JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

A Handwritten Mathematical Symbol Recognition Using Support Vector Machines with a Linear Kernel method

Girish S. Katkar[1] , Pooja C. Ukey[2], Ajay S .Ramteke [3]

[1]Assistant Professor, Department of Computer Science, Taywade College, Mahadula, Koradi

[2]Research Scholar, Taywade College, Mahadula, Koradi, Dist.-Nagpur

[3]Assistant Professor,Department of Computer Science, Taywade College,Mahadula, Koradi

Abstract: In Pattern Recognition is highly used because of its capability. One of the most important tasks is classification for different applications like text categorization, data classification, image classification etc. Handwritten Mathematical Symbol Classification is a process which decodes handwritten symbols by machine. In this paper, a Support Vector Machine is applied. It is a powerful machine model used for classification. This paper represents recognition techniques for training machine on handwritten images. The SVM model achieved an overall accuracy of high degree of accuracy on the test set.

IndexTerms - Hyperplane, Pattern Recognition, Support Vector Machine

INTRODUCTION

For classification Support Vector Machine is used. It is Supervised machine learning algorithm. Supervised learning is a machine learning technique uses labelled datasets for training algorithms to recognize patterns and predict outcomes. Support Vector Machine is a classification technique developed by Cortes and Vapnil. It is designed for binary classification. While traditional methods pattern recognition is trying to minimize the empirical risk. It is also tries to minimize structural risk, upper bound of generalization error by maximizing the distance between the separating hyperplanes and data points. It summarizes the information from the training data that makes classification much faster than the traditional empirical risk minimization techniques because of its generalization capability even in higher dimension. Support Vector Machine used in Object detection, Handwritten character recognition, Face recognition and Speech recognition. In this paper, we will focus on the recognition of handwritten mathematical symbols by using linear kernel.

Support Vector Machine

Support Vector Machines are effective because they focus on finding the maximum separating hyperplane between the different classes in the target feature and making them powerful for both binary and multiclass classification. It can be applied to regression problems but SVM is best suited for classification tasks. The objective of the SVM algorithm is to identify the optimal hyperplane in an N-dimensional space that can effectively separate data points into different classes in the feature space. The algorithm ensures that the margin between the closest points of different classes known as support vectors is maximized.

a binary classification problem with two classes, labelled as +1 and -1. The training dataset consists of input feature vectors X and their corresponding class labels Y.

The equation for the linear hyperplane can be written as:

$$w^T x + b = 0$$

The vector W represents the normal vector to the hyperplane. The parameter b in the equation represents distance of the hyperplane from the origin along the normal vector w.

The distance between a data point x_i and the decision boundary can be calculated as:

$$d_i = \frac{w^T x_i + b}{||w||}$$

where ||w|| represents the Euclidean norm of the weight vector w. Euclidean norm of the normal vector W

- For Linear SVM classifier:

$$\hat{y} = \begin{cases} 1 & : \ w^T x + b \geq 0 \\ 0 & : \ w^T x + b < 0 \end{cases}$$

Optimization:

- For Hard margin linear SVM classifier:

$$\underset{w,b}{\text{minimize}} \frac{1}{2} w^T w = \underset{W,b}{\text{minimize}} \frac{1}{2} \|w\|^2$$
$$\text{subject to } y_i(w^T x_i + b) \geq 1 \ for\ i = 1, 2, 3, \cdots, m$$

The target variable for the ith training instance is denoted by the symbol ti in this statement, ti=-1 for negative occurrences (when yi= 0) and ti=1 positive instances (when yi = 1) respectively. Because we need the decision boundary that satisfy the constraint:

- For Soft margin linear SVM classifier:

$$\underset{w,b}{\text{minimize}} \ \frac{1}{2} w^T w + C \sum_{i=1}^{m} \zeta_i$$
$$\text{subject to } y_i(w^T x_i + b) \geq 1 - \zeta_i \ \ and\ \zeta_i \geq 0 \ \ for\ i = 1, 2, 3, \cdots, m$$

A linear kernel support vector machine (SVM) is a linear classifier can be used for pattern recognition to separate two classes of a data by using a hyperplane. A linear kernel SVM is a supervised learning model. It creates a hyperplane to separate data into positive and a negative category. The hyperplane and margins are straight lines in the untransformed feature space. A linear kernel SVM is an effective for linear classification problems and the data can be separated with a line.

The mathematical symbols recognition is very difficult and these points conclude here:

1) In mathematical notations a large number of symbols are used.
2) The variety of styles used in single formulas different styles for symbols.
3) Large difference in the size of symbols.

.

Recognition of Handwritten Mathematical Symbols

I. Data Collection and Preprocessing

The dataset collected from various individuals to ensure diversity in handwriting styles. The dataset used in this paper consists of handwritten mathematical symbols. The dataset is divided into two parts: a training set and a testing set. Each image is organized into folders corresponding to the different symbols that are classified (e.g., addition, division).

1. **Image Preprocessing**: In the Dataset each image first converted into grayscale to reduce computational complexity. To ensure uniformity across the dataset, grayscale images are resized to a fixed dimension of 28x28 pixels. The image cut into one-dimensional arrays to work as inputs for the machine learning model.
2. **Data Normalization**: Pixel values normalized to a range between 0 and 1 by dividing each pixel by 255 to improve model performance.
3. **Training and Test Split**: After the preprocessing, the dataset is divided into two sets i.e. training and testing sets with 80% of the data for training and 20% for testing. The training set used to fit the machine learning model and the testing set reserved for evaluation.

II. Model Selection

For classification a Support Vector Machine (SVM) is used. The SVM is a widely-used supervised learning model and effective in high-dimensional spaces making it suitable for image classification tasks. A linear kernel to simplify the model and reduce training time.

II. Model Training

The SVM model trained on the pre-processed training set. The linear kernel was selected to maintain interpretability. Other kernels such as polynomial and radial basis function (RBF) could be used for comparison in future work. The SVM is optimized by fitting it to the 28x28 pixel image data with the corresponding labelling of symbols (e.g., addition, subtraction, multiplication).

III. Testing and Prediction

To evaluate the performance of the model, use the test set. Additionally, provided a function to predict the label of a single image by preprocessing it in the same way as the training data (grayscale conversion, resizing, flattening, and normalization). The SVM model used to predict the class of the given symbol image.

Results

The performance of the SVM model was evaluated using various metrics such as accuracy, confusion matrix, and classification report. After training the model with the 80% training data and tested on the 20% testing set. The model's performance across different classes of symbols was analyzed.

1. **Classification Accuracy**: The SVM model achieved an overall high degree of accuracy on the test set. It indicates the percentage of correctly predicted symbol images out of the total test images.

```
The predicted label for the test image is: divide
```

```
Accuracy of the model on test data: 100.00%
```

2. **Confusion Matrix**: The confusion matrix (see Figure 1) shows the number of correct and incorrect predictions for each symbol class. From the matrix, it is observed that the model performed well on distinguishable symbols like '+' and '×'.

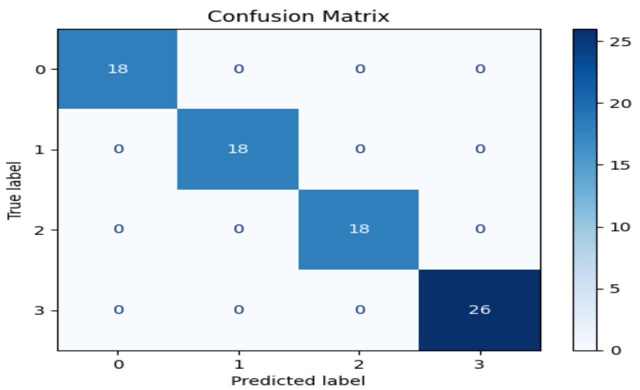

Fig 1: Confusion Matrix

3. **Classification Report**: The classification report (Table 1) provides precision, recall, and F1-score for each symbol. The precision and recall values for the majority of the symbols were above 0.90, reflecting the robustness of the SVM

classifier. However, certain symbols, particularly those that are visually similar, had lower scores, highlighting areas for future improvement.

Classification Report:

	precision	recall	f1-score	support
divide	1.00	1.00	1.00	18
minus	1.00	1.00	1.00	18
multiply	1.00	1.00	1.00	18
plus	1.00	1.00	1.00	26
accuracy			1.00	80
macro avg	1.00	1.00	1.00	80
weighted avg	1.00	1.00	1.00	80

Table 1: Classification Report

Conclusion

This paper presented a method for classifying handwritten mathematical symbols using a Support Vector Machine with a linear kernel. Preprocessing the images into grayscale, resizing them, and normalizing pixel values, we achieved a streamlined and effective input representation for the SVM model. The results show that the SVM classifier can accurately recognize and classify handwritten symbols with a high degree of accuracy.

However, we observed misclassifications between visually similar symbols such as division and subtraction, indicating room for improvement. Future work could involve comparing the performance of different kernels (such as RBF or polynomial) or incorporating convolutional neural networks (CNNs), which are known to excel in image classification tasks.

In summary, SVM proves to be a viable method for handwritten symbol classification, achieving promising results with minimal preprocessing, but further refinement could boost performance on visually similar classes of symbols.

References

[1] F. Alvaro, J.-A. Sanchez, and J. Benedi, "Recognition of printed mathematical expressions using twodimensional stochastic contextfree grammars," in International Conference on Document Analysis and Recognition, pp. 1225 –1229, sept. 2011.

[2] F. Alvaro and J. Sanchez, "Comparing several techniques for offline recognition of printed mathematical symbols," in 20th International Conference on Pattern Recognition, pp. 1953 –1956, Aug. 2010.

[3] T.Y. Zhang and C.Y. Suen, "A fast parallel algorithm for thinning dig ital patterns", Comm. of the ACM 27, 1984, pp. 236-239.

[4] H. Kumar and P. Kaur. "A Comparative Study of Iterative Thinning Algorithms for BMP Images " in International Journal of Computer Science and Information Technologies, Vol. 2 (5), 2011, 2375-2379.

[5] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," Int. J. Comput. Vision, vol. 42, pp. 145– 175, May 2001.

[6] A Bosch, A. Zisserman and X. Munoz. Representing shape with a spatial pyramid kernel. Proceedings of the 6th ACM international conference on Image and video retrieval.NK, USA, 2007:401-408.

[7] Y. Alginahi, "Pre-processing techniques in character recognition", vol.1.

[8] M. Bharambe, "*Segmentation of offline Printed and Handwritten Mathematical Expressions*" National Conference on Digital Image and Signal Processing, vol.1, pp.0975-8887,2016.