Machine Learning Approach for Recognition of Mathematical Symbols

Iffath Fathima S
P.G Student, Computer Science and Engineering, Bapuji Institute of Engineering and Technology
BIET, Davangere
Karnataka, India

Ashoka K
Assistant Professor, Computer Science and Engineering, Bapuji Institute of Engineering and Technology
BIET, Davangere
Karnataka, India

Abstract

Handwritten character or symbol recognition is one of the application domain in pattern classification. It is generally easy for a person to recognize handwritten or printed characters and symbols but it is difficult for a computer to recognize them. This difficulty can be overcome by adopting a machine learning approach .By building a system that recognizes the patterns. Pattern classification involves features extraction, concept behind the observation (label) and classifier. For this, character geometry as feature extraction technique and two classifiers Support Vector Machines (SVM) and K-nearest neighbour (KNN) are used. Two classifiers are used for the comparative analysis.

Keywords *– SVM, KNN, Character geometry, Machine Learning*

I. Introduction

Pattern recognition forms the basis of learning for all living things in nature. It is generally easy for a person to differentiate a handwritten number "5," from an "4"; However, it is difficult for a programmable computer to solve this kind of perceptual problem. This problem is difficult because each pattern usually contains a large amount of information, and the recognition problems typically have an inconspicuous, high-dimensional, structure. Pattern recognition is the science of making inferences from perceptual data, using tools from statistics, probability, computational geometry, machine learning, signal processing, and algorithm design. Pattern classification involves features extraction and classification. Pattern is defined as composite of features that are characteristic of an individual. In classification, a pattern is a pair of variables {x, w} where x is a collection of observations or features (feature vector) and w is the concept behind the observation (label). The quality of a feature vector is related to its ability to discriminate examples from different classes. Examples from the same class should have similar feature values and while examples from different classes having different feature values. Feature can be defined as any distinctive aspect, quality or characteristic which, may be symbolic (i.e., color) or numeric (i.e., height). The combination of d features is represented as a d-dimensional column vector called a feature vector. The d-dimensional space defined by the feature vector is called feature space. Objects are represented as points in feature space. The goal of a classifier is to partition feature space into class-labeled decision regions.

I-1 K-NEAREST NEIGHBOUR

In pattern recognition, the ***k*-nearest neighbour algorithm** (***k*-NN**) is the input consists of the *k* closest training examples in the feature space. In *k-NN classification*, the output is a class membership. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its *k* nearest neighbors (*k* is a positive integer, typically small). If $k = 1$, then the object is simply assigned to the class of that single nearest neighbor. *K-NN* is a type of instance-based learning, or lazy learning, where the function is only approximated locally and all computation is deferred until classification. The *k*-NN algorithm is among the simplest of all machine learning algorithms. Both for classification and regression, it can be useful to assign weight to the contributions of the neighbors, so that the nearer neighbors contribute more to the average than the more distant ones. For example, a common weighting scheme consists in giving each neighbor a weight of $1/d$, where d is the distance to the neighbor. The neighbors are taken from a set of objects for which the class (for *k*-NN classification) or the object property value (for *k*-NN regression) is known. This can be thought of as the training set for the algorithm, though no explicit training step is required.

I-2 SUPPORT VECTOR MACHINE

In machine learning, support vector machines (SVMs, also support vector networks) are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier (although methods such as Platt scaling exist to use SVM in a probabilistic classification setting). An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall.

II. Problem Definition

Today, as everything is getting digitized, there is a need to put the offline documents, books, notes and literature work in digital form. But this is a complicated task because the offline documents will be in printed or handwritten form. The task required here is to make computer understand the characters within the documents. This can be done by developing a system that recognizes the characters within the documents. This paper concentrates on offline handwritten and printed mathematical symbols. Recognition of Handwritten symbols is a complicated task due to the unconstrained shape variations, different writing style and different kinds of noise. The mathematical symbols recognition is not an easy task because unlike character recognition dataset, this mathematical symbols dataset includes not only the characters from the English alphabets but also the letters from Latin, Greek in addition to those it also contains the numerals, as well as other symbols. Hence handwritten mathematical symbols recognition is an major area of concern. The proposed work mainly focuses on reducing the amount of computation by extracting the relevant features for the efficient classification. for the classification ,two classifiers are used in the proposed system and their comparative analysis is done.

III. Character Geometry Feature Extraction Technique

Character geometry technique follows the steps described below

Universe of Discourse: At first, the universe of discourse is selected because the features extracted from the character image include the positions of different line segments in the character image.

Zoning: The image is divided into windows of equal size and feature extraction is applied to each individual zone rather than the whole image. In our work, the image was partitioned into 9 equal sized windows. Starters, Intersections and Minor Starters: To extract different line segments in a particular zone, the whole skeleton in that zone should be traversed. For this reason, particular pixels in the character skeleton are treated as starters, intersections and minor starters.

Character traversal: Character traversal starts after zoning by which line segments in each zone are extracted. The first step is the starters and intersections in a zone are identified and then occupied in a list. Then the algorithm starts by considering the starter list. Once all the starters are processed, minor starters find along the course of traversal are processed. The positions of pixels in each of the line segments obtained during the process are stored. After visiting all the pixels in the image, the algorithm stops.

Distinguishing the line segments: After all the line segments in the image are extracted, they are classified into any one of the following line-types – Horizontal line, Vertical line, Right-diagonal line, or Left-diagonal line.

Feature Extraction: After the line type of each segment is determined, feature vector is formed based on this information.

IV. Architecture Of The Proposed System

Input image: Input picture can be a picture comprising of either a character, image or scientific expressions. The info picture ought to be in a .png augmentation necessarily.

Preprocessing of input image: Picture pre-dealing is mandatory for any photo based applications. The precision and consolidating rate of such frameworks must be on a very basic level high in order to ensure the achievement of the resulting strides. In any case, more often than not, the noteworthiness of these systems stay unnoticed which brings about second rate results. The objective of the pre-handling step is to construct usable information.

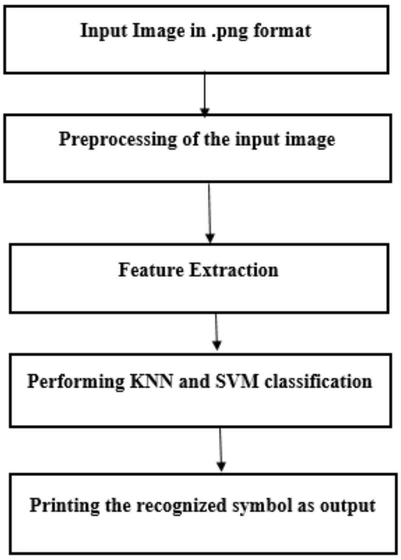
Input Image in .png format
Preprocessing of the input image
Feature Extraction
Performing KNN and SVM classification
Printing the recognized symbol as output

Fig.1. Architecture of the Proposed System

Feature extraction: This is a sort of dimensionality reducing that successfully addresses interesting parts of a photo as a littler component vector. This approach is useful when picture sizes are huge and a reduced segment depiction is required to quickly whole assignments, for instance, picture organizing and recovery. Character Geometry Feature Extraction Technique is used for the component extraction since it is one which support the unmistakable component extraction strategies.

Performing SVM and KNN classification: K-NN count is a versatile and clear estimation which arranges the given get ready cases in perspective of its neighbors. For gathering of the new case this count figures the Euclidian division with the new representation and recognizes its neighbors after that it consigns the class in light of the k regard .k regard will be customer portrayed. This estimation reestablishes the class that addresses the most outrageous of the k cases.
SVMs (Support Vector Machines) are an important technique for data classification. A classification task generally incorporates disengaging data into get ready and testing sets. Each event in the planning set contains one "target regard" (i.e. the class names) and a couple of "attributes" (i.e. the components or watched factors). The goal of SVM is to convey a model (in light of the arrangement data) which predicts the target estimations of the test data given only the test data properties. SVM requires that each data case is addressed as a vector of real numbers. Scaling before applying SVM is critical. The rule of SVM relies upon a straight separation in a high estimation feature space where data are mapped to consider the conceivable non-linearity of the issue

Printing the recognized symbol as Output: The expression, symbol or character given as the input picture is perceived and imprinted in the command window. Recognized image as yield is a consequence of the considerable number of procedures clarified above. The perceived image will be in the form which can be perceived by the PC, human and machines.

V. Experimental Results

Here we have used MATLAB as our computer language, here dataset contains both printed and handwritten images.
Mathematical Symbol dataset not only contains different symbols but also contains alphabets of English, greek and latin languages. For the handwritten mathematical symbol/character recognition system, an own dataset is used. This own dataset is prepared for Mathematical Symbols by considering all possible constraints such as variations in writing styles, samples consisting with some noise, etc.

For the printed images, the mathematical symbols are taken from the standard Infty MDB-1 dataset [10]. Here 26 printed images are examined which covers about 31 symbols, digits from 0-9, English alphabets both uppercase and lowercase letters and 9 formulae. and 28 handwritten images are examined which covers digits from 0-9, English alphabets both uppercase and lowercase letters, 39 symbols and 10 formulae.

The results are mainly shown in the form of graphs, the pie charts shows the accuracies of SVM and KNN for both printed and handwritten images respectively. Performance of both SVM and KNN are shown through bar graphs in Fig 4 and Fig 7 .

Performance analysis is done by calculating the accuracy of each classifier for both printed and handwritten images. Accuracy is derived using the formula given below,

Accuracy = (correct prediction/ total values (supplied)) * 100.

It is found from the experimental results that the accuracy of k-NN is high in most of the cases. But as size of dataset increases we can see accuracy of both system decreases. It is evident from the result that the overall accuracy of KNN is greater compared to SVM.

The below table 1 and table 2 gives the details related to printed and handwritten images respectively and the graphs gives their Accuracies and related performances.

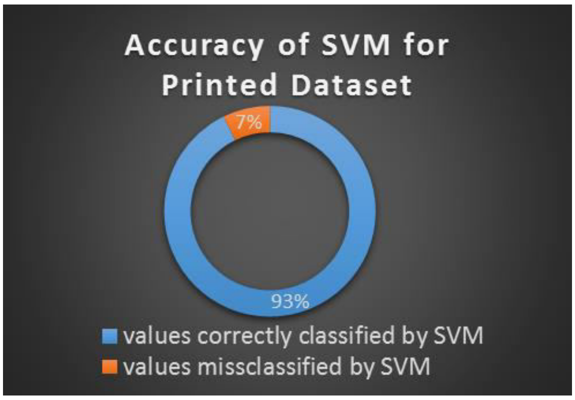

Fig.2. Accuracy of SVM for Printed Dataset

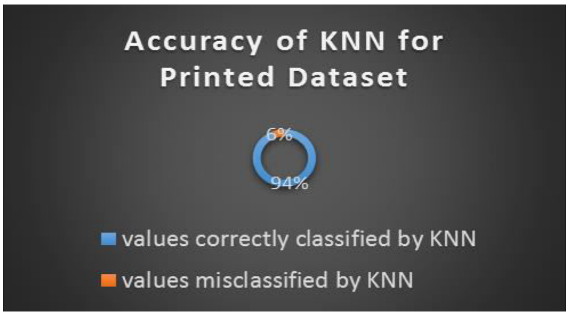

Fig.3. Accuracy of KNN for Printed Dataset

V-2 CLASSIFICATION OF INPUT IMAGES FOR PRINTED IMAGES DATASET

	Input image	SVM	KNN
1	0 1 2 3 4 5 6 7 8 9	100%	100%
2	A B C D E	100%	100%
3	F G H I J	100%	100%
4	K L M N O	100%	100%
5	P Q R S T	100%	100%
6	U V W X Y Z	100%	100%
7	a b c d e	100%	100%
8	f g h i j	40%	60%
9	k l m n o	100%	100%
10	p q r s t	100%	100%
11	u v w x y z	100%	100%
12	β u * λ -	100%	100%
13	( ) + ^ γ	100%	100%
14	σ → ◯ ~ ,	100%	100%
15	∏ ɳ ∫ ∞ Δ	100%	100%
16	. ↔ Θ X √	80%	80%
17	A + B * C	100%	100%
18	< k (ɳ)	100%	100%
19	u E { 1, . . , n – 3 }	100%	100%
20	a + b * c	100%	100%
21	2 n – 1	100%	100%
22	t a n Θ	75%	75%
23	a + c	100%	100%
24	∫ 1	100%	100%
25	ʑ (4)	100%	100%
26	! @ # ; : =	33%	33%

Table:1 Classification of input images for printed images dataset

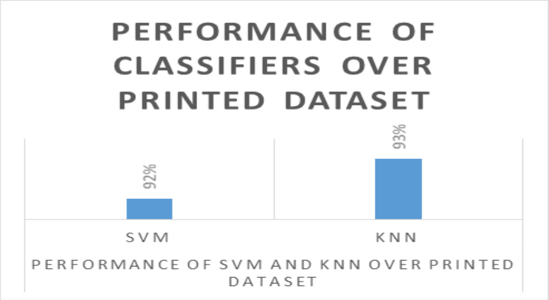

Fig.4. Performance of SVM and KNN classifiers over Printed Dataset

V-2 CLASSIFICATION OF INPUT IMAGES FOR HANDWRITTEN IMAGES DATASET

	Input image	SVM	KNN
1	0 1 2 3 4 5 6 7 8 9	100%	100%
2	A B C D E	100%	100%
3	F G H I J	100%	100%
4	K L M N O	100%	100%
5	P Q R S T	100%	100%
6	U V W X Y Z	100%	100%
7	a b c d e	100%	100%
8	f g h i j	40%	60%
9	k l m n o	100%	100%
10	p q r s t	100%	100%
11	u v w x y z	100%	100%
12	β u * λ -	100%	100%
13	( ) + ^ γ	100%	100%
14	σ → ◯ ~ ,	100%	100%
15	∏ ɳ ∫ ∞ Δ	100%	100%
16	. ↔ Θ X √	80%	80%
17	A + B * C	100%	100%
18	< k (ɳ)	60%	60%
19	u E { 1, . . , n – 3 }	100%	100%
20	a + b * c	100%	100%
21	2 n - 1	25%	100%
22	t a n Θ	25%	100%
23	a + c	100%	100%
24	∫ 1	100%	100%
25	ʑ (4)	100%	100%
26	! @ # ; : =	33%	33%
27	2 x – 3 y	100%	80%
28	Rs B ₡ ₤ Ɏ $ ₳ €	37.5	100%

Table:2 Classification of input images for Handwritten images dataset

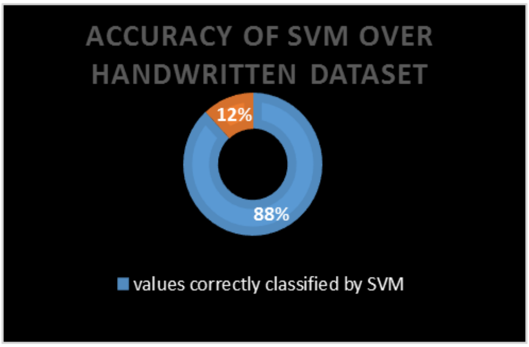

Fig.5. Accuracy of SVM for Handwritten Dataset

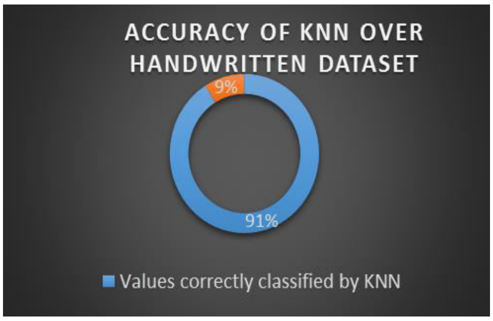

Fig.6. Accuracy of KNN for Handwritten Dataset

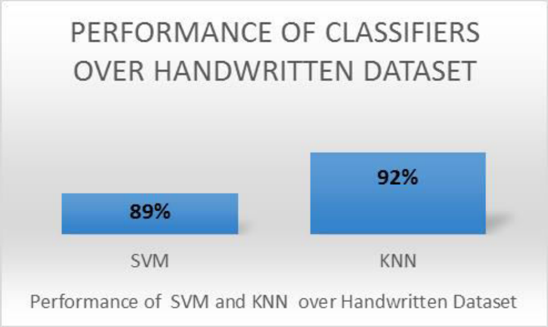

Fig.7. Performance of SVM and KNN classifiers over Handwritten Dataset

VI. Conclusion

Among different feature extraction techniques, Character geometry is selected as the feature extraction technique. Character geometry feature extraction technique is one which supports other different feature extraction techniques. SVM and K-NN classifiers are used for the classification. After numerous executions it has been found that k-nn classifier has great accuracy compared to SVM as classifier. The efficiency of K-NN decreases with the increase in dataset.

VII. Future Work

In future, this above technique can be attempted against standard databases. Likewise more tests could be directed with extra benchmark datasets. By using efficient segmentation technique, offline handwritten mathematical expressions with respect to superscript and subscript could be recognized efficiently.

References

[1] Richard Zanibbi and Dorothea Blostein, "Recognition and retrieval of mathematical expressions", IJDAR, Springer-Verlag, 2011.

[2] Zhao Xuejun, Liu Xinyul, Zheng Shenglingl, Pan Baochang and Yuan Y.Tang, "On-line Recognition Handwritten Mat hematical Symbols", IEEE, 1997.

[3] Dorothea Blostein and Ann Grbavec, " Handbook on Optical Character Recognition and Document Image Analysis", (Chapter22)-"Recognition of Mathematical Notation", World Scientific Publishing Company, 1996.

[4] Erik G. Miller and Paul A. Viola. Ambiguity and constraint in mathematical expression recognition. In

AAAI '98/IAAI '98: Proceedings of the fifteenth national/tenth conference on Artificial intelligence/Innovative applications of artificial intelligence, pages 784–791, Menlo Park, CA, USA, 1998. American Association for Artificial Intelligence.

[5] Taik Heon Rhee and Jin Hyung Kim. Efficient search strategy in structural analysis for handwritten mathematical expression recognition. Pattern Recognition, 42(12):3192 – 3201, 2009. New Frontiers in Handwriting Recognition.

[6] Zi-Xiong Wang and C. Faure. Structural analysis of handwritten mathematical expressions. In *9th International Conference on Pattern Recognition, 1988.*, volume 1, pages 32 –34, nov. 1988.

[7] Y. Eto and M. Suzuki. Mathematical formula recognition using virtual link network. In *6th International Conference on Document Analysis and Recognition, 2001. Proceedings.*, pages 762 –767, 2001.

[8] J. S. Raikwal and Kanak Saxena ," Performance Evaluation of SVM and K-Nearest Neighbor Algorithm over Medical Data set " In *International Journal of Computer Applications (0975 – 8887) Volume 50 – No.14, July 2012*M.

[9] Anchal Tomar , Anshika Nagpal ,"Comparing Accuracy of K-Nearest-Neighbor and Support-Vector-Machines for Age Estimation",International Journal of Engineering Trends and Technology (IJETT) – Volume 38 Number 6- August 2016.Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[10] http://www.inftyproject.org

AUTHOR PROFILE

Iffath Fathima S
P.G student,BIET Davangere,Karnataka,India

Ashoka K
Assistant Professor,BIET Davangere,Karnataka,India