarity is a simple yet very intuitive measure [14]. Following equation defines it:

$$Similarity(A,B) = n(A \cap B)/n(A \cup B) \quad (5)$$

Using principle of Inclusion and Exclusion above equation reduces to following form:

$$Similarity(A,B) = \frac{n(A \cap B)}{n(A) + n(B) - n(A \cap B)} \quad (6)$$

Where:

$$n(A \cap B) = \sum \min(A_i, B_i)$$

$$n(A) = \sum A_i$$

$$n(B) = \sum B_i$$

i = [0... n-1], where n is number of times term occur in term-document matrix.

3. Dice Similarity

Dice similarity is defined as

$$Similarity(A,B) = \frac{2n(A \cap B)}{n(A) + n(B)} \quad (7)$$

Where:

$$n(A \cap B) = \sum \min(A_i, B_i)$$

$$n(A) = \sum A_i$$

$$n(B) = \sum B_i$$

i = [0... n-1], where n is number of times term occur in term-document matrix.

F. Clustering Technique

Hierarchical Clustering Algorithm

Algorithm works on distance or similarity matrix. We have used three similarity matrices as described above. Therefore, algorithm is producing result set for each similarity.

In starting, we have N items which is total number of input words to the process and similarity matrix of dimension N * N. we have used average linkage clustering algorithm here as it

produces better result than single linkage and complete linkage. Algorithm is as follow:

- First give each item to a cluster. We have N words so N clusters are there. Take distances or similarities among clusters as same as the distances or similarities among the items that clusters have.
- Now find the clusters which are closest and merge them in a one cluster, thus one cluster will be less.
- Now calculate the similarities or distance between newly created cluster and the clusters which are old.
- Until all words are clustered into one single cluster which will have size N repeat steps 2 and 3.

Step 3 has following procedure to find distances between clusters:

In average linkage, the average distance between pairs of observations is used to calculate the distance between clusters. Average linkage yields clusters with same variance.

It uses following formula to calculate this distance:

$$D_{KL} = \sum_{i \in C_K} \sum_{j \in C_L} \frac{d(x_i, x_j)}{N_K N_L} \quad (8)$$

Where K and L are two clusters, x is observation of clusters and i and j are used to iterate all observation from clusters. $N_K$and N_L are total number of observation in cluster K and L respectively.

G. Overlapping with Hindi WordNet

For finding sense of word we use Hindi WordNet. The cluster result is a bag of words for input words. First step is finding the synsets, glosses and examples and Hindi WordNet relations of each word. Then cluster result of those words will be intersected with those synsets, glosses, examples, hypernyms and hyponyms of Hindi WordNet relations. Synsets means synonyms of word; glosses means textual definition of synset; examples tells synset usage; Hypernym called is-a relationship, for example color is hypernym of blue; hyponym is inverse of hypernym. For each sense count is noted for intersection.

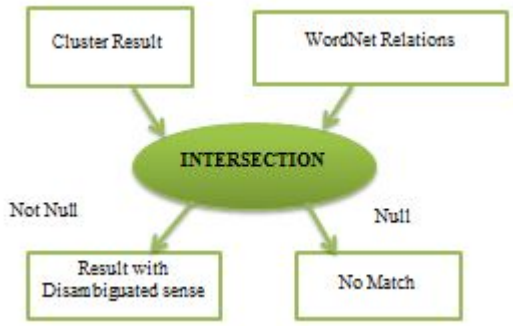

Fig 2. Overlapping cluster result with Hindi WordNet Synsets

Sense for which count is highest is correct sense for given word and if intersection result is null then result is no match of that sense.

Hindi WordNet

The Hindi WordNet gives various lexical and semantic relations that exist among Hindi words. It also includes glosses and examples for each word. Authors from IIT Bombay have taken inspiration from English WordNet to create Hindi WordNet[15].

Construction of this WordNet is still going on to include all the Hindi words. Current status of Hindi WordNet is as follows:

26,208 are the numbers of Synsets present in WordNet and 56,928 are total unique words available in it [15].

IV. EXPERIMENT SETUP

Experiment is conducted on different input words from different domain. First 246 input words are from History, second 1279 input words are from Social study, and 2672 input words are from Short story. Results vary according to the number of input words and according to similarity measures. Following two tables describe overall result of clustering techniques. Table1 shows results of overlapping of Hierarchical clustering results with Hindi WordNet.

V. RESULT

If only context is used to overlap with knowledge-based dictionary, then it does not give required output, as context does not contain all the information required for sense disambiguation. Clustering technique gives similar group of words and with knowledge source it gives better result

TABLE 1. Result of Overlapping

Similarity Measure	Input words	Precision (%)
Cosine	246	81.64
Jaccard	246	80.38
Dice	246	79.74
Cosine	1279	75.94
Jaccard	1279	75.80
Dice	1279	74.24
Cosine	2672	75.56
Jaccard	2672	75.21
Dice	2672	74.17

TABLE 2. Result of previous approaches

Approach	Input words	Result
Knowledge-based with context[1]	Tested on different domain (only for noun)	40% - 70% varies according to domain
Node neighbors connectivity[5]	1200	60(Accuracy %)
Graph clustering[5]	1200	41.25(Accuracy %)
Genetic algorithm[9]	12(only for noun)	91.6(Recall %)
Mining Association Rule[11]	6(only for noun)	72(Precision %)
Unsupervised approach with decision list classifier[4]	20	82 (Accuracy %)
Graph based approach[13]	913(only noun)	65.17 (Accuracy %)

From experiment, it is clear that Hierarchical clustering gives better result than previous approaches, which are shown with result in TABLE 2. Hierarchical clustering gives different precision for three of similarities, that is 81.64% for Cosine, 80.38% for Jaccard and 79.74% for Dice similarity on 246

input words from History domain. Performance depends on the number of input words plus from which domain corpus was. It has also been observed that results were not varying much when input words were 1279 and 2672. Also, for some words all three similarities gives different sense plus for some words cosine gives wrong sense but jaccard and dice gives correct sense for same word and vice-versa.

VI. CONCLUSION AND FUTURE WORK

This is the first experiment, which uses clustering techniques with knowledge base for Hindi language. Overall precision varies from 74% to 82%. It is better from previous approaches as it works for nouns, adjectives, adverbs and verbs, etc. For future work, other clustering which are applicable for sense disambiguation can be taken into consideration and result can be evaluated with large number of input words and with other domain.

REFERENECES

[1] Manish Sinha, Maheshkumar Reddy .R, Pushpak Bhattacharyya, Parabhakar Pandey, Laxmi Kashyap, "Hindi word sense disambiguation," Department of Computer Science and Engineering, IIT, Mumbai 2008

[2] Bartosz Broda, Wojciech Mazur, "Evaluation of clustering algorithms for Polish word sense disambiguation," Proceedings of the International Multi conference on Computer Science and Information Technology IEEE, 2010, ISBN 978-83-60810-27-9, ISSN 1896-7094, pp. 25-32.

[3] Roberto Navigli, "Word sense disambiguation: A survey", ACM Computing Surveys, Feb 2009, Volume 41, Issue 2, Article No- 10, pp. 10.1-10.

[4] Neetu Mishra, Shashi Yadav and Tanveer J. Siddiqui, "An unsupervised approach to Hindi word sense disambiguation" First International Conference on Intelligent Human Interaction, Springer India, 2009, Print ISBN: 978-81-8489-404- 2, pp. 327-335.

[5] Amiat Jain, Sudesh Yadav, Devendra Tayal,"Measuring context-meaning for open class words in Hindi language", Sixth International Conference on Contemporary Computing IEEE , 2013, ISBN: 978-1-4799-0190-6, pp. 118 – 123.

[6] Sayali Charhate, Anurag Dani, Rekha Sugandhi,"Adding intelligence to non-corpus based word sense disambiguation", 12th International Conference on Hybrid Intelligent Systems IEEE, 2012, Print ISBN: 978-1-4673-5114-0, pp. 173 – 178.

[7] Francisco Tacoa, Danushka Balloegala and Mitsuru Ishizuka, "A context expansion method for supervised word sense disambiguation" Sixth International Conference on Semantic Computing IEEE, 2012, Print ISBN: 978-1-4673-4433-3, pp. 339 – 341.

[8] Udaya Raj Dhungana, Subarna Shakya, "Word sense disambiguation in Nepali Language" Fourth International Conference on Digital Information and Communication Technology and it's Applications IEEE, 2014, Print ISBN: 978-1-4799-3723-3, pp. 46 – 50.

[9] Sabnam Kumari, Prof. Paramjit Singh, "Optimized word sense disambiguation in Hindi using genetic algorithm" International Journal of Research in Computer and Communication Technology, 2013, Print ISSN: 2320-5156, pp. 445-449.

[10] Parul Rastogi, Dr.S.K. Dwivedi, "Performance comparison of word sense disambiguation algorithm on Hindi language supporting search engines" International Journal of Computer Science Issues , 2011, vol. 8, Issue 2, Online ISSN: 1694-0814, pp. 375-379.

[11] Preeti Yadav1, Sandeep Vishwakarma, "Mining association rule based approach to word sense disambiguation for Hindi language" International Journal of Emerging Technology and Advanced Engineering, 2013, ISSN 2250-2459, vol. 3, Issue 5, pp. 470-473.

[12] Satyendr Singh, Tanveer J. Siddiqui, "Evaluating effect of context window size, stemming and stop word removal on Hindi word sense disambiguation", International Conference on Information Retrieval & Knowledge Management IEEE , 2012, Print ISBN: 978-1-4673-1091-8, pp. 1 – 5.

[13] Sandeep Kumar Vishwakarma, Chanchal Kumar ishwakarma, "A graph based approach to word sense disambiguation for Hindi language", International Journal of Scientific Research Engineering & Technology, 2012,ISSN: 2278 – 0882, vol. 1, Issue 5, pp. 313-318.

[14] G.R.J.Srinivas, Niket Tandon,Vasudeva Varma, "A weighted tag similarity measure based on a collaborative weight model", SMUC'10, October 30, 2010, Toronto, Ontario, Canada. Copyright 2010 ACM 978-1-4503-0386-6/10/10.

[15] HindiWordNet: http://www.cfilt.iitb.ac.in/wordnet/webhwn/

[16] Hindi Corpus: http://www.ciil.org

[17] Natural Language Processing: https://www.scm.tees.ac.uk/isg/website/d-Ownloads/lkit.

[18] Patrick André Pantel, PhD Thesis "Clustering by commitee", Department of Computing Science, University of Edmonton, Alberta, Springer 2003.

[19] E.Agirre, P.Edmonds, "Word sense disambiguation algorithms and application", Springer, New York, NY. 2006, vol. 33/

Web users Browsing Behavior Prediction by Implementing Support Vector Machines in MapReduce using Cloud Based Hadoop

Pradipsinh k. Chavda
Department of Computer Science and Engineering
Government Engineering College
Modasa,Gujarat,India
pradipchavda.it@gmail.com

Prof. Jitendra S. Dhobi
Department of Computer Science and Engineering
Government Engineering College
Modasa, Gujarat, India
jsdhobi@gmail.com

***Abstract*—The motivation behind the work is that the prediction of web user's browsing behavior while serving the Internet, reduces the user's browsing access time and avoids the visit of unnecessary pages to ease network traffic. This research work introduces the parallel Support Vector Machines for web page prediction. The web contains an enormous amount of data and web data increases exponentially, but the training time for Support vector machine is very large. That is, SVM's suffer from a widely recognized scalability problem in both memory requirement and computation time when the input dataset is too large. To address this, we aimed at training the Support vector machine model in MapReduce programming model of Hadoop framework, since the MapReduce programming model has the ability to rapidly process a large amount of data in parallel. MapReduce works in tandem with Hadoop Distributed File System (HDFS). The so proposed approach will solve the scalability problem of present SVM algorithm. The performance of the proposed approach is evaluated in amazon cloud EC2 using cloud-based Hadoop. Our experiments show the effectiveness in term of training time and improved the preprocessing time. We find in our research study that a number of nodes increased the training time of proposed algorithm is decreased. We checked that parallelization of SMO has no more negative effect on accuracy level as compared to the standard approach.**

Keywords— Web Page Prediction, Support Vector Machines, Hadoop, MapReduce, HDFS,Amazon EC2.

I. INTRODUCTION

Web page prediction [1] is a kind of classification problem which helps to predict the next such set of web pages that a user might visit based on the knowledge of access behavior of the past visited web pages. It needs the understanding of mining the web usage patterns. The goal of web usage mining is to find and analysis of usage patterns from web server logs.

In Web page prediction, we have the source of input data is actual user sessions. The user sessions give the information regarding the sequence of page visited by a user in given time span. The data regarding the path visited, browsing rate and relative duration of access time is considered while discovering user's interest on the web.

In Web page prediction, two main challenges that we have faced name as preprocessing and prediction challenges. Preprocessing challenges are, handling the huge amount of data that generally cannot be loaded in memory in a single scan, choose the perfect size of sliding window, recognizing sessions. Prediction challenges are the limitation of memory, needs more time for training and prediction and accuracy of the classifier are lower. To address these challenges, we want to implement the web page prediction problem using MapReduce programming model of Hadoop framework. Apache Hadoop is an open-source software framework for distributed storage and handling a large amount of data. MapReduce programming model of the Hadoop framework has the ability to process a very large amount of data in distributed manner.

This paper covers various sections. Section II describes literature survey. Section III describes about background study. Section IV describes problem gap. Sections V describes proposed approach. Section VI describes experimental results. Section VII describes Conclusion and future work.

II. LITERATURE SURVEY

A. Related Work

The idea behind of the Markov model [2, 3] is to forecast the next web page depending on the history of previous web pages. The Kth-order Markov model is defined as that a user will visit kth page when he or she has visited k-1 page. But, the problem with Markov model is that it cannot forecast for a session that did not appear in the training dataset. In all-Kth Markov model [2, 3], we generate all orders of Markov models and utilize them collectively in prediction. It predicts the next page by constructing all order of Markov if gets fail then it use another order Markov model. Compare to Markov model, it gains far better prediction and when all Markov models fail than it becomes fail also. The

978-1-4799-9991-0/15/$31.00 ©2015 IEEE

goal of modified Markov model [2] is to consider the different order of web page as a same, hence it reduce its size. That means session <a, b> is consider same as session <b, a>.

In [4], propose an algorithm based on longest common Subsequence concept for web page prediction. In his research work, they divide architecture in two main phase: offline phase and online phase. In [5], propose an algorithm based on weighted algorithm rule. The approach will use three parameters name as page frequency, page time spent and page click history to assign a quantitative weight to each page for a user. After that based on the weight of each page, it recommends to users.

In [6], proposed a hybrid approach using the artificial neural network. They used backpropagation algorithm for multilayer neural network learning and prediction. Problem with the present approach is that they are costly and very complicated for large datasets. Large data sets are not possible to train using these hybrid approaches. In [3], proposed a hybrid approach using Support Vector Machines.

In [7], proposed distributed support vector machine (DSVM) algorithm that finds support vectors (SVs) on strongly connected networks. In [8], proposed parallel optimization methods for Kernel Support Vector Machines on multicore CPUs and GPUs. In [9], had an algorithm that implemented distributed processors into cascade top-down network topology, namely Cascade SVM.

III. BACKGROUNG STUDY

A. *Apache Hadoop Framework*

Apache Hadoop [14] is an open source software framework for distributed storage and handling and processing large amount of data.

The Apache Hadoop framework includes following modules:

- Hadoop Common
- Hadoop distributed File System (HDFS)
- Hadoop YARN
- Hadoop MapReduce

B. *Hadoop Distributed File System*

Hadoop Distributed File System [15] is extended version of the Google's Google File System (GFS). The work of HDFS is responsible for storing the data on a cluster of machines. There is single node and multiple nodes possible in HDFS. Master node contains meta information of the file. By default, HDFS divide the data into 64MB block and divide among the nodes in the The client sends a request of a file to the Name Node only. HDFS has mastered and slave architecture. An HDFS cluster has a one name node, a multiple data nodes

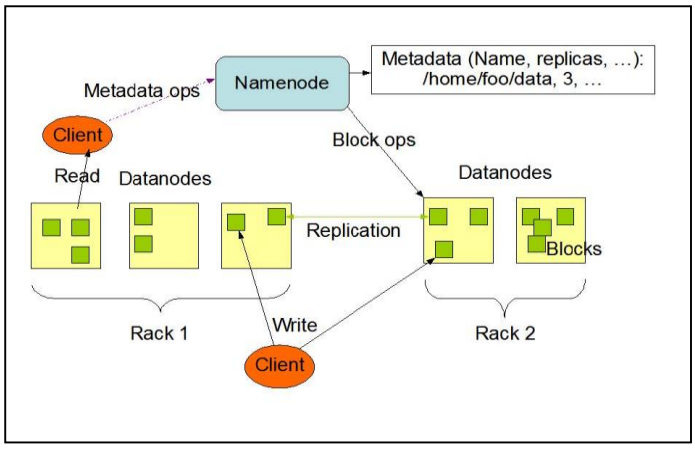

Fig. 1. Architecture of HDFS [15]

C. *MapReduce Programming Model*

MapReduce [10, 12] is a programming model and an associated implementation for processing and generating large data sets with a parallel, distributed algorithm on a cluster. Users specify a *map* function that processes a key/value pair to generate a set of intermediate key/value pairs, and a *reduce* function that merges all intermediate values associated with the same intermediate key.

D. *Data sets Preprocessing*

In [11], proposed various tasks related to preprocessing that need to be performed before giving input to any classifications algorithms. Source of input data to the preprocessing step is HTTP web server log. And we will use same for our collected web server logs. They describe the following steps:

- Collect required Web server log files, which contains all information about visitor's activity by the Web server when it service user request.
- Next, a step is to clean the log file by finding irrelevant log entry. It includes URL with image extension like jpg, gif etc., scripts can be removed.
- Next step is to identify the unique users from the log file. It is possible that same ip may have assigned to the different user in the different network. This kind of situation can be solved by checking browser information of log entries.
- Next step is to find the session of the particular user from log files.

E. *N-Gram Representation of Sessions*

N-gram [1, 2] means subsequence of N contiguous items within a sequence of items. N-gram can be represented in the form of tuple <n1, n2…..nn> to represent the click stream. For example, the n-gram <P1, P2,P3,P4> for some user A, it means that user A has visited web page in the sequence of web page 1,

web page 2, web page 3, web page 4 and each number has the corresponding URL.

F. Feature Extraction

In [3], Proposed feature extraction and we have employed it on our research work, as we know the source of input data is server log and it contains limited amount of information like timestamp, URL visited etc. Among all information many of are useless for web page prediction that means we have limited amount of features in the log entry, so we need to extract more feature.

G. Support Vector Machines

A Support Vector Machine (SVM) [16] is a classifier and it defined by using a separating hyperplane. For the set of input example, SVM maps each of example to one of the categories, results of SVM training is the model that help to predict the new example to one of these class. The goal of SVM is to find a maximum margin hyperplane that divides the data into two sets. The training vector near to the hyperplane is defined as support vector.

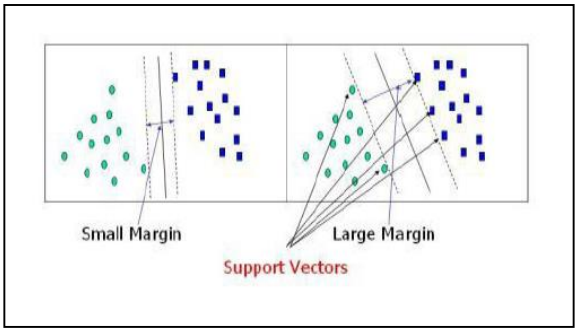

Fig. 2. Support Vector

H. Multiclass classification

In case of multiclass data sets[16], there are two main schemes, namely:

- One-vs-one
- One-vs-all

The one-vs-one approach makes a classifier for each pair of classes. The total number of classifiers computed is $n(n-1)/2$, where n is the number of classes in the data set. A new instance x belongs to the class upon which most classifiers agree, i.e., majority voting. One-vs-all creates a classifier for each class in the data set against the rest of the classes. Both are basically extensions to the binary classification of SVM.

I. Why SVM??

- Stable with the changes on data
- SVMs classification shows greater accuracy in predicting seen as well as unseen data as compared to Markov model
- The Diversity of the kernel tricks for different problems.
- High accuracy

IV. PROBLEM GAP

For classification using support vector machine, WEKA Tool [17] is available. Weka is java based open-source software framework and it has various machine learning algorithms.

For the experiments we used, WEKA version 3.6, Operating System: 64-bit Windows 7 Home Premium with Intel Core i5 CPU @ 2.30 GHz and 3 GB of RAM. Also, we take the preprocessed dataset in ARFF format from the library of the university of British Columbia (UBC).

Dataset name: Amazon
Record: 1050, 1065, 1080
Class: 50

We used LIBSVM [19] library in weka for the experiment purpose and set the parameter like SVM type, kernel type etc.

TABLE I. SVM Training Time in Weka

No of Instance & Size of file	Training Time (sec)
1050 (20.3 MB)	55.36
1065 (20.6 MB)	58.78
1080 (20.9 MB)	60.03

Awad et al. [3] perform the experiments and saw that for 5430 different web page, they need to create 5430 classifiers, so total time of training for SVM is 26.3 h.

A. Problem Statement

From the above result we noted, when the size of training data increase it will increase the training time proportionally and having the lower prediction accuracy. At some point, weka will not work because of computation power and memory limitation of the system itself. So we have to come up with new approach that parallelize the computation and improve the accuracy and lower the training time when the source of input data is too large.

V. PROPOSED APPROACH

As we discussed earlier, prediction of web user's browsing behavior using support vector machines and other classification algorithm with big data is having extensive training and prediction time and low prediction accuracy. So we have to

overcome this scalability problem by using parallel processing using Hadoop framework with MapReduce programming model.

A. Logical Design of Proposed Work

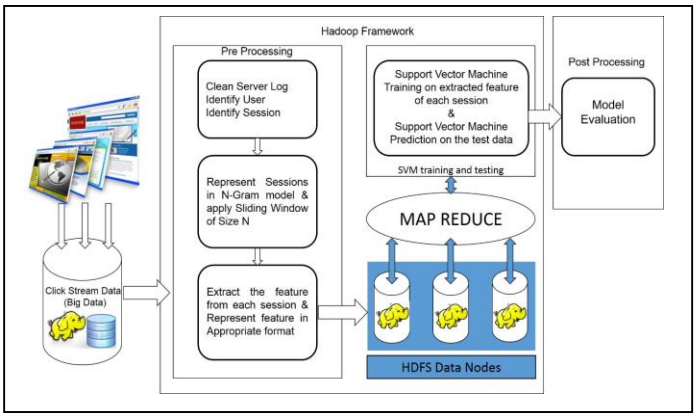

Fig. 3. Overview of Proposed System

B. Proposed Parallel SVM Training algorithm

Input: Training dataset having feature and class label of each session
Output: Resultant Model

Divide dataset into pair of class

1. Algorithm of Mapper in MapReduce for training

I. Read input training file.
II. Train each session in the divided dataset using SVM classification.
III. Output the model along with the number of local support vector, alpha array and bias etc.

2. Algorithm of Reducer in MapReduce for Testing

I. Take input from the mapper.
II. Compute the global SVM model by combining all the support vectors generated by the different model and measure the overall training time.

C. Design of Proposed SVM training algorithm

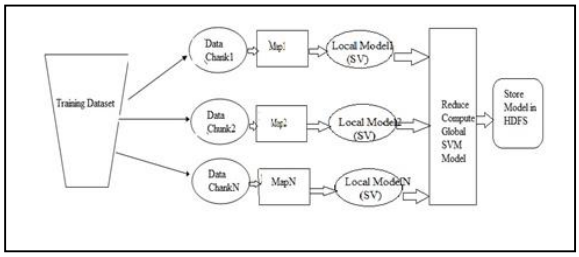

Fig. 4. Parallel SVM Training Algorithm

D Proposed Parallel SVM Prediction algorithm

Input: Testing dataset, model
Output: Next page prediction for each of session in testing datasets

1. Algorithm of Mapper in MapReduce for Testing

I. R**e**ad the input testing file and model.
II. Predicting next web page for each session in testing set by consulting with different classifiers or models.
III. Write the output into the file.

2. Algorithm of Reducer in MapReduce for Testing

I. Take input from each mapper.
II. Merge the file generated for each mapper process.
III. Measures overall prediction time and accuracy of the model.

VI. EXPERIMENTAL RESULTS

A. Experiment Setup

For the experiment of **single node Hadoop cluster based approach** we used Operating System: 64-bit Ubuntu 14.04 LTS with Intel Core i7 CPU @ 2.20 GHz and 4 GB of RAM, Hadoop 1.2.1, JDK 1.7 and Eclipse Kepler. In which all daemons are running on the single machine.

For the experiment of **multi-node Hadoop cluster based approach,** we have setup cluster on amazon web service cloud using elastic cloud computing (EC2) [18]. For our series of experiments we have used t2.micro instance which is based on Intel(R) Xeon family CPU @ 2.5 GHZ processor with 1 GB of RAM, 8 GB hard disk and operating system is 64 bit Ubuntu Server 14.04 LTS. Hadoop 1.2.1 and 64 bit OpenJdk 1.7. For the experiments, we have setup 1 node, 2 nodes and 3 nodes Hadoop cluster. The multi-node cluster is made up of a total 3 nodes means 1 node as Name Node and 2 nodes as Data Node. Similarly for 2 nodes.

B. Datasets and Data Preprocessing

Http log from NASA server[20]:

Description**:**

We have collected two months Jul and Aug of all HTTP requests from the Internet.

Log Entry:

133.68.18.180 - - [01/May/1998:00:01:48 -0440] "GET /persons/nasa-cm/jmd.html HTTP/1.0" 200 4067

In the preprocessing step, we have first removed irrelevant log entry like HTTP code and we kept HTTP 200 code log entry.Also removed entry containing image file extensions like .gif, .jpeg etc because when we request any pages, associated images will be download too and create log entry. For identification of sessions, we used 15 minutes as default time for experiments that means if the user page request exceed 15 minutes then it creates new sessions. We have used sliding window size 3 for experiments, after extracting the feature as defined in an earlier section and represent the sessions in lib SVM format. Table II shows preprocessing time of input weblog in different approaches.

TABLE II. Summary Results of preprocessing in different approaches

Task \ Approaches	Java based (minutes)	Single Node Hadoop (minute)	Multi Node Hadoop (3 node) (minutes)
Clean & Group record by time	249.98	2.36	1.53
Session Identification	24	1.25	0.57
Ngram Session	3.16	2.93	1.72
Sliding WindowNSize	0.9	0.36	0.24
Feature Extraction	1.8	1.05	0.45

C SVM Training

For the training, we have used Fast training of SVM using sequential minimal optimization [13] and extended, implemented. We took the data from preprocessing step of feature extraction. For the series of experiments, we have considered 10 most accessible URL from log files. Each session has 7 features that we have generated in feature extraction step. We have used One VS One approach for SVM training due to its work on the subset of data.

Fig. 5. Indicate the training overhead of sequential SMO versus parallel SMO. From that, we see that if the training instance is smaller than SMO work better than proposed approach.But when the size of training instance increase than proposed approach is better than the Sequential SMO.

Fig. 6. saw the training overhead of parallel SMO on a different number of nodes on Amazon Cloud EC2. From that, we see that if the node is increased in the cluster than corresponding sessions training time is decrease.

Besides that, we evaluated the accuracy of standard SMO and parallel SMO in classification .we randomly check the different session and average accuracy level considered. It is clear that parallelization of SMO has no more negative effect on accuracy level as compared to the standard approach. The results show that parallel SMO achieves 89 % which is same as the sequential SMO.

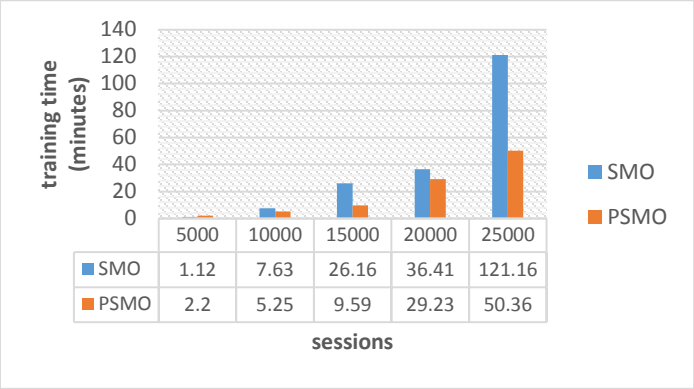

Fig. 5. Comparison of SMO and PSMO training time

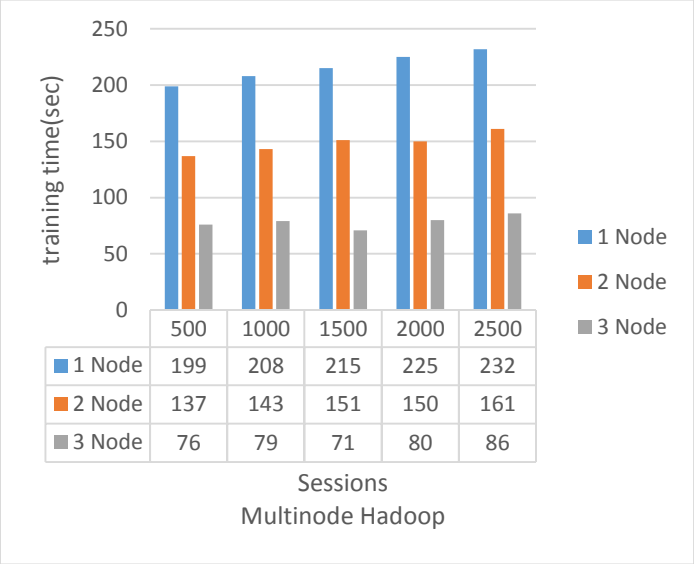

Fig. 6. Comparison of PSMO training time on different number of node in Amazon cloud EC2

VII. CONCLUSION AND FUTURE WORK

Web page prediction is important since the prediction of the user's browsing behavior reduces the browsing access time and avoids the visit of unnecessary pages, to ease network traffic. According to our observations, Support vector machine and many other machines learning algorithms do not fit when the source of input data is too large. We have proposed parallel support vector machines for web page prediction based on MapReduce programming model and runs on Hadoop framework. It removes scalability problem of present SVM algorithm. From the experiments, we have improved the preprocessing time by

comparing the results with non-Hadoop based approach and Hadoop-based approach. Besides that proposed parallel SVM algorithm reduce the training time compare to sequential SVM. We have also checked the performance of SVM training algorithm by considering different node on cloud-based Hadoop . We checked that parallelization of SMO has no more negative effect on accuracy level as compared to the standard approach.

In this research work we have used one vs one approach and we are planning to check the results against the one vs all approach of multiclass classification. In future, we will also check the performance of proposed algorithm in heterogeneous environments.

ACKNOWLEDGMENT

We take this opportunity to express our profound gratitude and deep regards to my guide Professor J.S.Dhobi (Head of the Computer Science & Engineering Department), friends, family and others faculty member of CSE Department of Government Engineering College, Modasa, Gujarat, India for their exemplary guidance, monitoring and constant encouragement throughout the course of this paper. The blessing, help and guidance given by them time to time shall carry us a long way in the journey of life on which we are about to embark.

REFERENCES

[1] Pruthvi R "Web-Users' Browsing Behavior Prediction by Implementing Neural Network in MapReduce", IJAFRC vol. 1, issue 5, May 2014.

[2] Mamoun A. Awad and Issa Khalil, "Prediction of User's Web-Browsing Behavior: Application of Markov Model", IEEE Transactions On Systems, Man, And Cybernetics—Part B: Cybernetics, vol. 42, no. 4, pp. 1131-1142, August 2012.

[3] M. Awad, L. Khan, and B. Thuraisingham, "Predicting WWW surfing using multiple evidence combination," *VLDB J.*, vol. 17, no. 3, pp. 401– 417, May 2008.

[4] Jalali, Mehrdad; Mustapha, Norwati; Mamat, Ali; Sulaiman, Md. Nasir B, "A new classification model for online predicting users' future movements", International Symposium on Information Technology,IEEE ,pp. 1-7,2008.

[5] Agarwal, Rohit; Arya, K.V.; Shekhar, Shashi; Kumar, Rakesh "An Efficient Weighted Algorithm for Web Information Retrieval System" IEEE 2011 International Conference on Computational Intelligence and Communication Networks (CICN) , pp. 126-131, 2011.

[6] M. Awad and L. Khan, "Web navigation prediction using multiple evidence combination and domain knowledge," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 37, no. 6, pp. 1054–1062, Nov. 2007.

[7] Lu et al. Distributed parallel support vector machines in strongly connected networks. IEEE Trans. Neu- ral Networks, 19, 1167-1178 (2008)

[8] Stephen Tyree, Jacob R. Gardner, Kilian Q. Weinberger, Kunal Agrawal, and John Tran. Parallel Support Vector Machines in Practice. Technical Report, arXiv:1404.1066, 2014.

[9] Graf,H. P., Cosatto, E., Bottou, L., Durdanovic, I., Vapnik, V.: Parallel support vector machines: The cascade SVM.In: Proceedings of the Eighteenth Annual Conference on NIPS, pp. 521-528. MIT Press, Vancouver (2004)

[10] Jeffrey Dean and Sanjay Ghemawat. MapReduce: Simplified data processing on large clusters. Communications of the ACM, 51(1):107–113, 2008.

[11] R. Cooley, B. Mobasher, and J. Srivastava, "Data preparation for mining World Wide Web browsing patterns," *J. Knowl. Inf. Syst.*, vol. 1, no. 1, pp. 5–32, 1999.

[12] Tom White, Hadoop the definitive guide Yahoo Press, second edition, 2011

[13] John C. Platt, "Fast Training of Support Vector Machines Using Sequential Minimal Optimization", MIT Press, January 1998.

[14] Hadoop official site May 2015, http://hadoop.apache.org/core/.

[15] HDFS Architecture Guide, April 2013: http://hadoop.apache.org/docs/r1.2.1/hdfs_design.html

[16] Support Vector Machines, December 2014: http://en.wikipedia.org/wiki/Support_vector_machine

[17] Weka: http://www.cs.waikato.ac.nz/ml/weka/

[18] Amazon EC2: http://en.wikipedia.org/wiki/Amazon_Web_Services

[19] Libsvm: http://www.csie.ntu.edu.tw/~cjlin/libsvm/

[20] Internet Traffic Archive: http://ita.ee.lbl.gov/html/ traces.html

Brain Computer Interface: A Review

Parmar Prashant[1], Anand Joshi[2], Vaibhav Gandhi[3]

[1]*Mechatronics Department, TeamLease Skills University, Vadodara, India*
E-mail prashant.parmar@teamleaseuniversity.org
[2]*Mechatronics Department, G. H. Patel College of Engg. & Tech., Vallabh Vidyanagar, India,*
E-mail anandjoshi@gcet.ac.in
[3]*Department of Design Engineering and Mathematics, Middlesex University, London, UK,* E-mail v.gandhi@mdx.ac.uk

***Abstract* - A brain-computer interface (BCI), also referred to as a mind-machine interface (MMI) or a brain-machine interface (BMI), provides a non-muscular channel of communication between the human brain and a computer system. With the advancements in low-cost electronics and computer interface equipment, as well as the need to serve people suffering from disabilities of neuromuscular disorders, a new field of research has emerged by understanding different functions of the brain. The electroencephalogram (EEG) is an electrical activity generated by brain structures and recorded from the scalp surface through electrodes. Researchers primarily rely on EEG to characterise the brain activity, because it can be recorded non-invasively by using portable equipment. The EEG or the brain activity can be used in real time to control external devices via a complete BCI system. A typical BCI scheme generally consists of a data acquisition system, pre-processing of the acquired signals, feature extraction process, classification of the features, post-processing of the classifier output, and finally the control interface and device controller. The post-processed output signals are translated into appropriate commands so as to control output devices, with several applications such as robotic arms, video games, wheelchair etc.**

Keywords- *Brain-Computer Interface; Electroencephalography; Neural Activity*

I. INTRODUCTION

The ability to communicate brain activity with the peripheral devices is possible through advancements in cognitive neuroscience and brain imaging [1]. The technology has put the brain thought process to be able to monitor or control an activity in real-time. This technology is highly beneficial to the disabled individuals who find it very difficult to communicate. A brain-computer interface (BCI) can assist people with severe motor disability[1], supporting biofeedback training to a patient suffering from the stroke, epilepsy[2] or attentional deficit hyperactivity disorder (ADHD)[3]. The electrodes (sensors) record the change in electrical potential, magnetic field and metabolic supply of ions generated due to the excitation and inhabitation of neural networks based on the brain activity.

A. What is BCI?

A Brain-Computer incapacity Interface (BCI) technology is a means of communication that allows individuals with severe movement disability to communicate with external assistive devices using the electroencephalogram (EEG) or other brain signals [2]**.** The brain is composed of more than 100 billion neurons [3]. Basic and clinical research has yielded detailed knowledge of the signals that comprise the information from these neurons. Recording of these signals gives the (EEG) [4]. The BCI system should be able to classify different EEG signals of brain activity as accurately as possible and the BCI user should learn to produce distinct brain signals to perform the different task**.** BCI has become a synergetic combination of computational neuroscience, physiology, engineering, signal processing, computer science and several interdisciplinary types of research. Research groups have been focusing on several areas ranging from light and television control, yes/no questions, text processing, wheelchair control, robotic prosthetics, autonomous vehicles, auto calling using brain activity, virtual reality games etc. [5] [6] [7] [8]**.**

B. Early History

Richard Caton in the year 1875, first recorded EEG signal from the cortical surface in animals, and later a German neuroscientist Hans Berger discovered electrical signals from the human brain using EEG in the year 1929 [9]. Berger also recorded Alpha Waves (frequency 8-10 Hz) from the human brain for the first time. Adrian and Matthew [10] and Adrian and Yamagiwa [11] discovered that EEG signals varied at different locations on the head and suggested the standardized position of electrodes over the scalp. Later in the 1970s, the Defence Advanced Research Projects Agency of USA initiated a programme to further explore EEG activity. Until this stage, the research on EEG was limited to clinical diagnosis and exploring the brain function only.

C. Advancement.

Until 1995, very few research groups worked on BCI. However, with the advancement in technology and over time, several research teams were involved within this interdisciplinary field of BCI and its applications. The BCI Information Transfer Rates (ITR)[4] increased from 5-25 bits/min [12] to 84.7 bits/min [13]. The improvement in BCI is based on four factors. The first and foremost is that the lives of 'locked-in'[5] patients will be improved considerably with the advancement in BCI technology. With the use of BCI, totally 'locked-in' patients will be able to make their life more productive, although the means of communication and control will still be limited. The second factor is an increase in the understanding of nature and the functionality of EEG and related brain activity. With further research in this area, new means of recording brain signal

[1] Motor disability is a disability that affects a person's ability to learn motor tasks such as moving, walking, running, etc.
[2] Epilepsy is a type of neurological disorder of the nervous system.
[3] ADHD is a childhood behavioural disorder that can continue through adulthood. Symptoms include difficulty staying focused, paying attention etc.
[4] ITR is a ratio of information bits to total bits per character.
[5] It is immobile condition of body due to a stroke which damages part of brainstem.

978-1-4799-9991-0/15/$31.00 ©2015 IEEE

activity may come up, which may give us ideas to understand the advanced functional operation of brain activity. The third factor is the availability and advancement in low-cost microelectronics, which will enable BCI users to perform complicated tasks through embedded circuits. Lastly, it is the recent surge in advanced machine learning algorithms and self-decision making approaches that may help to further explore the boundaries of brain-controlled applications.

II. BRAIN COMPUTER INTERFACE: BASIC

Neurons are the micro-processing stations interconnected with each other [3]. In general neurons have four functionalities: input, trigger, conduction and output [4]. BCI uses the information in the form of the electrical signals generated by either firing or inhibition of these neurons. Based on training/experience, the neurons fire when the severity of the data is above the threshold value, or inhibit if the severity of the data is below the threshold value [14]. The brain is divided into 52 discrete local points and named as cytoarchitectural map [15]. A particular neuron activated on the scalp as electrical pulse or magnetic field based on visual, listening, speaking, physical movement or other routine activity. The signal transmission takes place from one neuron to another through the synapse [3]. Fig. 1 depicts function of BCI from input to the real-time actuation.

The major phases of BCI are:

1. Data Acquisition
2. Signal Processing & Classification
3. Computer Interface
4. Application

A. Data Acquisition

In the BCI, signal acquisition is usually done using three approaches; invasive, partial invasive and non-invasive.

Invasive- is a method of reading brain signal from inside of a grey matter of the brain. The brain signals are taken directly from the cerebral cortex of mind; electrophysiology is an example of this method. This acquisition method requires brain surgery, to place the electrode or sensor inside the skull so that one gets the highest quality of signals. William Dobell devised the invasive technique by implementing 68 electrodes into Jerry's visual cortex in 1978 [16]**.** This technique provides functionality to paralyzed people, but signals become weaker or even lost as the human body has natural characteristics to oppose foreign implantation of an object in any part of a body.

Partially Invasive- is another method of reading brain signal from the skull by the implantation of BCI device outside the grey matter. Electrocorticography (ECoG) is a type of partially invasive BCI [17]. Compared to invasive BCI, the signals acquired using partially invasive method is weak while better than noninvasive. However, it does have the less risk of scar tissue formation as compared to invasive BCI. In this approach, a thin plastic pad is placed above the cortex. Eric Leuthardt and Daniel Moran from Washington University in St. Louis used this first in 2004 [18]. The researchers succeeded to enable a boy to play Space Invaders using his ECoG implant, but they concluded that controlling the kinematic BCI devices having more than one dimension of control are difficult to control [19].

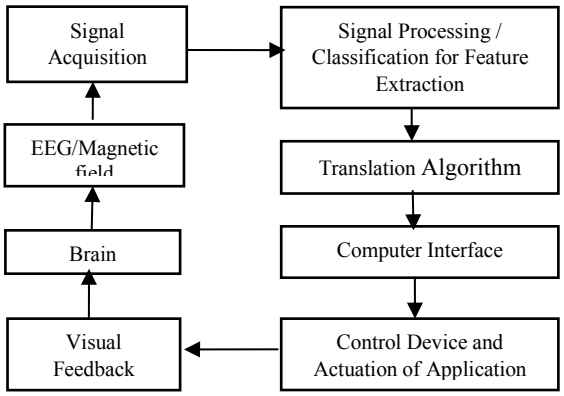

Fig. 1. Block diagram of brain-computer interface

Noninvasive- is the most popular neuron signal imaging method in which electrodes are mounted on the scalp of a person for the acquisition of the EEG. This approach provides reasonable signal quality with low cost and is easy to use. However, the acquired EEG signal is noisy compared to the one obtained using invasive means [20]. There are several non-invasive signal acquisition techniques besides the EEG; such as Magnetoencephalography (MEG), Magnetic Resonance Imaging (MRI) and Functional Magnetic Resonance Imaging (fMRI) and P-300 based BCI, Positron Emission Tomography (PET) [4]. In MEG, superconducting quantum interface devices (SQUIDs) arrays, developed by James E. Zimmerman, to record magnetic fields produced by electrical current flowing naturally in the brain neuron network [21]. In 1968, David Cohen, the physicist of University of Illinois first measured the MEG signals with copper induction coil as a detector [22].

The fMRI is a type of MRI scan that measures the hemodynamic response (change in blood flow) generated by the neural activity of spinal code [23]. The relatively low invasiveness, the absence of radiation exposure and a relatively wide availability, makes fMRI very popular. MEG, Positron Emission Tomography (PET) and fMRI are still technically demanding and expensive methods. In addition to this, PET and fMRI have large time constants and thus are less responsive to rapid communication. At present only EEG is widely used for real-time applications, as it has relatively short time constant and involves easy to use equipment for non-muscular communication. Adrian and Matthews's recommendation to standardise the position of electrode after which the International Federation of Societies for Electroencephalography and Clinical Neurophysiology recommended a specific system of electrode placement for in all laboratories under the standard condition. The standard placement recommended by the American EEG society for use in the International 10-20 system is for 21 electrodes [24].

The conventional approach to acquire EEG signals is by using wet Ag/AgCl electrodes. However, this approach has limitation in the form of several minutes of preparation time to mount the electrodes. The conventional wet electrode characteristics have been studied to a great extent and stated in detail with their applications in [25]. This method uses

conductive gel which inevitably leaves residues on the scalp, short circuit may take place because of leak out of gel, and electrode placement is time-consuming, uncomfortable, and painful for the subject. Repeated use of gel may also induce allergic reactions or infections. Also, the signal quality may deteriorate over time period as the skin regenerates and/or the conductive gel dries. To overcome all these problems of conventional wet EEG electrodes, a modern fabrication system called Microelectromechanical Systems (MEMS) is used to design and fabricate dry MEMS electrodes to acquire EEG signals [26]. The EEG signal acquisition process [27] [28] of recording the brain's spontaneous electrical activity usually requires a typical time period of 20–40 minutes per run, by mounting multiple electrodes (sensors) at particular locations on the scalp. The electrodes are also categorised as active and passive. The passive electrode collect signal, which is amplified for further feature extraction and translation. From sensing to amplification, voltage drop occurs as the signals are in μv range, and leads to loss of information. The active electrodes collect signals and at the same time it's amplified which provide a good strength of signal for the further process.

B. Signal Processing and Classification

In all non-invasive EEG acquisition process, the input signals are always recorded with some unwanted data i.e. noise. Electronics interference, electromyography (EMG) signal evoked by muscular activity and ocular artifacts (Eye blinking) also gets recorded along with EEG. These unwanted components may lead to wrong conclusion, so the acquired input signals must be pre-processed. Signal processing can be done in time domain (e.g. neuron firing rates or evoke potential) or frequency domain (e.g. mu or beta rhythm amplitude) [29] or one can take advantage of both to improve the performance [30]. Following are some of the common pre-processing techniques for filtering noise.

Basic Filtering- Bandstop filter passes most frequencies unaltered but attenuates in the range of 50 to 60 Hz. However in the electromyogram (EMG) signal and ocular artifacts affect a large frequency band, as well as their spectrum, may vary over the time. So bandstop filters are not effective to arrest such artifacts. Useful information within the EEG lies within specific frequency bands; 4-8 Hz (theta band), 8-10 Hz (alpha-1 band), 10-20 Hz (alpha-2 band), 12-30 Hz (beta band) and 30-100 Hz (gamma band) [31]. Bandpass filter usually extracts such frequency where the specific range of frequency is allowed to pass and rejects all the other frequencies.

Adaptive Filtering- The frequency band within which the artifacts or unwanted signals lay is unknown most of the time and hence applying bandpass or bandstop filters won't always work. The noise filter should adapt to the spectrum of an input signal, and it should attenuate the input signal in frequency ranges that mostly contains these artifacts. In addition to this, the EOG or EMG has a strong correlation with measurement, which can be used to design adaptive filter

Blind Source Separation- EEG signals can be described to a good approximation by a fine set of source, identified within the brain which creates certain components of EEG. EOG and EMG signals can also be incorporated into the analysis. As assumption one can remove artifacts generated by a subset of the external sources and reconstruct EEG. Unwanted data (signals) must be arrested using adaptive filtering.

Independent Computer Analysis (ICA) decomposes the noise to statistically independent components. It is assumed that the number of a source is equal to the number of electrodes (sensors), input signals are statistically autonomous and there is no time delay between input and the place of measurement. There are two major steps of ICA, Centering and Whitening [32]. Centering subtracts the mean values of the signals whereas Whitening linear transformation results in uncorrelated signals with unit variance.

Matching Pursuit Algorithm decomposes the input signals into several components from the dictionary of Gabor [33]. EEG signals having a high correlation with this dictionary is selected and subtracted from it. The process is repeated till the source signals are represented by Gabor dictionary components. After comparing the performance of Self-Organizing Map and Multilayer Perceptron based neural classifier with the standard Bayes optimal classifier, Barreto *et al.* [34] suggested Welch's periodomances as an effective feature extracting pre-processing method. Furthermore, recurrent quantum neural network (RQNN) [35] based filtering and similar techniques are the future scope in filter design.

C. Computer Interface

The recorded EEG signal can be processed online or offline for feature extraction in conjunction with computer algorithms. The feature extraction of EEG signals may be amplitude or latencies of specific potentials (e.g. P-300), amplitude or frequencies of specific rhythms (alpha, theta, beta and delta) [36]. For effective and accurate BCI, it is necessary that the system extracts the proper feature of a signal and converts it into meaningful control. The translation algorithms translate these features into device command or messages which may be simple or complex neural networks. It must be able to accommodate, encourage and improve user's control. Signal processing techniques are essential but they cannot resolve all problems. The neurophysiological activation is for feature extraction while classifier and translation algorithm is obtained by offline analysis of previously recorded EEG data. These translation algorithms might use linear [37] or nonlinear methods [36] and convert independent variables (i.e. signal features) into dependent variable (i.e. device control commands).

The user adapts algorithms on three levels, first when user first-time access to adapt signal feature for example if the signal feature is mu-rhythm amplitude, the algorithm adjusts to user range of mu-rhythm amplitude. It adjusts to user characteristic P300 if the input is P300 amplitude. If the feature is firing single neuron than it adjust to neuron's distinguish firing range. However, the EEG signal affected by hormonal levels, immediate environment, fatigue, events, illness, short and long term variation to time so BCI need the second level of adaptation. The second level of algorithms takes care of this spontaneous variation to map user signal

input value to device command value. BCI is the effective interaction of user brain and control device, the above to levels are not addressing this. The third level of adaption accommodates and engages the adaptive capacity of the brain. If the feature is mu-rhythm amplitude then the relationship between that amplitude and the user's intent will increase, the third level rewarding faster communication by accommodating this. A proper design of the third level of adaption is important for the BCI growth.

III. BCI CHARACTERISTIC

A. *Dependent and independent BCIs*

BCI is also categorized as dependent BCI and independent BCI. In dependent BCI, the brain's normal pathways are not used to carry messages, but activity in the process is used to generate brain activity (EEG). In this process, the brain output is EEG, but the generation of EEG depends on the projected direction. A dependent BCI is another method for detecting messages carried out in brain's normal output pathway. In contrast, by any way brain's normal output pathways are related to an independent BCI. In an independent BCI, the normal output pathways of peripheral nerves and muscles do not have any role. Independent BCIs are more likely to be used for those people with most severe neuromuscular disability who lack in all normal output channels of communication.

B. *Skill in BCI*

To understand speculation of mind, simple BCI evolves mind reading or wiretapping analogy with a goal to reflect brain thought process by EEG signal. Electrophysiology signals are the mere reflection of central nervous system (CNS) which produces the signal that commands the external world. It changes a signal such as EEG rhythms and neural firing rate for control. The brain neuromuscular out channels depends on the successful operation on feedback, such as dancing, singing, walking, lifting and touching etc. In the absence of feedback from a start, motor skill does not develop properly moreover if feedback lost later on, skills deteriorates. A BCI system must be provided with feedback and it must be incorporated with brain to makes it responsive for that feedback, this indicate BCI system deals with two adaptive controllers: one is brain and other is BCI itself. So successful BCI system depends on user's skill of generating EEG signals which consist of proper muscle control at the same time it's required that BCI translate that control into output which accomplish user's intent. In the independent BCI, the response generated towards the desired letter without any initial training in P300. The adaptive modification undergoes, once this P300 engaged with the communication channel, for successful BCI recognition and productive engagement it's important. The brain adaptive capacity can be extended to control various electrophysiological signal and to increase an amplitude of a specific EEG feature.

C. *The mental strategy*

The controlling of computer peripherals is a learning process with the aim of self-controlling brain cortical potential shifts or brain sensorimotor rhythms with the help of suitable feedback. The process does not dependent on continuous feedback but seeking desired brain potential. The researcher applied an operant condition to understand the communication system for the locked-in patient. The motor imagery is another mental strategy. The researcher concludes this motor imagery trigger cortical areas similar to those triggered by executing the same movement. To concentrate on flickering lights, flashed letters, mental arithmetic and imaging the rotational objects are the mental task besides motor imagery are appropriate to modulate the brain signals.

IV. BRAIN-COMPUTER INTERFACE: CHALLENGES

Development of BCI depends on selection of signals, data acquisition methods, feature extraction methods, translation algorithms, output devices, depended/ independent modes, synchronous/asynchronous mode, development of training strategies, protocols, choice of application and user group. During movement proprioceptive and another sensory feedback occurs as part of cortical and subcortical neuron activity. Without actual movement, it is still not clear that up to what extent users can produce this activity and other sensory mobility. With long-term stability whether neuronal activity can function without movement is yet to be established. Among the users, an ability to use BCIs and best choice of BCIs are likely to differ. For long-term assessment of performance, it is required to evaluate what specific BCIs are needed in specific user groups. The performance of BCIs depends on its signal to noise ratio, and also on a variety of options for improving the signal to noise ratio. Major work is so far limited to offline analysis, research seeking online analysis with the comparison of alternative methods. Translation algorithms convert independent variables (user input) into dependent variable (control output), a success of these algorithms depends on appropriate selection of signal features. Deep study is still required in several of BCI including signal preprocessing, feature extraction, translating algorithm, user interfaces etc.

V. TOOLS IN BCI RESEARCH

Several powerful numerical data analysis tools for BCI research are available such as Matlab, Octave and Scilab for offline and online mode. Offline mode is not only to determine reliable and stable classifiers, but also to find frequency range, feature extraction method's window length etc. Matlab is one of the most powerful tools incorporating ANN, signal processing, fuzzy logic, and other mathematical analysis tools on a single platform. BIOSIG [38] is a free and open source library initiated in the year 2003 for biomedical signal analysis. BioSig library is available for Octave and Matlab as well as C/C++. For annotated electrocardiogram data, Federal Drug Administration and Health Level Seven, a healthcare standard group has proposed XML-based data format. With biosig4c++ data can process with Matlab and Octave. The BioSig provide

common interface to the 50 data formats and 25 supported by biosig4octmat while the most common format in BCIs are BCI2000, BioSemi Data Format (bdf), General Data Format (gdf), BrainVision and other Matlab files [38].

To detect EMG artifacts, BioSig offers several tools as well as fully automated methods to reduce ocular artifacts. To implement real-time BCI system within BioSig framework, rtsBCI modules are available which are also suitable for rapid prototyping [39]. To efficiently compute and update new classifier, rstBCI module can be integrated with Matlab, it contains the function to correct for ocular and facial muscle artifacts also estimate band power and Adaptive Autoregressive (AAR)[6] parameter and control a virtual environment [38]. In BCI research software development is a crucial concern. The software can provide similarities and dissimilarities in data processing methods and its characteristic also makes clear hyperparameters for particular algorithms. With BioSig, BCI researchers avoid reinvention and duplication for every project.

VI. APPLICATION

Robotics prosthetic or an exoskeleton with brain control are the applications related to mobility. A first BCI to restore full body mobility to patients with severe paralysis, Walk Again Project is under development. With the direct communication with mouse, monitor and keyboard, BCI has potential to insert a user into the virtual world, the EPC-Emotive device is an example of this [40]. Work to create speech using a voice synthesizer has been presented by Brumberg in 2011 [8]. In addition to daily activities, environment control, locomotion and exercise are five potential applications of BCIs. Asynchronous EEG analysis and machine learning techniques are used in mobile robot control with non-invasive brain activity. Abdullah *et al.* [41] presented an interface for navigating a mobile robot that moves at a fixed speed in planar workspace. R. Bickford [42] discussed the research and clinical applications of the EEG in humans and animals such as Brain monitoring, Detection of brain injury, stroke and tumour, Monitor alpha rhythm, Control anesthesia depth, Investigate epilepsy, Observation of neuromuscular behaviour of epilepsy drug, Observation of human and animal brain growth etc. To assist a person on mobility, communication and interaction with the surrounding many applications have been developed so far for the lock-in patient which happens to be the main reason of BCI development. The area of development are Neuro-prosthetics, [43] robotics wheelchair, autonomous car, mental disturbance treatment, voice synthesizer and mobile devices.

VII. CONCLUSION

This survey focused on BCI components in general, method of signal acquisition, feature identification and acquisition by the various algorithm, challenges in BCI, tools used in BCI research and application. BCI is the worship for physically disabled people, especially locked-in people who cannot use the brain's normal output pathways and muscle movement. BCI techniques vary as per application as well as require different algorithms for feature identification of the pre-processed EEG signals and controlling peripheral devices. The work presented evaluation and current trends in BCI. In the present scenario, EEG signals can be recorded by invasive and noninvasive methods. The invasive method records the EEG signal from inside the grey matter of a brain while noninvasive records outside the grey matter of a brain. Noninvasive methods such as EMG, fMRI and NIRS are more popular and user-friendly as it does not require surgical implantation. Microelectromechanical system and Nanotechnology-based invasive methods are open research problems in BCI. In addition, work also needs to be done to improve information transfer rate for smooth control of peripheral devices.

Non-invasive BCI records the brain signals with inherently embedded noise which may include electromyography, signal evoked by muscular activity, eye blinking etc. These unwanted components can be filtered out with appropriate filters and algorithms through several platforms as GNU Octave, FreeMat, Scilab and MATLAB tools. Appropriate features can only be extracted if the input signal is a true estimation of the thought process. Therefore, there is a lot of scope in the field of BCI research, which begins right at the state of EEG acquisition itself. The BCI application such as light and television control, yes/no questions, text processing, wheelchair control, robotic prosthetics, autonomous vehicles, auto calling using brain activity and virtual reality games has extended the boundaries of research and become a truly inter-disciplinary field for neuroscientists, engineers, psychologists, computer scientists and many more.

[6] A parameter required to extract specific frequency band in autoregressive spectral analysis

REFERENCES

[1] G. Pfurtscheller and C. Neuper, "Motor imagery activity primary sensorimotor area in humans," *Neurosci. Lett,* vol. 239, no. 2-3, pp. 65-68, 1997.

[2] V. Gandhi, Brain-computer Interfacing for Assistive Robotics: Electroencephalograms, Recurrent Quantum Neural Networks, and User-Centric Graphical Interfaces, AcademicPress, 2014.

[3] A. Guyton and J. Hall, Textbook of medical physiology, Philadelphia: Elsevier Saunders, 2006.

[4] J. Wolpaw, N. Birbaumer, D. McFarland, G. Pfurtscheller and T. Vaughan, "Brain-Computer Interface for Communication and Control," *Clinical Neurophysiology,* vol. 113, no. 6, pp. 767-791, 2002.

[5] V. Gandhi, G. Prasad, D. Coyle, L. Behera and T. M. McGinnity, "EEG-based mobile robot control through an adaptive brain-robot interface," *IEEE Transaction on System Man & Cybernetics: Systems,* vol. 44, no. 9, pp. 1278-1285, 2014.

[6] V. Gandhi, G. Prasad, D. Coyle, L. Behera and T. M. McGinnity, "Evaluating Quantum Neural Network filtered motor imagery brain-computer interface using multiple classification techniques," in *Neurocomputing*, Elsevier, 2015.

[7] R. Scherer, G. Muller, B. Graimann and G. Pfurtscheller, "An asynchronously controlled EEG-based virtual keyboard: improvement of spelling rate," *IEEE Transactions Biomedical Engineering,* vol. 51, pp. 979-984, 2004.

[8] F. H. Guenther and J. S. Brumberg, "Brain-machine interfaces for real-time speech synthesis," in *33rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Boston, MA, U.S, 2011.

[9] H. Berger, "Uber das Elecrtenkphalogramm des Menchen," *Arch*

Psychiat Nervernkr, vol. 87, pp. 527-570, 1929.

[10] E. Adrian and B. Matthews, "The interpretation of potential waves in the cortex," *Journal of Physiology,* vol. 81, pp. 440-471, 1934.

[11] E. Adrian and K. Yamagiwa, "The origin of the Berger rhythm," *Brain,* vol. 58, p. 323–351, 1935.

[12] N. Birbaumer, W. Heetderks, J. Wolpaw, W. Heetderks, D. McFarland, P. H. Peckham, G. Schalk, E. Donchin, L. Quatrano, C. Robinson and T. Vaughan, "Brain-Computer Interface Technology: A Review of the First International Meeting," *IEEE Transaction on Rehalititation Engineering,* vol. 8, no. 2, pp. 164-173, 2000.

[13] P. Meinicke, M. Kaper, F. Hoppe, M. Heumann and H. Ritter, "Improving Transfer Rates in Brain-Computer Interfacing: A Case Study," in *Proceedings of the Advances in Neural Inf. Proc. Systems*, Vancouver, 2002.

[14] S. Haykin, Neural Networks: A comprehensive Foundation, NJ, USA: Prentice Hall, second edition.

[15] K. Eric, J. Schwartz and T. Jessell, Principles of Neural Science, USA: McGraw-Hill, Fourth Edition.

[16] N. K. Cauvery, G. LINGARAJU and H. Anupama, "Brain-Computer Interface and its types-A Study," *International Journal of Advances in Engineering & Technology,* vol. 3, no. 2, pp. 739-745, 2012.

[17] A. Elghrabawy and M. Wahed, "Prediction of Five-Class Finger Flexion Using ECoG," in *Cairo International Biomedical Engineering Conference*, Cairo, Egypt, 2012.

[18] E. Leuthardt, G. Schalk, J. Wolpaw, J. Ojemann and D. Moran, "A brain–computer interface using electrocorticographic signals in humans," *Journal of Neurual Engineering,* vol. 1, pp. 63-71, 2004.

[19] G. Schalk, K. Miller, N. Anderson, J. Wilson, M. Smyth, J. Ojemann, D. Moran, J. Wolpaw and E. Leuthardt, "Two-dimensional movement control using electrocorticographic signals in humans," *Journal of Neural Engineering,* vol. 5, no. 1, pp. 1741-2560, 2008.

[20] P. Ahmadian, S. Cagnoni and L. Ascari, "How capable is non-invasive EEG data of predicting the next movement? A mini review," *Frontiers in Human Neuroscience,* vol. 7, no. 124, pp. 1-7, 2013.

[21] J. E. Zimmerman, "SQUID instruments and shielding for low-level magnetic measurements," *Journal of Applied Physics,* vol. 48, no. 2, pp. 702-710, 1977.

[22] D. Cohen, "Magnetoencephalography: evidence of magnetic fields produced by alpha rhythm currents," *Science,* vol. 161, pp. 784-786, 1968.

[23] K. A. Moxon , A. Melisiotis and G. Foffani, "Functional Changes in Sensorimotor Regions of the Brain Following Spinal Injury," in *2nd International IEEE Conference on Neural Engineering*, Arlington, VA, 2005.

[24] R. Gilmore and J. Clin, "American Electroencephalographic Society guidelines in electroencephalography, evoked potentials, and polysomnography," *Journal of Clinical Physiology,* vol. 11, p. 147, 1994.

[25] Y. M. Chi, S. Diego and J. Tzyy-Ping, "Dry-contact and noncontact biopoential electrodes: Methodological review," *Biomedical Engineering, IEEE Reviews,* vol. 3, pp. 106-119, 2010.

[26] L.-D. Liao, I.-J. Wang, S.-F. Chen, J.-Y. Chang and C.-T. Lin, "Design, Fabrication and Experimental Validation of a Novel Dry-Contact Sensor for Measuring Electroencephalography Signals without Skin Preparation," *sensors,* vol. 11, no. 6, pp. 5819-5834, 2011.

[27] G. Pfurtscheller, C. Neuper, C. Guger, W. Harkam, H. Ramoser, A. Schlögl, B. Obermaier and M. Pregenzer, "Current Trends in Graz Brain-Computer interface (BCI) Research," *IEEE Transactions on Rehabilitation Engineering,* vol. 8, no. 2, pp. 216-219, June 2000.

[28] W. Penny, S. Roberts, E. Curran and M. Stokes, "EEG-based communication: a pattern recognition approach," *IEEE Transactions on Rehabilitation Engineering,* vol. 8, no. 2, pp. 214-215, June 2000.

[29] M. Polak and A. Kostov, "Parallel man-machine training in development of EEG-based cursor control," *IEEE Transaction Rehabilitation Engineering,* vol. 8, no. 2, pp. 203-205, 2000.

[30] G. Schalk, J. Wolpaw, D. McFarland and G. Pfurtscheller, "EEG-based communication: presence of an error potential," *Clin Neurophysiol,* vol. 111, no. 12, pp. 2138-2144, 2000.

[31] L. F. Nicolas-Alonso and J. Gomez-Gil, "Brain-Computer Interfaces, a Review," *Sensor,* vol. 12, no. 2, pp. 1211-1279, 2012.

[32] A. Kachenoura, L. Albera, L. Senhadji and P. Comon, "ICA: a potential tool for BCI systems," *Signal Processing Magazine, IEEE,* vol. 25, no. 1, pp. 57-68, 2008.

[33] Z. Lwin and M. Thaw, "Mental Tasks Classification from Electroencephalogram (EEG) Signal Using Gabor Based Matching Pursuit (MP)," *International Journal of Computer Science And Technology,* vol. 6, no. 1, pp. 22-26, 2015.

[34] G. Barreto, R. Frota and F. Medeiros, "On the classification of mental tasks: a performance comparison of neural and statistical approaches," *In proceedings of IEEE Workshop on machine learning for Signal Processing,* pp. 529-538, 2004.

[35] V. Gandhi, G. Prasad, D. Coyle, L. Behera and T. M. McGinnity, "Quantum Neural Network-Based EEG Filtering for a Brain-Computer Interface," *IEEE Trans. on Neural Network and Learning System,* vol. 25, no. 2, pp. 278-288, 2014.

[36] F. Lotte, F. Lamarche, A. L´ecuyer, M. Congedo and B. Arnaldi, "A review of classification algorithms for EEG-based brain-computer interfaces," *Journal of Neural Engineering,* vol. 4, no. 2, pp. 1-25, 2007.

[37] A. Jain, P. W. Robart and J. Mao, "Statistical Pattern Recognition: A Review," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 22, no. 1, pp. 4-37, 2000.

[38] A. Schlog and C. Brunner, "BioSig: A Free and Open Source Software Library for BCI Research," *Computer,* vol. 41, no. 10, pp. 44-50, 2009.

[39] C. Gurger, A. Schlogl, C. Neuper, D. Walterspacher, T. Strein and G. Pfurtscheller, "Rapid Prototyping of an EEG-Based brain-computer Interface (BCI)," *IEEE Transaction on Neural Systems and Rehabilitation Engineering,* vol. 9, no. 1, pp. 49-58, March 2001.

[40] J. Wolpaw, D. McFarland, G. Neat and C. Forneris, "An EEG-based brain-computer interface for cursor control," *Electroencephalography and clinical Neurophysiology,* vol. 78, pp. 252-259, 1991.

[41] A. Akce, M. Johnson, O. Dantsker and T. Bretl, "A Brain–Machine Interface to Navigate a Mobile Robot in a Planar Workspace: Enabling Humans to Fly Simulated Aircraft With EEG," *IEEE Transactions on Neural Systems and Rehabilitation Engineering,* vol. 21, no. 2, pp. 306-318, march 2013.

[42] R. D. Brickford, "Electroencephalography," in *Encyclopedia of Neuroscience*, Birkhauser, Cambridge (USA), 1987, pp. 371-373.

[43] J. N. Mak, Y. Arbel, J. W. Minett, L. M. McCane, B. Yuksel, D. Ryan, D. Thompson, L. Bianchi and D. Erdogmus, "Optimizing the P300-based brain–computer interface: current status, limitations and future directions," *Journal of Neural Engineering,* vol. 8, pp. 1-7, 2011.

[44] J. Walters-Williams and L. Yan, "A New Approach to Denoising EEG Signals-Merger of Translation Invariant Wavelet and ICA," *International Journal of Biometrics and Bioinformatics,* vol. 5, no. 2, pp. 130-148, 2011.

[45] P. Griss, P. Enoksson, Tolvanen-Laakso, P. Merilainen, G. Stemme and S. Ollmar, "Micromachined electrodes for biopotential measurement," *IEEE Microelectromechanical Systems,* vol. 10, no. 1, pp. 10-15, 2001.

[46] Y. Yan, . N. Mu, D. Duan, D. Linguo, X. Tang and T. Yan, "A dry electrode based headband voice brain-computer interface device," *IEEE International Conference on Complex Medical Engineering,* pp. 205-210, 2013.

[47] L. Simon and H. Jane, "A direct brain interface based on event-related potentials," *IEEE Transactions on Rehabilitation Engineering,* vol. 8, no. 2, pp. 180-185, 2000.

Data Streams and Privacy:
Two Emerging Issues in Data Classification

Radhika Kotecha
Assistant Professor
Department of Information Technology
V.V.P. Engineering College, Rajkot, India
kotecha.radhika7@gmail.com

Sanjay Garg
Professor and Head
Department of Computer Science and Engineering
Nirma University, Ahmedabad, India
garsgsv@gmail.com

***Abstract*— Several real-world applications generate data streams where the opportunity to examine each instance is concise. Effective classification of such data streams is an emerging issue in data mining. However, such classification can cause severe threats to privacy. There are several applications like credit card fraud detection, disease outbreak or biological attack detection, loan approval, etc. where the data is homogeneously distributed among different parties. These parties may wish to collaboratively build a classifier to obtain certain global patterns but will be reluctant to disclose their private data. Privacy-preserving classification of such homogeneously distributed data is a challenging issue too. In this paper, we present a brief review of the work carried out in data stream classification and privacy-preserving classification of homogeneously distributed data; followed by an empirical evaluation and performance comparison of some methods in both these areas. We also propose and evaluate an approach of creating an ensemble of anonymous decision trees to classify homogeneously distributed data in a privacy-preserving manner. We further identify the need to develop efficient methods for privacy-preserving classification of homogeneously distributed data streams and propose a suitable approach for the same.**

Keywords- Data Streams Classification; Privacy-Preserving Classification; Privacy-Preserving Classification of Homogeneously Distributed Data Streams

I. INTRODUCTION

With technological advancements, several applications generate a vast amount of data. Such data, also known as data streams, are characterized as continuously arriving at unprecedented rates [1, 2]. Examples of data streams include internet traffic streams, stock trading streams, streams generated by e-commerce sites, data generated from scientific projects, supermarket data, multimedia data, medical data streams, data streams generated through industry production processes, remote sensor and video surveillance streams, etc. As these data streams contain valuable knowledge, there is an enormous demand for analyzing and mining them.

Since last few decades, the data mining techniques have been successfully applied to several real-world problems. However, these traditional data mining methods are not feasible for applications generating stream data as they require the data to be first stored and then perform multiple scans on that data in order to mine it. But, storing the data streams consistently and reliably is not possible as its daily volume may possibly range in the millions. Even if the storage is possible, gathering the infinite streams at one time, in one place, in a format suitable for mining, and performing multiple scans on it is not cost-effective [3]. Extracting patterns and models from such continuous data streams is called data stream mining.

In recent years, data stream classification has been an active area of research in data stream mining due to its several real-world applications. Some examples of such applications are: e-mail spam detection, credit card fraud detection, malicious web page detection, intrusion detection, detection of any abnormal disease outbreaks from continuous streams of data arriving at hospitals, etc. Predicting class label of the incoming data streams, that is, data stream classification is an emerging issue and is discussed in the paper.

Further, advancement in networking technologies has triggered mining of distributed data. Different organizations (data holders) want to undertake a joint data mining task to obtain certain global patterns. Such collaboration is essential because of the mutual benefits it brings. However, free sharing of data is restricted due to privacy and security concerns. For e.g., Health Insurance Portability and Accountability Act (HIPAA) laws [2] require that medical data should not be released without appropriate anonymization. Similar constraints arise in many applications. Hence, such collaborating parties are reluctant to disclose their private data sets to one another. The problem of preserving the privacy of individuals while mining such distributed data is a challenge and is known as privacy-preserving distributed data mining.

The need of privacy-preserving distributed data mining exists in several domains. For example, several pharmaceutical companies want to collaboratively mine their data and perform genetic experiments on it to discover meaningful patterns among genes [4]. Multiple competing supermarkets wish to conduct data mining on their joint dataset because the result of this collaboration will be valuable to them [5]. The field of homeland security with an aim to counter-terrorism depends on collaboration and information sharing across different mission areas and nations [5]. Sharing healthcare and consumer data also enables study and early detection of a disease outbreak.

Recent developments in the field of privacy-preserving data mining focus on these two major subjects: preserving the privacy of homogeneously distributed data and preserving the privacy of heterogeneously distributed data. Homogeneously distributed data or horizontally partitioned data are the terms used when different sites collect similar kind of data over different individuals. Heterogeneously distributed data or vertically partitioned data refers to data collected by different sites on same individuals but with different feature sets.

The issue of privacy-preserving data classification has emerged to address several problems abound in various diverse domains. The goal is to build accurate classifiers without unveiling the privacy of the data being mined [6, 7]. We focus on homogeneously distributed data [8, 9] as numerous applications fall under this data model. For example, several banks collect transactional information for credit card

978-1-4799-9991-0/15/$31.00 ©2015 IEEE

customers where the features collected, such as age, gender, balance, average monthly deposit, etc. are the same for all banks [8]. Identifying whether a particular transaction is fraudulent or not is a problem called "privacy-preserving classification of homogeneously distributed data" where the privacy of customers' data needs to be protected.

The rest of the paper is organized as follows: The next section briefly describes the work accomplished in data streams classification and privacy-preserving classification of homogeneously distributed data. Section III presents the empirical evaluation and comparison of some data stream classification methods as well as the techniques for privacy-preserving classification of homogeneously distributed data. A need and a suitable approach for privacy-preserving classification of homogeneously distributed data streams in proposed in Section IV. Section V concludes the paper.

II. Related Work

In this section, we present the existing methods of data streams classification and privacy-preserving classification of homogeneously distributed data.

A. Data Streams Classification

Conventional classification techniques include three phases: a training phase that uses labeled data to train a classifier model; a test phase that uses previously unseen data to test the model; and a deployment phase wherein the model is applied to classify the unlabeled data [10]. On the other hand, data stream classification [2, 11, 12] consists of only a single stream of data, in which labeled and unlabeled data are mixed collectively within the stream [12, 13]. Hence, the training, testing and deployment phases have to be interleaved [14]. Further, most existing algorithms assume that data streams come under stationary distribution, where the concept of data remains unchanged. When the underlying class (concept) of the data changes over time, it is referred as concept drift, which is quite frequent in real-world applications.

In case of concept drift, there is a quandary: whether to update the classifier often and waste resources on changes that might be momentary (or insignificant) or else to update the model occasionally (which may result into degradation of the accuracy of the classifier). There are three algorithmic approaches in order to tackle this quandary: 1) periodic approach; where the classifier is rebuilt periodically, 2) incremental approach; where the classifier is updated with every concept drift and 3) reactive approach, where changes are monitored and the classifier is rebuilt only if it no longer matches the underlying data. Each of this algorithmic approach has its own benefits and limitations. The periodic approach is simple and has fixed communication and computation cost but wastes resources when there is no concept-drift. The incremental approach is accurate and efficient, but immediately updating the classifier might be a waste of resources if the drift is momentary. The reactive approach is suitable if monitoring of the classifier's match with the incoming data stream is done accurately and efficiently. Updating the classifier seldom will save resources, and rebuilding the classifier when it does not suit the data any longer will maintain the accuracy.

A decision tree [15] induction method named Very Fast Decision Tree (VFDT) that is capable of learning from high-speed data streams is proposed in [16]. VFDT is based on the fact that it may be adequate to use only a small subset of the available examples in order to choose the best splitting attribute at any given node. To decide the number of examples needed at each node, VFDT uses a statistical result called Hoeffding bound [17]. Consider r is real-valued random variable with range R and after n independent observations of this random variable, $\bar{r}$ is its mean. The Hoeffding bound ensures with a probability $1 - \delta$ that the true mean of r is at least $\bar{r} - \varepsilon$. Equation (1) depicts the Hoeffding bound.

$$\varepsilon = \sqrt{\frac{R^2 \ln(1/\delta)}{2n}} \qquad (1)$$

Let G be the heuristic measure used to choose the splitting attribute at a node. If G is information gain, its range R is $\log_2$(#classes). After observing n examples and assuming G is to be maximized, let x_a be the attribute with highest $\overline{G}$, x_b be the attribute with second-highest $\overline{G}$ and $\Delta\overline{G} = \overline{G}(x_a) - \overline{G}(x_b)$ be the difference between the two attributes. After observing n examples at any node, if $\Delta\overline{G} > \varepsilon$, the Hoeffding bound guarantees with a probability $1 - \delta$ that the true $\Delta G \geq 0$ and x_a is indeed the best splitting attribute at that node. VFDT produces decision tree classifiers that are nearly similar to the tree produced by a conventional batch learner.

Several variants of VFDT address the issue of concept drift. For example, the method Concept-Adapting Very Fast Decision Tree (CVFDT) [3] works by maintaining a model consistent with respect to a sliding window of instances from the data stream. It creates an alternate subtree when it detects that the distribution of data is changing at a node, but this subtree replaces the old tree only after it becomes accurate.

The disadvantage of CVFDT is that it does not define any optimal window size. A small window reflects accurately the current distribution whereas a large size accommodates several examples to work on and increases accuracy when the concept is stable. This disadvantage is overcome by a technique named Hoeffding Adaptive Tree using Adaptive Windowing (HAT-ADWIN) [18]. This method keeps a variable-length window of recently seen items which automatically grows when no change is apparent and shrinks it when data changes.

A general framework for classification of data streams with concept-drift is proposed in [19]. Instead of constantly modifying a single classifier model, the framework proposes to train a weighted classifier ensemble from sequential data chunks partitioned from the stream. A classifier is learned for each chunk and the weight of classifier is inversely proportional to its expected prediction error. The paper shows that classifier ensembles outperform single classifiers while classifying concept-drifting data streams.

Most of the traditional data stream classification techniques assume a fixed number of classes. But in real-world, novel classes may emerge at any time in the data stream, which remain undetected by these traditional data stream classification methods until a classifier is trained with the novel classes. A data stream classification method proposed in [20] automatically detects a novel class. It is assumed that the instances that belong to the same class should appear closer to each other (called cohesion) and should be distant from the instances that belong to other classes (called separation). Any test instance that is found to be separated from the training data

may probably be an instance of a novel class. However, a novel class is assumed to be detected only if an adequate number of such instances that exhibit strong cohesion appear in a stream.

Several real-world applications such as intrusion detection, disease outbreak detection, credit card transaction fraud detection, etc. generate imbalanced data streams where the number of data instances from one class is quite less as compared to the other class. Correctly classifying such minority class examples is a major issue. Like [20], in [21] too, the incoming data stream is divided into chunks, a classifier is trained from each chunk and an ensemble of these classifiers is created. Here, the imbalanced data streams are classified by accumulating minority class instances from previous data chunks and adding them into the current training chunk. Using previous minority class instances give a benefit over the traditional method of synthetically creating such examples.

Naïve Bayes classifier can be adopted for Data Streams in its original form [22] and requires only maintaining a statistics table. For discrete valued attributes, the only class label counts per attribute value are required. Continuous numeric attributes can be discretized a priori. An attribute with *m* unique attribute values and *n* possible classes can be stored in a table with *mn* entries only. On arrival of new training instances in the stream, tables are updated by merely incrementing the appropriate entries as per attribute values and class.

Empirical evaluation and comparison of some of these methods is shown in section III.

B. Privacy-Preserving Classification of Homogeneously Distributed Data

The goal in privacy-preserving classification is to build accurate classifiers without unveiling the privacy of the data being mined [7]. Randomization approach has been widely used for privacy-preserving classification [6, 23]. Initially, in randomization approach, the data providers randomize the data and transmit it to the data miner, i.e. a sufficiently large noise is added to the data with a goal that the true values of the records cannot be recovered. Further, a distribution reconstruction algorithm that reconstructs the original data distribution from the randomized data instances is employed at the central site. Several distribution reconstruction algorithms like EM and Bayes reconstruction method have been used in the literature. Finally, a classifier is constructed from the reconstructed data. The approach has the advantage of simplicity, but it lacks a formal framework that proves how much privacy is assured.

In [24], an ensemble method is used to perform privacy-preserving distributed data classification among sites. When the data is homogeneously distributed, each site constructs a decision tree classifier from the local data available with it, and a central trusted party integrates these results by producing a classifier ensemble. This ensemble is used by all the sites to classify the unseen data. The paper empirically evaluates the proposed method on two medical datasets.

A large portion of literature in privacy-preserving data classification discusses the application of cryptographic techniques and Secure Multiparty Computation (SMC) for building classifiers in a privacy-preserving manner [25, 26]. For example, in [27], the training set is considered to be homogeneously distributed between two parties and the authors attempt to securely construct an ID3 decision tree. In such techniques, secure log algorithm, secure sum, etc. sub-protocols are used to securely calculate the conditional entropy for an attribute shared by the two parties and a classifier using ID3 is built securely. Classifiers so created resemble the classifier induced if the data is centrally accumulated. Although such methods are secure enough, they demand a lot of computation and the cost of communication is also high.

In [28], a privacy-preserving decision tree learning method based on the ID3 algorithm and Shamir's secret sharing is proposed for homogeneously distributed data. Shamir's secret sharing method operates in three phases and is used to compute the summation of the secret values over *n* parties without revealing the secrets to other parties. In the first phase, each party has a secret value and they choose a random polynomial of degree *n-1*. The constant term in the polynomial is the secret value. Also, each party creates a random number of its own and reveals it to others. Further, using Shamir's secret sharing algorithm, each party computes the share of all other parties based on the random numbers revealed by them and sends the respective shares to all the parties. In the second phase, each party performs the summation of the shares it obtains from other parties and sends this intermediate result to all parties. In the final phase, each party solves the set of equations to find the sum of secret values using the intermediate results received from the second phase. Hence, this method lets all the parties to obtain the conditional entropy of each attribute from the data at all the parties. Finally, from this conditional entropy, the best attribute at a node can be determined and the tree can be induced. This method is scalable up to a large number of parties but suffers from minor information leakage issues.

In [29], a framework with a multi-round algorithm is proposed for classification of homogeneously distributed data using privacy-preserving k-Nearest Neighbor (kNN) classifier. In case of distributed environment, an instance's *k* nearest neighbors may be distributed among several nodes. That is, each node will contain a few data tuples that are *k* nearest neighbors of each query instance. Hence, the classification process is divided into two steps: In the first step, the tuples in the database at each node that belong to *k* nearest neighbors of the query instance *q* (locally) are selected. Further, a privacy-preserving algorithm is applied to identify *k* nearest neighbors between the tuples in the union of the databases and query instance *q* (globally). In the second step, each node classifies *q* locally and all the nodes cooperate to determine the classification of *q* globally, in a privacy-preserving way. Higher the value of *k*, more the privacy is protected.

K-anonymization techniques [30] have been widely used for privacy-preserving data mining and ensure that individuals cannot be uniquely distinguished by linking attacks. That is, the technique performs suppression or generalization on the microdata in a way that any record in the dataset is indistinguishable from at least $(k - 1)$, $k \geq 1$ other records within the same data set.

In [31], a method for directly building a k-anonymous decision tree from a private table has been proposed. Here, both mining and anonymization are carried out in a single process. In the beginning, the decision tree consists of only the root representing all the tuples in a private table. At any given stage of induction, while splitting a node in the tree, the algorithm selects the attribute in the quasi-identifier with the

highest gain (considering Information Gain or Gini Index as gain functions), only if the split does not violate k-anonymity. Whenever splitting a quasi-identifier causes a breach of anonymity, a generalized version of that attribute is selected as a potential candidate for splitting the node. The algorithm terminates when no further node can be inserted without compromising k-anonymity. The paper shows how this hybrid method of mining and anonymization together is better than doing the said tasks separately.

Empirical evaluation and comparison of some of these techniques is shown in next section.

III. Implementation and Results

In this section, tools and datasets used for evaluation are discussed, and comparative results of data stream classification methods, as well as techniques of privacy-preserving classification of homogeneously distributed data, are presented.

A. Data Stream Classification

Several experiments to compare the performance of the existing data stream classification techniques were conducted. Here, the goals were to evaluate the accuracies of classifiers; to compare the running times of the methods; to estimate their ability to deal with concept drift; to identify and characterize the situations when one classifier outperforms the other.

1) Data Streams

Experiments were performed on both synthetic and real-life data streams with the number of instances varying from lakhs to millions. Table I lists out the four data streams used along with their details. Forest Covertype and Waveform data streams are available on the UCI machine learning repository [32] whereas Loan Approval and Rotating Hyperplane are synthetically generated data streams introduced and used in [6] and [3] respectively. Further, Waveform and Rotating Hyperplane data streams have concept drifts.

TABLE I. Composition of Data Sets

Data Stream	No. of Attributes	No. of Instances	No. of Classes
Forest Covertype	54	5.8 lakhs	7
Waveform	40	1 million	3
Loan Approval	9	10 million	2
Rotating Hyperplane	10	10 million	2

2) Implementation Details

In our experiments, four data stream classification algorithms have empirically been evaluated and compared: Naïve Bayes Classifier, VFDT, HAT-ADWIN, and Accuracy Weighted Ensemble Classifier with Hoeffding Classifiers as base learners. All four techniques have been implemented in Massive Online Analysis (MOA) [33]. MOA is an open source framework for data stream mining that includes a collection of machine learning algorithms for evaluation. Batch learning uses holdout or cross-validation method to evaluate the performance of classifiers. Since data streams are continuously arriving, an alternative scheme called 'Prequential' which interleaves testing with training is used. In this method, each individual example can be used to test the classifier before it is used for training, and from this, the accuracy can be incrementally updated. Using this method, the model is always being tested on examples it hasn't seen.

3) Results

The results of predictive accuracy and running time of all four classifiers on the stated four data streams are tabulated in Table II and III.

TABLE II. Accuracy of Classifiers (in %)

Data Stream	Naïve Bayes	VFDT	HAT-ADWIN	Accuracy Weighted Ensemble
Forest Covertype	63.30	82.25	83.15	80.95
Waveform	81.05	81.21	84.50	84.35
Loan Approval	65.98	90.64	88.39	85.29
Rotating Hyperplane	81.06	83.57	91.65	91.99

TABLE III. Running Time of Classifiers (in seconds)

Data Stream	Naïve Bayes	VFDT	HAT-ADWIN	Accuracy Weighted Ensemble
Forest Covertype	48.28	46.27	71.85	2433.33
Waveform	41.17	64.62	249.57	1785.12
Loan Approval	81.20	234.33	888.77	4322.63
Rotating Hyperplane	129.22	247.75	865.07	8523.69

From the results, it can be seen that Naïve Bayes classifier is quick in training and evaluating all types of data stream instances. But it is the least accurate among all mentioned classifiers as it assumes that attributes are independent from one another. Further, it can be seen from the results that VFDT classifier shows remarkably good results in an acceptable time. But when the data streams have concept-drifts; as in Waveform and Rotating Hyperplane streams, the accuracy of VFDT falls.

Accuracy Weighted Ensemble classifier is accurate enough but with the increase in number of attributes or data instances, the time required to train and evaluate the stream is very high. HAT-ADWIN can be regarded as the best classifier among the four in terms of accuracy. However, the time taken by this approach is higher as compared to VFDT. But unlike VFDT, even in presence of concept drift, accuracy of HAT-ADWIN does not degrade. Further, as compared to Accuracy Weighted Ensemble classifier, the time required by HAT-ADWIN is less.

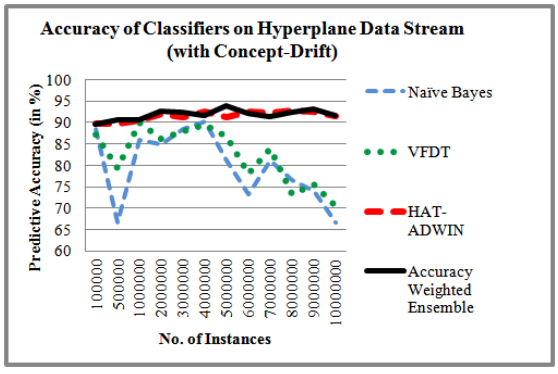

Fig. 1. Accuracies of Classifiers on Hyperplane Data Stream

The effect of concept drift on the performance of classifiers can be specifically observed in Fig. 1 which depicts the case of Rotating Hyperplane stream. It can be concluded from the results of Table II and III that VFDT is the most suitable algorithm in the absence of concept drift; whereas in its presence, HAT-ADWIN is preferable.

B. *Privacy-Preserving Classification of Homogeneously Distributed Data*

Several experiments were performed to compare the performance of different privacy-preserving classifiers where the data tuples are stored at multiple autonomous sites. Throughout the experiments, it was assumed that three parties want to collaboratively conduct the data classification.

1) Data Sets

For comparing the performances, experiments on four data sets from various real domains were conducted. These data sets are available on UCI machine learning repository [32] and its details are described in Table IV. All these data sets contain private information which is to be protected from disclosure.

TABLE IV. COMPOSITION OF DATA SETS

Data Set	No. of Attributes	No. of Instances	No. of Classes
Transfusion	5	748	2
Diabetes	9	768	2
Bank Marketing	8	4521	2
Spambase	57	4601	2

2) Implementation Details

We compare the performance of four different approaches: i) An approach that assumes a trusted third party where the data of all the sites is accumulated centrally and a classifier CART is induced at this central site; which is then sent to all the parties ii) approach where all the parties calculate the attribute measures using SMC and produce a classifier CART at their respective site iii) ensemble classifier approach as discussed in Section II and iv) proposed approach of creating ensemble of k-anonymous decision trees.

In literature, k-anonymous decision tree classifier [31] has been described. We propose and implement an ensemble of such k-anonymous decision trees. Within this approach, each site will induce its own k-anonymous decision tree and a global classifier ensemble from these local classifiers will be created at a non-trusted third party or at either of the parties. We calculate the value of parameter k as 5% of the total number of training instances. All the algorithms have been implemented in MATLAB 7.8.0 (R2009a).

To evaluate the performance of these algorithms, we choose a holdout method with $3/4^{th}$ of the data used for training and rest for testing. To create an environment where the data is homogeneously distributed, we divide this training data into three random parts with each party owning one of these parts.

3) Results

The results of predictive accuracy and training time taken by all four methods on the stated data sets are tabulated in Table V and VI respectively. All the results are averaged over 10 runs.

TABLE V. ACCURACY OF CLASSIFIERS (IN %)

Data Set	CART (Trusted third party/SMC)	Ensemble Classifier	Anonymous Decision Tree Ensemble
Transfusion	75.1	78.4	74.57
Diabetes	74.62	76.82	74.15
Bank Marketing	86.65	88.46	84.93
Spambase	91.3	90.99	88.36

Since the classifier obtained by assuming a trusted third party and the one induced using SMC is same, their accuracy is mentioned together. Irrespective of the accuracy obtained, the former approach is not feasible because, in the competitive era, it is difficult to trust a third party.

TABLE VI. TESTING TIME OF CLASSIFIERS (IN SECONDS)

Data Set	CART (Trusted third party)	SMC	Ensemble Classifier	Anonymous Decision Tree Ensemble
Transfusion	2.07	10.79	3.39	4.24
Diabetes	0.32	12.68	1.51	2.09
Bank Marketing	0.99	13.18	2.33	3.11
Spambase	11.02	39.21	13.35	17.28

Further, it can be observed that when a classifier is induced using SMC, the accuracy is high; but, as a large amount of communication is required, the time taken in training a classifier is very high. Moreover, with the increase in number of attributes, the communication required also increases. This is because each party has to share each of its attribute-value pair with the other parties. Hence, SMC is not a very suitable technique in today's big data epoch. As ensemble classifiers produce more accurate results, the approach is quite suitable for Privacy-Preserving Classification of Homogeneously Distributed Data and the same is proved experimentally. However, few conclusions about data at other sites can be easily derived from the classifiers released by those sites and privacy can be breached. Our proposed approach of k-anonymous decision tree classifier ensemble overcomes this disadvantage and preserves privacy to a greater extent. Also, unlike traditional privacy protection techniques such as data swapping and adding noise, information preserved using k-anonymization remains truthful. From the experiments it is clear that k-anonymous decision tree ensemble has good accuracy and the training time is also acceptable. Hence, they can be regarded as one of the best privacy-preserving data classification algorithms for homogeneously distributed data.

IV. PRIVACY-PRESERVING CLASSIFICATION OF HOMOGENEOUSLY DISTRIBUTED DATA STREAM

Data stream classification and privacy-preserving classification of homogeneously distributed data are both emerging issues in data mining and several attempts have been made in literature to solve these issues. In the previous section, several methods for the same have empirically been evaluated and compared. But, an even more imminent challenge is when the training data tuples are stored in

multiple autonomous sites and the goal is to classify the data streams arriving at these sites while preserving the privacy of the data. In literature, privacy-preserving classification of heterogeneously distributed data streams has been attempted a little. However, as mentioned in section I, there are several real-world problems which demand classification of homogenously distributed data streams. Some examples of such applications include credit card fraud detection, classifying loan applications, customer churn prediction, ATM transactions fraud detection, telecom fraud detection, insurance fraud detection, disease outbreak detection, biological attack detection, credit scoring, etc.

From our experiments, we found anonymous decision tree ensemble and Hoeffding adaptive tree (HAT-ADWIN) as efficient and suitable candidates for privacy-preserving classification of homogeneously distributed data and data stream classification respectively. Therefore, we propose an approach to combine the said two methods by inducing an anonymous adaptive Hoeffding tree at the local sites and building an ensemble classifier from these local classifiers. This global ensemble can hence be employed for privacy-preserving classification of homogeneously distributed data streams; which shall be the subject of our future work.

V. CONCLUSION

In this paper, the two emerging issues in data classification are discussed: data stream classification and privacy-preserving classification of homogeneously distributed. We compare some existing data stream classification techniques and identify Hoeffding Adaptive Tree with Adaptable Window as a very efficient classifier in presence as well as in the absence of concept drift. Further, from the literature survey it was found that inducing an anonymous decision tree from the private data gives good classification accuracy and also preserves privacy to a greater extent. Hence, we propose an approach of building an ensemble of anonymous decision tree classifiers for an environment where the data is homogeneously distributed across sites. The experimental results have proved that this approach gives promising results. Further, we discuss that even more rising issue is a combination of the two, known as privacy-preserving classification of homogeneously distributed data streams. We propose to use an amalgamation of Anonymous Decision Tree Ensemble and Hoeffding Adaptive Tree to produce the needed classifier and consider it as a subject of future work.

REFERENCES

[1] C. Aggarwal, "On abnormality detection in spuriously populated data streams", in *ACM SIAM Conference on Data Mining*, 2005.

[2] C. Aggarwal, *Data Streams Models and Algorithms*, Advances in Database Systems, Springer Verlag, 2006.

[3] G. Hulten, L. Spencer, and P. Domingos, "Mining time-changing data streams," in *ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2001.

[4] J. Zhan, "Privacy-Preserving Collaborative Data Mining", in *IEEE Computational Intelligence Magazine*, 2008.

[5] X. Zhuojia, Y. Xun, "Classification of Privacy-preserving distributed data mining protocols", in 6th *International Conference on Digital Information Management*, 2011.

[6] R. Agarwal and R. Srikant, "Privacy-preserving data mining", in *ACM SIGMOD International Conference on Management of Data*, 2000.

[7] Y. Lindell and B. Pinkas, "Privacy preserving data mining", in *20th Annual International Cryptology Conference on Advances in Cryptology*, pages 36–54, Springer Verlag, 2000.

[8] Y. Hwanjo, J. Xiaoqian and J. Vaidya, "Privacy-Preserving SVM using Nonlinear Kernels on Horizontally Partitioned Data", *in ACM SAC International Conference*, 2006.

[9] M. Kantarcioglu, Ed. Charu Aggarwal, Philip Yu, *Privacy-Preserving Data Mining*, Advances in Database Systems, Springer, 2008.

[10] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*. San Mateo: The Morgan Kaufmann Series in Data Management Systems, 2 ed., 2006.

[11] A. Bifet, G. Holmes, R. Kirkby and B. Pfahringer, "Data Stream Mining - A Practical Approach", *Technical report*, New Zealand: Department of Computer Science, San University of Waikato, 2011.

[12] L. Golab and T. Ozsu, *Data Stream Management*. San Mateo: Morgan and Claypool Publishers, 2010.

[13] C. Aggarwal, J. Han, J. Wang and P. Yu, "A Framework for On-Demand Classification of Evolving Data Streams", in *IEEE Transactions On Knowledge And Data Engineering*, 2006.

[14] H. Abdulsalam, D. Skillicorn, and P. Martin, "Classification using streaming random forests," *IEEE Transaction on Knowledge and Data Engineering*, 2011.

[15] L. Breiman, J. Friedman, R. Olshen, and C. Stone, *Classification and Regression Trees*. Chapman and Hall, 1993.

[16] P. Domingos and G. Hulten, "Mining high-speed data streams," in *6th ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2000.

[17] W. Hoeffding, "Probability inequalities for sums of bounded random variables", *Journal of the American Statistical Association*, 1963.

[18] A. Bifet and R. Gavalda, "Adaptive Parameter-free Learning from Evolving Data Streams", *Technical report*, Polytechnic University of Catalonia, 2009.

[19] H. Wang, W. Fan, P. Yu and J. Han, "Mining concept-drifting data streams using ensemble classifiers," in *9th ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2003.

[20] M. Masud, J. Gao, L. Khan, J. Han, and B. Thuraisingham, "Classification and novel class detection in concept-drifting data streams under time constraints," *IEEE Transactions On Knowledge And Data Engineering*, 2011.

[21] A. Godase and V. Attar, "Classifier ensemble for imbalanced data stream classification," in *ACM CUBE International Information Technology Conference*, 2012.

[22] R. Kirkby, Improving Hoeffding Trees. *PhD thesis*, Department of Computer Science, University of Waikato, 2007.

[23] N. Zhang, S. Wang, and W. Zhao, "A new scheme on privacy-preserving data classification," in *ACM International Conference on Knowledge Discovery and Data Mining*, 2005.

[24] Y. Peng, G. Kou, Y. Shi and Z. Chen, "Privacy-Preserving Data Mining for Medical Data: Application of Data Partition Methods" in *Communications and Discoveries from Multidisciplinary Data*, Springer Berlin Heidelberg, pages 331-340, 2008

[25] C. Clifton, M. Kantarcioglu, J. Vaidya, and M. Zhu, "Tools for privacy preserving distributed data mining," in *ACM SIGKDD Explorations Newsletter*, 2004.

[26] S. Samet and A. Miri, "Privacy preserving ID3 using Gini index over horizontally partitioned data," in *Computer Systems and Applications*, 2008.

[27] L. Yehuda and P. Benny, "Privacy preserving data mining" in *Advances in Cryptology*, Springer Verlag, 2000.

[28] F. Emekci, O. Sahin, D. Agrawal, and A. Abbadi, "Privacy preserving decision tree learning over multiple parties," in *Data and Knowledge Engineering*, 2007.

[29] L. Xiong, S. Chitti, and L. Liu, "Mining multiple private databases using a kNN classifier," in *ACM SAC International Conference*, 2007.

[30] V. Ciriani, S. Vimercati, S. Foresti, and P. Samarati, "*k-anonymous data mining: A survey*," Advances in Database Systems, Springer Verlag, 2008.

[31] A. Friedman, A. Schuster, and R. Wolff, "Providing k-anonymity in data mining," VLDB Journal, Springer Verlag, 2008.

[32] A. Frank and A. Asuncion, "UCI machine learning repository," University of California, Irvine, School of Information and Computer Sciences, 2014. [Online]. Available: http://archive.ics.uci.edu/ml.

[33] A. Bifet, G. Holmes, R. Kirkby and B. Pfahringer, MOA: Massive Online Analysis, 2014. [Online]. Available: http://moa.cms.waikato.ac.nz.

Comparative Analysis of 3D Face Recognition Using 2D-PCA and 2D-LDA Approaches

Ms. Dhara Marvadi
Electronics and Communication Dept.,
Chhotubhai Gopalbhai Patel Institute of Technology,
Surat, India
dharamarvadi2001@gmail.com

Dr. Chirag Paunwala
Electronics and Communication Dept.,
Sarvajanik College of Engineering and Technology,
Surat, India
chirag.paunwala@scet.ac.in

Dr. Maulin Joshi
Electronics and Communication Dept.,
Sarvajanik College of Engineering and Technology,
Surat, India
maulin.joshi@scet.ac.in

Ms. Aarohi Vora
Electronics and Communication Dept.,
Sarvajanik College of Engineering and Technology,
Surat, India
Aarohi.vora@scet.ac.in

Abstract—**Even if, most of 2D face recognition approaches reached recognition rate more than 90% in controlled environment, current days face recognition systems degrade their performance in case of uncontrolled environment which includes pose variations, illumination variations, expression variations and ageing effect etc. Inclusion of 3D face analysis gives an age over 2D face recognition as they give vital informations such as 3D shape, texture and depth which improve discrimination power of an algorithm. In this paper, we have investigated different 3D face recognition approaches that are robust to changes in facial expressions and illumination variations. 2D-PCA and 2D-LDA approaches have been extended to 3D face recognition because they can directly work on 2D depth image matrices rather than 1D vectors without need for transformations before feature extraction. In turn, this reduces storage space and time required for computations. 2D depth image is extracted from 3D face model and nose region from depth mapped image has been detected as a reference point for cropping stage to convert model into a standard size. Two Dimensional Principal Component Analysis (2D-PCA) and Two Dimensional Linear Discriminant analysis (2D-LDA) are employed to obtain feature vectors globally compared to feature vectors obtained locally using PCA or LDA. Finally, euclidean distance classifier is applied for comparison of extracted features. A set of experiments on GavabDB 3D face database, which has 61 individuals in total, demonstrated that 3D face recognition using 2D-LDA method has achieved recognition accuracy of 93.3% and EER of 8.96% over database, which is higher compared to 2D-PCA.So, more optimized performance has been achieved using 2D-LDA for 3D face recognition analysis.**

Keywords— Eigen-Surface, Fisher-Surface, PCA, 2D-PCA, LDA, 2D-LDA, EER

I. INTRODUCTION

Modern biometric security systems based on face found many applications such as criminal ID identification, verification of person's identity, border control and illegal immigration etc [1]. There is an indeed requirement of developing efficient biometric authentication system to withstand in continuously changing environment to meet above application requirements. Face as a biometric has lot of advantages as compared to other commonly used biometrics such as iris and fingerprint in different applications[1,2] because iris recognition systems are highly accurate, reliable, too intrusive yet expensive to implement and fingerprint recognition systems are highly socially accepted, reliable, non-intrusive but not applicable for non-collaborative people. More over this, they also require much explicit user co-operation. On the contrary, face recognition systems represent a good compromise between socially acceptances, reliability, no requirement of physical contact and balances security as well as privacy. Experiments under Face Recognition Vendor Test (FRVT) have explored that most of 2D face recognition approaches reached recognition rate more than 90% in controlled environment but their performance degrade significantly under uncontrolled environment which has different pose variations, illumination variations, expression variations, occlusions and ageing effect etc. 3D face recognition approaches give an age over 2D face recognition approaches as they provide much important informations such as 3D shape, texture and depth which help in improving discrimination power of an algorithm because they are somewhat invariant to transformations and intensity variation[3,1]. During initial work on 3D face, 3D acquisition was major problem because 3D capturing process was quite time consuming, costlier and not much accepted by user due to low quality of 3D face data. In today's scenario, there are lots of cheap and efficient 3D sensors are available which have opened a new door for research in the direction of 3D face recognition.

II. PREVIOUS WORK

For dimensionality reduction of face space, Principal Component Analysis (PCA) [3,5] and Linear Discriminant Analysis (LDA) [4,5] are widely used methods for efficient feature extraction and classifying face images. In 2D face recognition analysis, Yang et al. [6] investigated Two-Dimensional PCA (2D-PCA) as a solution for problem of

978-1-4799-9991-0/15/$31.00 ©2015 IEEE

estimating the covariance matrix under the small sample size condition in case of eigenface approach [3,5]. It has provided recognition accuracy of 84.24% and maximum 96.1% on Yale face database [9] as well as AR- Face database [10]. Ye et al. [7] performed Two-Dimensional Linear Discriminant Analysis (2D-LDA) which was also an extension of LDA [4, 5] and provided more robust performance for small sample size problem. A set of experiments on ORL database [11] have demonstrated recognition rate of 97.50%.

In case of 3D face analysis, Hesher et al. [12] extended PCA approach to 3D face recognition analysis. In this, PCA has been applied on range images with euclidean distance classifier for comparison among the feature vectors. The recognition accuracy has been achieved by them was 100%. Then, further analysis has been carried out by Heseltine et al. [13, 14]. They investigated 3D face recognition method based on eigen-surface approach and also obtained recognition accuracy of 87.3% when tested against York database [20]. Heseltine et al. [14] performed 3D face recognition method based on Belhumeu's fisher-surface approach which involved implementation on PCA before LDA or Fisher-Surface approach and the recognition accuracy achieved by them was greater than 88.7% on York database [20].

Khalid et al. [15] inspected 3D face recognition using local geometric feature-PCA with euclidean distance classifier and produced recognition rate of 86% with first rank matched on GavabDB database [19]. Li et al. [16] evaluated 3D face recognition using geometric feature extraction with consecutive two steps, which involved ICP algorithm need to be performed for matching the probe images with all database face models to make the final decision. The optimized performance inspected was recognition rate of 91.1% with CASIA 3D database [21]. Ming et al. [17] examined 3D face recognition using learn correlative features which can be obtained by using 3DLBP approach (3D Local Binary Pattern) and PCA. They obtained recognition accuracy of 94.17% on CASIA 3D face database [21]. Yashar et al. [18] studied 3D face recognition using 2D-PCA which first need to detect nose region and obtained recognition accuracy of 98% after set of experiments performed on CASIA 3D face database [21].They also claimed that their approach provides better stabilization against above challenges.

III. Pre-Processing

In depth image, first nose tip has to be detected because nose tip has highest value of depth information. As shown in Fig.1 for depth image or range image, the lighter areas are more close to 3D sensors and darker areas are far away from sensor. Thus, lighter area of face has large amount of depth information as compared to darker areas. Since, nose is a part more close to 3D sensors in case of 3D face; it will definitely have more depth information contained rather than any other fiducial points. To detect nose, depth images will be scanned with a window of *3×3 or 5x5* size, then sum of all points inside and below the *3×3 or 5x5* windows will be obtained. The largest number in these windows data is the nose tip. In certain conditions including pose variation or illumination variation, sometime the depth information of chin is more than nose. In such situations to prevent wrong nose detection, we need to adapt some modification to above discussed method. In such cases, this method will only accept the points from the central areas of pictures as nose generally lies at the center area of face image.

After nose detection, cropping has to be performed on a range image to convert it into fixed size matrices. Then after smoothing need to perform on an image. Here, Gaussian filtering method has been applied on overall database for elimination of noise effect and variation due to facial expression which is the most basic approach for smoothing.

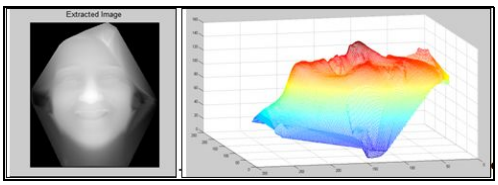

Fig.1. Range image and wireframe diagram of 3D face from GavabDB database in MATLAB

IV. 2D-PCA:A Brief Review

The major advantage of 2D-PCA over eigen- surface approach or PCA is that it can directly work on 2D image matrices instead of 1D vectors without any need for transformations [7]. Therefore, it has less memory requirement and computational complexity with increased speed of operation. The image covariance matrices or correlation matrices can be computed from depth image matrices. Consider U denotes the depth image matrix with m rows and n columns. Then, this matrix is projected on to n-dimensional column vector x as described in Eq.(1) as follows:

$$Y=U*x \qquad (1)$$

Here, Y is the projected feature vector of the depth image U with size of m dimensional column vector. Let S_t be the covariance or scatter matrix and is given by Eq.(2),

$$S_t= E[] \qquad (2)$$

Where, E[.] is the expectation operator and is the mean range image of the entire training samples. If C be the total number of eigenvectors evaluated and B_i as well as B_j are two different vectors applied for euclidean distance classifier than the difference between the two principal component vectors from test and training image can be given as shown in Eq.(3). Generally, the depth image with lowest euclidean distance will be selected as matched image in all security systems..

$$d(\qquad (3)$$

Unlike the conventional PCA, the 2D-PCA does not involve computation of a large correlation matrix and therefore, it is relatively less computationally complex and more speedy.

V. 2D-LDA: A Brief Review

2D-LDA [8] is also an extension of LDA for two dimensional matrices analysis. In this also, there is no need of transformation for the depth image matrices into 1D vectors as a column vectors. Consider a depth image U of m rows and n columns with projection given as Y = U*x, where Y is the projected vector and x is the projection vector. The major concern is to have optimized performance for cost function and it is given by Eq.(4),

$$J(x) = \qquad (4)$$

Where, S_b and S_w are between-class scatter matrix and within-class scatter matrix as described below in Eq.(5) and Eq.(6),

$$(\qquad (5)$$

$$(6)$$

Compared to 2D-PCA, 2D-LDA has less computational complexity, less time requirement and has always more discrimination power for class distributions.

VI. Experiments Results

GavabDB database has been used to test and evaluate performance of approaches such as eigen-surface, fisher-surface, 2D-PCA and 2D-LDA for 3D face recognition. It contains 549 three-dimensional images of facial surfaces for 61 individuals having 9 images for each person. Each image is given by a mesh of connected 3D points of the facial surface without texture. This database also provides systematic variations with respect to the pose and the facial expression.

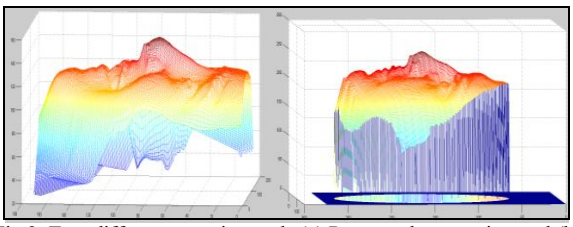

Fig.2. Two different cropping tools (a) Rectangular cropping tool (b) Designed elliptical cropping tool

Table I gives performance comparison of 3D face recognition analysis using above mentioned methods and it has been proved that 3D face analysis based on 2D-LDA is far better as compared to eigen-surface, fisher-surface and 2D-PCA over 3D face models. 2D-PCA and 2D-LDA provide more optimized recognition as compared to 3D implementation of eigen-surface and fisher-surface approaches as well as PCA and LDA method on 2D databases. So, we can conclude that 3D face analysis based on 2D-LDA is faster, computationally efficient and results in higher recognition performance as compared to other 3D face recognition approaches mentioned above.

On overall database, 3D face recognition using 2D –LDA has achieved recognition accuracy of 93.33% and computation time of 14.43 sec which is lowest time required compared to all other methods. 3D face recognition by 2D-LDA has also provided recognition accuracy of 98.9% on only frontal 3D GavabDB database.

Fig.3 shows comparison of ROC curves for 3D face recognition analysis based on eigen-surface[13], fisher-surface[14], 2D-PCA[18] and 2D-LDA [7] that are being implemented. As can be seen from ROC curve analysis, the performance of 2D-LDA is far better as compared to eigen-surface, fisher-surface and 2D-PCA over GavabDB database. As the performance of 2D-LDA slightly better than 2D-PCA, it has provided a scope for lot of research in this direction.

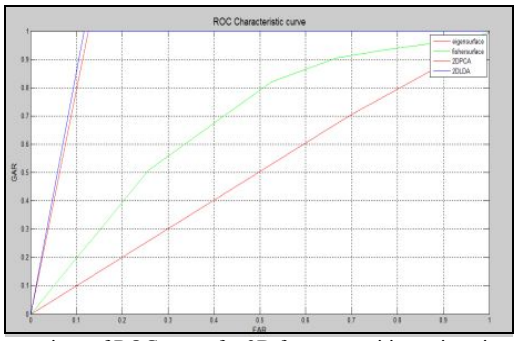

Fig.3. Comparison of ROC curve for 3D face recognition using eigen-surface, fisher-surface, 2D-PCA and 2D-LDA for 30 user database

Comparison of performance curves for 3D face recognition based on eigen-surface and fisher-surface for 30 user database has been demonstrated in Fig.4. As can be seen from performance curve analysis, EER for fisher-surface is lower as compared to eigen-surface approach for 3D face recognition analysis. EER of low value is desirable for any practical biometric authentication system.

Table I. Performance comparison of Eigen-Surface[13] , Fisher-Surface[14], 2D-PCA[18] and 2D-LDA[7,18] on GavabDB database

3D Database	Eigen-Surface		Fisher-Surface		2D PCA		2D LDA	
	Accuracy	*Time (sec)*	*Accuracy*	*Time (sec)*	*Accuracy*	*Time (sec)*	*Accuracy*	*Time (sec)*
GavabDB (Frontal)	90.93%	23.59	92.89%	22.87	97.33%	15.13	98.9%	13.98
GavabDB (Full)	89.63%	26.36	91.26%	25.98	92.67%	17.83	93.33%	14.49

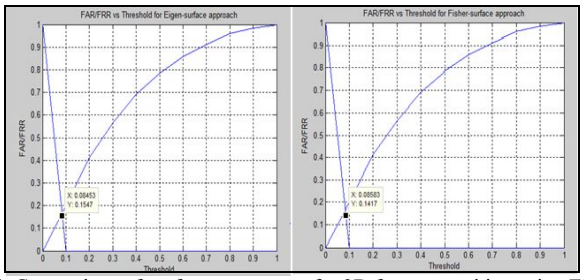

Fig.4. Comparison of performance curve for 3D face recognition using Eigen-Surface and Fisher-Surface approaches over 30 user database

Fig.5 illustrates comparison of performance curves for 3D face recognition based on 2D-PCA and 2D-LDA for 30 user database. As can be seen from performance curve analysis, the performance of 2D-LDA is far better as compared to eigen-surface, fisher-surface and 2D-PCA over 3D face database of 30 users with lowest value of EER.

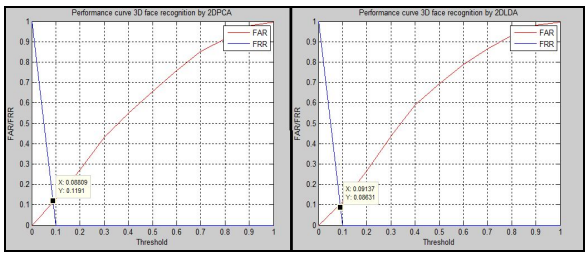

Fig.5. Comparison of performance curve for 3D face recognition using 2D-PCA and 2D-LDA approaches over 30 user database.

TABLE II. COMPARISON OF EER FOR DIFFERENT 3D FACE RECOGNITION USING EIGEN-SURFACE, FISHER-SURFACE, 2D-PCA AND 2D-LDA APPROACHES

Methods	Eigen-Surface	Fisher-Surface	2D-PCA	2D-LDA
EER(%)	15.25	14.1	11.91	8.96

Table 2 gives detailed analysis of EER for all four implemented method for depth images and from this analysis, we can conclude that 2D-PCA and 2D-LDA methods on 3D databases has achieved very less error rate as compared to 3D face analysis by eigen-surface and fisher-surface approaches. The optimized value of EER can be derived by 3D face recognition using 2D-LDA is 8.96%. So, we concluded that 2D-LDA based 3D face recognition system has optimized performance over all other methods and it can be used for different 3D face recognition applications.

VII. CONCLUSION

From above analysis, we can conclude that 2D-LDA based 3D face recognition performs far better as compared to 2D-PCA, eigen-surface and fisher-surface methods in case of 3D face analysis. The GAR of 2D-LDA based recognition is better as compared to all other methods. It provides almost about 96-99% recognition rate and EER of 8.96%. From above analysis, we can also conclude that 3D face analysis using 2D-LDA outperform 2D-PCA based 3D face analysis and gives more optimized performance for above mentioned challenges.Thus,3D AFR (Automatic Face Recognition) system based on 2D-LDA saves memory, time and provides very high recognition accuracy. These experiments have been demonstrating that 3D face recognition system has good scope in future compared to conventional 2D based face recognition systems if efficient algorithms have been designed.

REFERENCES

[1] Jain, Anil K., Patrick Flynn, and Arun A. Ross. "*Handbook of biometrics",* Springer, 2007.

[2] Stan, Z. Li, and K. Jain Anil. *"Handbook of face recognition",* Springer, 2005.

[3] Kirby, Michael, and Lawrence Sirovich. "Application of the Karhunen-Loeve procedure for the characterization of human faces." *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol.12, no. 1, pp.103-108, 1990.

[4] Lu, Juwei, Konstantinos N. Plataniotis, and Anastasios N. Venetsanopoulos. "Face recognition using LDA-based algorithms." *IEEE Transactions on Neural Networks,* vol.14, no. 1, pp.195-200, 2003.

[5] Martínez, Aleix M., and Avinash C. Kak. "Pca versus lda." *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol.23, no. 2, pp.228-233, 2001.

[6] Yang, Jian, David Zhang, Alejandro F. Frangi, and Jing-yu Yang. "Two-dimensional PCA: a new approach to appearance-based face representation and recognition." IEEE *Transactions on Pattern Analysis and Machine Intelligence*, vol.26, no.1, pp.131-137, 2004.

[7] Wang, Xiao-ming, Chang Huang, Xiao-ying Fang, and Jin-gao Liu. "2DPCA vs. 2DLDA: face recognition using two-dimensional method." *International Conference on Artificial Intelligence and Computational Intelligence, AICI'09.*, vol. 2, pp. 357-360, 2009.

[8] Ye, Jieping, Ravi Janardan, and Qi Li. "Two-dimensional linear discriminant analysis." In *Advances in neural information processing systems*, pp. 1569-1576. 2004.

[9] Georghiades, A. "Yale face database." *Center for computational Vision and Control at Yale University, http://cvc. yale. edu/projects/yalefaces/yalefa*, 1997.

[10] Martinez, Aleix M. "The AR face database." *CVC Technical Report* 24 ,1998.

[11] AT & T (ORL) Face Database: http://www.cl. cam. ac. uk /research/dtg/ attarchive/face data base.html

[12] Hesher, Curt, Anuj Srivastava, and Gordon Erlebacher. "A novel technique for face recognition using range imaging." In *Seventh international symposium on Signal processing and its applications*, vol. 2, pp. 201-204, 2003.

[13] Heseltine, Thomas, Nick Pears, and Jim Austin. "Three-dimensional face recognition: An eigensurface approach." *IEEE International Conference on Image Processing, ICIP'04.*, vol. 2, pp. 1421-1424, 2004.

[14] Heseltine, Thomas, Nick Pears, and Jim Austin. "Three-dimensional face recognition: A fishersurface approach." *In Springer Berlin Heidelberg Image Analysis and Recognition*, pp. 684-691, 2004.

[15] Khalid, F., Tengku Mohd Tengku, and K. Omar. "Face recognition using local geometrical features-PCA with euclidean classifier." IEEE *International Symposium on Information Technology*, ITSim., vol. 2, pp. 1-6, 2008.

[16] Li, Yong-An, Yong-Jun Shen, Gui-Dong Zhang, Taohong Yuan, Xiu-Ji Xiao, and Hua-Long Xu. "An efficient 3D face recognition method using geometric features." IEEE *2nd International Workshop on Intelligent Systems and Applications (ISA),* pp. 1-4., 2010.

[17] Ming, Yue, Qiuqi Ruan, Xueqiao Wang, and Meiru Mu. "Robust 3D face recognition using learn correlative features." IEEE 10th International Conference on Signal Processing (ICSP), pp. 1382-1385, 2010.

[18] Taghizadegan, Yashar, Hassan Ghassemian, and Mohammad Naser-Moghaddasi. "3D Face Recognition Method Using 2DPCA-Euclidean Distance Classification." ACEEE International Journal on Control System and Instrumentation, vol.3, 2012.

[19] Moreno, A. B., and A. Sanchez. "GavabDB: a 3D face database." In *Proc. 2nd COST275 Workshop on Biometrics on the Internet, Vigo (Spain)*, pp. 75-80, 2004.

[20] The 3D Face Database, The University of York. www.cs.york.ac.uk/~tomh

[21] CASIA 3D face database, National Laboratory of Pattern Recognition, CASIA, P. O. Box 2728, Beijing, 100190, China