Logical Symbol Recognition using Normalized Chaincode and Density Features

Ms. Manisha Bharambe
Department of Computer Science,
Yashwantrao Mohite College,
Bharati Vidyapeeth Deemed University
Pune, Maharashtra, India

***Abstract* - Mathematical expressions are used in many scientific documents. The recognition of handwritten mathematical symbols is one of the most challenging research areas in the field of image processing and pattern recognition. The difficulties of handwritten mathematical symbol recognition is due to variability of the symbols and their two dimensional structure. This paper implements feature extraction algorithms and analyzes the performance of a recognizer. The strength of the proposed approach is its methods for efficient preprocessing and feature extraction. In this work, mathematical logical symbol recognition system has been developed by using support vector machine (SVM) and back propagation neural network. By applying combinations of normalized chain code, density, image invariance features to SVM and Artificial Neural Network (ANN), a high recognition accuracy is attained. A database of 2000 symbols was created. Preprocessing techniques are used to remove noise and thinning of the image, and features are extracted. The recognition rate for handwritten mathematical logical symbols is observed to be high when SVM is used.**

Keywords- Mathematical logical symbols, Preprocessing, Feature extraction, chain code feature, density feature, ANN, SVM

I. INTRODUCTION

Mathematics is widely used in all fields of science such as physics, engineering, medicine, etc. Input methods to include mathematical expressions into scientific documents are required. Logical symbols are used in mathematical expressions. This paper deals with recognition of handwritten logic symbols. Since logic symbols are used in electronic circuit design, artificial intelligence, knowledge representation schemes, and chemical equations, there is a need to recognize logic symbols more efficiently and convert them into a form understandable by computer. Considerable research has been conducted on pattern recognition problems and much attention has been paid to develop methods for recognition of mathematical symbols Hsi-Jim Lee et al [1] present a system to segment and recognize texts and mathematical expressions in a document. 4-Dimensional direction features are extracted from each image block and has been normalized, When the aspect ratio of a symbol is very small (smaller than the threshold *TI),* the horizontal feature and the two diagonal features are more important than the vertical feature. The system can be divided into six stages: page segmentation and labeling, character segmentation, feature extraction, character recognition, expression formation, and error correction and expression extraction. Similar symbols are group together, six groups are form. Then applying heuristic rules syntax tree is generated, which form an expression. This paper mainly focus on segmentation of expression from documentation. In 2008, Christopher Malon et al [2] proposed the use of support vector machines to improve the classification of InftyReader and compare the performance of SVM kernels and feature definitions on pairs of letters that InftyReader usually confuses. By running the purified InftyReader engine on the training data, which produce an integer-valued confusion matrix, with rows that count ground truth and columns that count recognition results. Every nonzero off-diagonal entry of this matrix represents a confusing pair, for which an SVM should be trained. Infty engine is used for recognition and results are compared with SVM. Without SVM, the pure Infty engine recognized characters with 96.10% accuracy on testing data set. Using SVM, the recognition rate rised to 97.70%,. In 2013,Dipak Bage, K.P. Aditya, Sanjay Gharde [3] proposed a new approach for offline handwritten mathematical symbol recognition system using character geometry. A number of feature extraction techniques such as: chain code, directional features, local features like center of gravity, aspect ratio, width, height, vertical and horizontal projection and global features like geometric features based character geometry on are used. Recognition rates for SVM, KNN, Neural Network classifier are compared in the case of offline mathematical symbols. To recognize online symbols template matching is used. Francisco Alvaro, John Andren Sanchez [5] have tested classical and novel classification techniques for offline printed mathematical symbol recognition on two databases. The proposed techniques were evaluated for two different databases: the UWIII database that is a small database with degraded images and, the InftyCDB-1 database that is large database with good quality images. Three classification methods HMM, KNN and SVM were compared. The best results were obtained with SVM, but KNN obtained analogous competitive results. The worst results in this experiment were obtained by HMM on the both databases. [8] Stephen M. Watt presents a recognition system for handwritten mathematical symbols that uses Elastic matching which is a model-based method that involves computation proportional to the set of candidate models. This recognizer can recognize digits, English letters, Greek letters, most of the common mathematical operators and notations. On a database of

10,000 mathematical handwriting samples, the recognizer achieved 97% accuracy. In 2004, Utpal Garain et al [11]deals with recognition of printed mathematical symbols. A group of classifiers arranged hierarchically is used to achieve robust recognition of a large number of symbols appearing in expressions. The various classifiers used were Run-number based Classification (*C*2), Grid based Classification (*C*3) Wavelet based Classification (*C*4) and C1(combination of all three). Number of symbols used were 59,288 and accuracy obtained was 98.3%. Table I shows the analysis of feature extraction and classification of mathematical symbols.

II. PHASES OF OCR SYSTEM

The steps in any OCR system are Data collection, Preprocessing, Feature extraction and Classification. There is no standard database for logical symbols, so a database is developed by collecting data from different writers. A4 sheets were used for data collection. Data is collected from twenty writers from different fields, each symbol written by each writer 10 times. Ten logical symbols are used for recognition. Figure 1 shows the sample sheet of handwritten mathematical logical symbols. The data sheets were scanned and individual symbol images were cropped from this scanned image manually, which resulted in gray scale image of symbols.

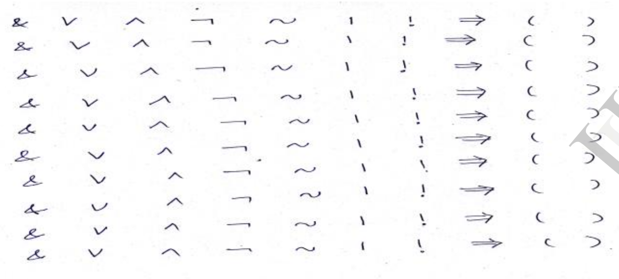


Fig 1: Sample sheet of handwritten logical symbols

A. *Preprocessing:*

To enhance the image and prepare for next phase, preprocessing is done. The following steps were involved in preprocessing:

- Read the gray scale image.
- Remove noise.
- Convert the gray scale image into binary image using threshold value.
- Invert the image, foreground pixels are ‘1’ value pixels.
- Crop the image and get its height and width.
- Normalize image to a predefined size.

Convert the normalize image to a single pixel thickness using thinning.

III. FEATURE EXTRACTION TECHNIQUES

Feature extraction techniques were applied to offline input data set after preprocessing the image.

A. *Normalized Chain Code*

Chain code representation gives the code of the boundary of character image. Using freeman chain code method, extract chain code of each image. Length of chain code varies from image to image. So the chain code is normalized [12] by the following method:

Step1: Find chain code of image using Freeman method.

Step 2: Find the frequency of each digit to get vector A1 of size 8.

Step 3: Take sum of all A1 elements (A2=∑A1)

Step 4: Calculate probability of each digit A3=A1 / |A2|

After concatenation of the two vectors A1 and A3 , we get 16 features.

B. *Moment Invariant Features*

A set of seven 2-D moment invariant features are insensitive to translation, scale change, mirroring, and rotation. These can be derived from seven equations. It computes the moment invariants of the image and obtained seven-element row vector. Using these features we get a high degree of invariance. Hu’s Seven Moment Invariants are invariant under translation, changes in scale, and also rotation. So it describes the image despite of its location, size, and rotation. The seven features of moment invariant are extracted from the image.

C. *Density Feature: UDRL density(Up-Down, Right-Left).*

Step 1: Image of size (32*32) is divided into zones. Each zone of size 8*8 gives 16 zones.

Step 2: Sum foreground pixels in each zone, get the 16 array (density of each zone).

Step 3: Extract height and width, difference between left and right zone density, and up and down density.

Up=sum of Density 1 to 8 zones, and Down =sum of density of 9 to 16 zones

Left= sum of density of 1, 2,5,6,9,10,13,14 zones, and Right= total density-left

Step 4: Calculate difference, diff1=up-down and diff2=left-right

Step 5: If the diff1 >2 then d1=1, if diff1 <-2 then d1=2 else d1=0

If diff2>2 then d2=1 , if diff2<-2 then d2=2 else d2=0 **(Here** -2 to 2 is error rate)

Step 6: To get average density, Combine two consecutive zones, and find sum of density, then sum is divide by number of pixels in that zone. Finally, we obtained 8 features of average density of 8 zones.

D. . *Methods of creation of feature vector*

The feature selection is an important factor to increase the recognition rate. A combination of features are used for better recognition.

- The 8 features of average density and 2 features d1 and d2 (*method C*), 16 features of normalized chain code (*method A*) are extracted, to form feature vector of size 26.

- The 7 moment invariant features (*method B*) are combined with normalized chain code to form feature vector of size 23.

IV. CLASSIFICATION

The two feature vectors created are given as the input to the classifier for recognition of symbols with class labels as target. The two classifiers Artificial Neural Network (*ANN*) and Support Vector Machines (*SVM*) are used to carry out experimentation.

A. Artificial Neural Network

A feed forward back propagation neural network is used for classification of handwritten logical symbols. The structure of neural network includes an input layer, one hidden layer and an output layer. As discussed in Section 3, feature vector is created for each image. The fig. 2 and fig. 3 shows the two combinations of feature vectors with input neurons 26 and 23 respectively.

- The input layer of ANN consists of 26 neurons, and output layer with 10 neurons.

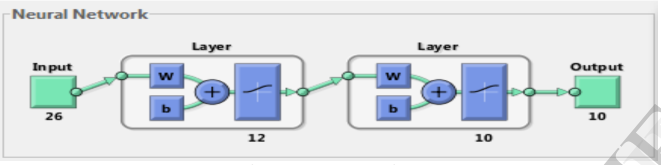


Fig 2: View of network1 with feature vector 26

- The input layer of ANN consists of 23 neurons, and output layer with 10 neurons.

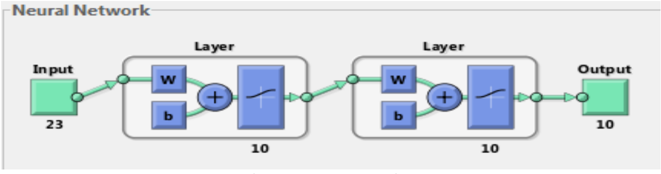


Fig 3: View of network1 with feature vector 23

The back propagation method with momentum and adaptive learning rate and log-sigmoid transfer functions are used for neural network training. Neural network has been trained using known dataset. Recognition systems using 26 and 23 features are built. The number of input nodes is chosen based on the number of features. With default parameters the data will be divided randomly into 70%/15%/15% for training /validating /testing. The default training function is 'trainlm' which uses the Levenberg-Marquardt algorithm for back propagation. The number of hidden neurons are decided to get efficient result.

B. SVM

Support Vector Machines is one of the supervised learning method. SVM are based on the concept of decision planes that define decision boundaries. A decision plane is used to separates a set of objects having different class memberships. SVM is classifier which construct hyper planes in a multidimensional space that separates cases of different class labels and perform classification. The linear kernel bsvm2 is used for training to build a model that assign a class labels for each category.

V. EXPERIMENTAL RESULTS

The recognition system has been implemented using Matlab. Classification of testing data is carried out by using ANN and SVM with input vector 26 and 23. The result of test data is shown in the table II.

Table II: Result of classification using ANN and SVM

Feature Vector Size	**Train data**	**Test data**	**Accuracy Using ANN**	**Accuracy Using SVM**
26	2000	500	83%	94.2%
23	2000	500	83.2%	98.2%

VI. CONCLUSIONS

This paper describes an efficient methods to extract features by combining features which gives high recognition rate. Table III and Table IV show the results of test data using confusion matrix with SVM classifier. Table V and Table VI show the confusion matrix for trained samples with 23 and 26 feature vectors respectively with ANN. Experimental results show that the recognition rate using ANN with normalized chain code combine with density feature and combine with moment invariants feature are same. The SVM classifier gives better results than ANN classifier. Future work will focus on the recognition of mathematical logical expressions with high accuracy rate by reducing the size of the feature vector.

ACKNOWLEDGMENT

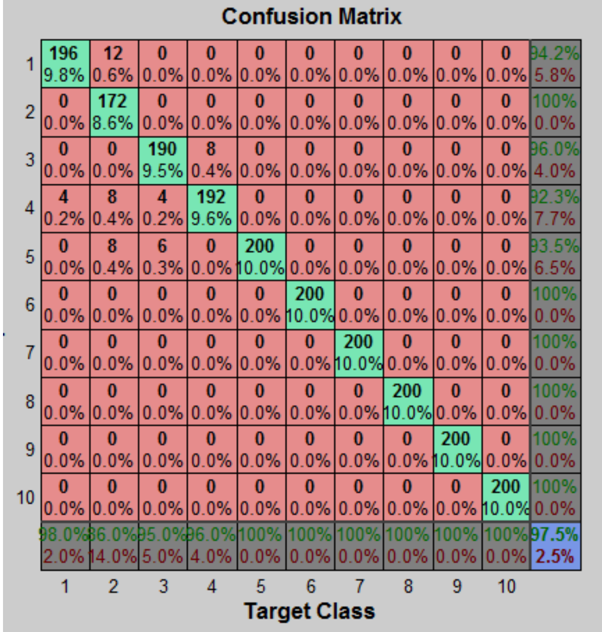
The author is grateful to Dr. M.S. Prasad for his contributive support and encouragement during this work.

REFERENCES

1. His-Jian Lee And J. Wang. Design of a mathematical expression recognition system, 0-8186-7128-9/95,IEEE
2. Christopher Malon, Seiichi Uchid, Masakazu Suzuki, Mathematical symbol recognition with support vector machines, Pattern Recognition Letters 29 (2008),1326 1332, Elsevier.
3. Dipak D.Bage, K.P. Aditya, Sanjay Gharde, A new approach for Recognizing offline handwritten mathematical symbols using character geometry, International Journal of Innovative Research in Science, Engineering and Technology, vol. 2,Issue 7, July 2013,ISSN;2319-8753.
4. Erik G. Miller and Paul A. Viola, Ambiguity and Constraint in Mathematical Expression Recognition, , American Association for Artificial Intelligence,1998.
5. Francisco Alvaro, John Andren Sanchez. Comparing several techniques for offline recognition of printed mathematical symbols. 1051 4651/10/2010, IEEE.

6. Frank D. J. Aguilar and Nina S. T. Hirata, *ExpressMatch* : A system for creating ground-truthed datasets of online mathematical expressions, Department of Computer Science, Institute of Mathematics and Statistics University of Sao Paulo S˜ao Paulo, Brazil
7. M. Koschinski, H.-J. Winkler, M. Lang, Segmentation and Recognition Of Symbols Within Handwritten Mathematical Expressions, 0-7803- 2439, 1995 IEEE.
8. Stephen M. Watt Xiaofang Xie, Prototype Pruning by Feature Extraction for Handwritten Mathematical Symbol Recognition, Department of Computer Science, University of Western Ontario,Canada.
9. Stephen Watt and Xie. Recognition for large sets of handwritten mathematical symbols. 1520-5263/05, IEEE.
10. Sumit Saha Tanmoy Som, Handwritten character recognition by using Neural-network and Euclidean distance metric, IJCSIC – International Journal of Computer Science and Intelligent Computing Vol. 2, No. 1, November 2010 ISSN: 0976.
11. Utpal Garain, B. B. Chaudhuri, R. P. Ghosh, A Multiple-Classifier System for Recognition of Printed Mathematical Symbols, Proceedings of the 17th International Conference on Pattern.Recognition (ICPR'04),1051-4651/04 IEEE.
12. G.G.Rajput, S.M.Mali ,Marathi Handwritten Numeral Recognition using Fourier Descriptors and normalized Chain Code, IJCA,special issue on "Recent trends in Image Processing and Pattern Recognition", 2010.
13. J. Pradeep, E. Srivasan, S. Himavathi, Diagonal Feature Extraction Based Handwritten Character System using Neural Network, International Journal of Computer Applications® (IJCA),(0975-8887),volume 8-no.9, October 2010.
14. Md Fazlul Kader1 and Kaushik Deb,Neural Network-Based English Alphanumeric Character Recognition, International Journal of Computer Science, Engineering and Applications (IJCSEA) Vol.2, No.4, August 2012
15. Sanjay S. Gharde, Baviskar Pallavi, V K. P. Adhiya, Evaluation of Classification and Feature Extraction Techniques for Simple Mathematical Equations, International Journal of Applied Information Systems (IJAIS) – ISSN : 2249-0868 Foundation of Computer Science FCS, New York, USA Volume 1– No.5, February 2012.
16. N. VenkateswaraRao, Dr. A. Srikrishna, Dr. B. RaveendraBabu,G. Rama Mohan Babu, An Efficient Feature Extraction And Classification Of Handwritten Digits Using Neural Networks, International Journal of Computer Science, Engineering and Application (IJCSEA), Vol.1, No.5, October 2011.
17. Sumit Saha Tanmoy Som, Handwritten character recognition by using Neural-network and Euclidean distance metric, IJCSIC – International Journal of Computer Science and Intelligent Computing Vol. 2, No. 1, November 2010 ISSN: 0976
18. VenkateswaraRoa N., Dr A. Shrikrishna, Dr. Ravindrababu. An efficient feature extraction and classification of handwritten digits using neural networks.IJCSEA,Vol1, No.5 October 2011.
19. V. Vijaya Kumar, A. Srikrishna, B.Raveendra Babu and M. Radhika mathematical morphology. Sadhana Vol. 35 Part 4, pp 419-426, Aug 2010.
20. J. Pradeep, E. Srivasan, S. Himavathi, Diagonal Feature Extraction Based Handwritten Character System using Neural Network, International Journal of Computer Applications, (IJCA),(0975-8887),volume 8-no.9, October 2010.

Table V: Confusion matrix for train data using ANN with feature vector 23

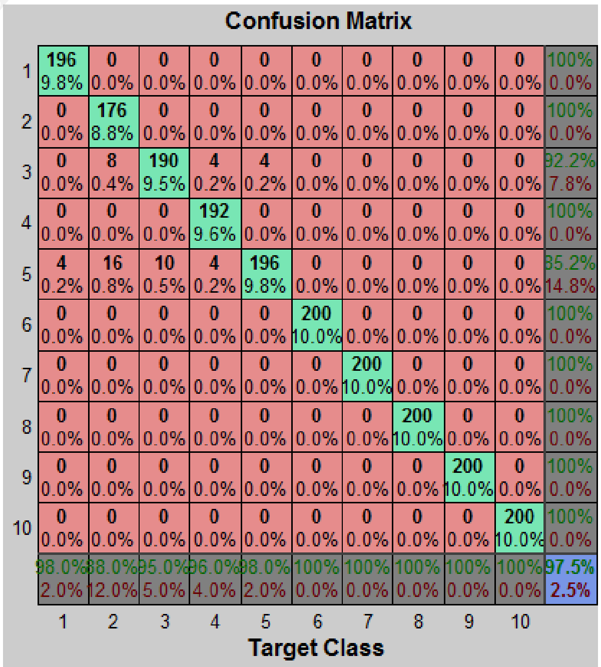


Confusion Matrix

1	196 9.8%	12 0.6%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	94.2% 5.8%
2	0 0.0%	172 8.6%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
3	0 0.0%	0 0.0%	190 9.5%	8 0.4%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	96.0% 4.0%
4	4 0.2%	8 0.4%	4 0.2%	192 9.6%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	92.3% 7.7%
5	0 0.0%	8 0.4%	6 0.3%	0 0.0%	200 10.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	93.5% 6.5%
6	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	200 10.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
7	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	200 10.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
8	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	200 10.0%	0 0.0%	0 0.0%	100% 0.0%
9	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	200 10.0%	0 0.0%	100% 0.0%
10	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	200 10.0%	100% 0.0%
	98.0% 2.0%	86.0% 14.0%	95.0% 5.0%	96.0% 4.0%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	97.5% 2.5%
	1	2	3	4	5	6	7	8	9	10	

Target Class

Table VI: Confusion matrix for train data using ANN with feature vector26



Confusion Matrix

1	196 9.8%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
2	0 0.0%	176 8.8%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
3	0 0.0%	8 0.4%	190 9.5%	4 0.2%	4 0.2%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	92.2% 7.8%
4	0 0.0%	0 0.0%	0 0.0%	192 9.6%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
5	4 0.2%	16 0.8%	10 0.5%	4 0.2%	196 9.8%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	85.2% 14.8%
6	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	200 10.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
7	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	200 10.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
8	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	200 10.0%	0 0.0%	0 0.0%	100% 0.0%
9	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	200 10.0%	0 0.0%	100% 0.0%
10	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	200 10.0%	100% 0.0%
	98.0% 2.0%	88.0% 12.0%	95.0% 5.0%	96.0% 4.0%	98.0% 2.0%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	97.5% 2.5%
	1	2	3	4	5	6	7	8	9	10	

Target Class

Table I: Analysis of feature extraction and classification of Mathematical symbols

Paper	Dataset	On line	Offline	Model / Features	Dataset Size	Accuracy Claimed
[1]	Mathematical Expressions		√	Using heuristic rules/ Aspect ratio Vertical, horizontal, diagonal features	127 letters, 36 mathematical operators, 20 numerical numbers	96%
[2]	Mathematical symbols		√	SVM/ Confusion matrix	InftyCDB-1	97.70%
[3]	Mathematical symbols Printed And handwritten	√	√	SVM, Template maching(KNN)/ Chain coding,Center of gravity, Aspect ratio, Intersection points, Gaber, Zoning,Characcter geometry (directional features)	284739 62100	97% 98%
[5]	Printed Mathematical symbols		√	SVM KNN HMM	UWIII database InftyCDB-1 (2074 symbols)	98.5% 94% 91%
[8]	Handwritten mathematical expression		√	Elastic matching method	10000 symbols	97%

Table III: Confusion matrix with feature vector 23 using SVM

CM	¬	Λ	V	~	&	!	|	=>	(	)	Accuracy
¬	50	0	0	0	0	0	0	0	0	0	100
Λ	0	42	8	0	0	0	0	0	0	0	84
V	0	0	50	0	0	0	0	0	0	0	100
~	0	0	0	50	0	0	0	0	0	0	100
&	0	0	0	0	50	0	0	0	0	0	100
!	0	0	0	0	0	50	0	0	0	0	100
|	0	0	0	0	0	1	49	0	0	0	98
=>	0	0	0	0	0	0	0	50	0	0	100
(	0	0	0	0	0	0	0	0	50	0	100
)	0	0	0	0	0	0	0	0	0	50	100

Table IV: Confusion matrix with feature vector 26 using SVM

CM	¬	Λ	V	~	&	!	|	=>	(	)	Accuracy
¬	50	0	0	0	0	0	0	0	0	0	100
Λ	0	46	2	0	2	0	0	0	0	0	92
V	0	16	32	2	0	0	0	0	0	0	64
~	0	0	0	50	0	0	0	0	0	0	100
&	0	0	1	0	49	0	0	0	0	0	98
!	0	0	0	0	0	50	0	0	0	0	100
|	0	0	0	0	0	6	44	0	0	0	88
=>	0	0	0	0	0	0	0	50	0	0	100
(	0	0	0	0	0	0	0	0	50	0	100
)	0	0	0	0	0	0	0	0	0	50	100