IJCSE International Journal of Computer Sciences and Engineering Open Access

Research Paper Vol.-7, Issue-2, Feb 2019 E-ISSN: 2347-2693

A Study on Text Recognition using Image Processing with Datamining Techniques

U.Karthikeyan[1*], M. Vanitha[2]

[1]Department of Computer Applications, Alagappa University, Karaikudi, Tamilnadu, India
[2]Department of Computer Applications, Alagappa University, Karaikudi, Tamilnadu, India

**Corresponding Author: kartikeyan.u@gmail.com, Tel.: +91- 81482-21832*

DOI: https://doi.org/10.26438/ijcse/v7i2.783787 | Available online at: www.ijcseonline.org

Accepted: 19/Feb/2019, Published: 28/Feb/2019

Abstract**—** Text recognition is a technique that recognizes text from the paper document in the desired format (such as .doc or .txt). The text recognition process involves several steps, including pre-processing, segmentation, feature extraction, classification, and post-processing. The preprocessing is performed as a binarized image to convert a grayscale image, and noise is reduced on the input image of the basic operation performed by removing the noise of the image signal. The segmentation phase is used to segment the image given online and segment each character of the segmentation line. Feature extraction is to compute the characteristics of the image document. This document describes techniques for converting the textual content of a paper document into a machine-readable format. This paper analyzes and compares the technical challenges, methods, and performance of text detection and recognition studies in color images. It summarizes the basic issues and lists the factors that should be considered when addressing them. The prior art is classified as step-by-step or integrated and highlights sub-problems including text localization, verification, segmentation and identification of text. This survey provides a basic comparison and analysis of the scope and challenges in the field of text recognition.

Keywords**—** Classification , Datamining, Segmentation, Text recognition

I. Introduction

Text recognition is important for a lot of applications like automatic sign reading, navigation, language translation, license plate reading, content-based image search etc. So it is necessary to understand scene text than ever. Texts in images carry high-level semantic information of the scene. Images in the webs and database are increasing. Developing effective ways to manage and restore the content of these resources is an urgent task. With the rapid growth of digital technology and devices manufactured by megapixel cameras and other devices such as Personal Digital Assistants (PDA), mobile phones, etc., are responsible for increasing the attention for information retrieval and it leads to a new research task.

Texts, in the images, contain valuable information and provide cues about images. So it is very important for a human as well as the computer to understand the scenes. It is a complex method to recognize and segment text from the scene or captured images for many reasons like different types of text patterns like font size, style, orientations, colors, background outlier similar to the text characters. Text recognition is applied after the detection of text from the image and segmentation to convert the image into readable text, but it performs inadequately when there is a text on the complex background. The MATLAB, ORANGE, KNIME, WEKA are the most popular open source tools used for the field of text recognition in data mining.

Obtaining high accuracy in character recognition is a challenging task. Several factors like background noise, variations in character size, width, pen ink, character spacing, skew and slant, similarity of some characters in shape and size, influence the character recognition rate. Other significant factors, for instance, the absence of header line or segmentation of modifiers and touching characters also become significant in design an efficient character recognition system. Generally, text recognition is the scanned pictures of pre-written textual material on paper and online text identification are the throughout writing activity on a specially designed pen in an electronic device. Recognition of documents has been a vital field for research in the broad domain of pattern recognition. Over the past few years, various laboratories all over the world showed their intense involvement studies on text recognition. The main goal of this paper is to evaluate the character from the given image which is text written image and print on the document or word file.

Section II contains the literature survey of the proposed text recognition method, Section III contains some methods and materials used for the present study, and Section IV concludes present study with directions for further work.

II. Literature Survey

C. Patel and A. A. Desai [1] have proposed segmentation of text lines into words. They have used projection profile and morphological operations for segmentation. They have proposed zone identification for words. They have used distance transform method for identification of zone like upper, middle, and lower. They have proposed a handwritten character recognition system. They have used hybrid classifier using tree and k-NN. They have used structural and statistical features. They have achieved an accuracy of 63%.

A. A. Desai [4] has proposed character segmentation from old documents. He has used some pre-processing methods and Radon transform for segmentation. He has proposed a character recognition system for Gujarati numerals. He has used binarization, size normalization and thinning pre-processing methods. He has used hybrid features like a subdivision of skeletonized image and aspect ratio. He has used k-NN classifier with Euclidean distance method and achieved 96.99% accuracy. He has proposed similar work using profile vector-based features. He has used a multilayer feed forward neural network. He has achieved an accuracy of 82%.

M. Maloo and K. V. Kale [9] have proposed a handwritten numeral recognition system for Gujarati. They have used pre-processing methods like binarization, dilation, and skeletonization. They have used affine invariant moments (AMI) for feature extraction and SVM for classification and achieved 91% accuracy.

M. B. Mendapara and M. M. Goswami [10] have used binarization, noise removal, and thinning pre-processing methods. They have used stroke based directional feature and used k-NN as a classifier. They have achieved 88% accuracy.

R. Nagar and S. Mitra [11] have used binarization and thinning pre-processing methods. They have used orientation estimation features and SVM as a classifier and achieved 98.97% accuracy.

A. Vyas and M. Goswami [13] have used binarization, noise removal, and thinning pre-processing methods. They have used modified chain code, Discrete Fourier Transform, and Discrete Cosine Transform as a feature. They have used k-NN, SVM and ANN as a classifier and achieved 85.67%, 93.60%, and 93.00% accuracy respectively.

Prutha Y M and Anuradha SG [14] have proposed a real-time traffic analysis system. They have used different morphological and edge detection techniques.

In Malayalam online handwritten character recognition, S. Joseph and A. Hameed [17] have used basic preprocessing methods and used six-time domain features with directional and curvature features. They have used SVM as a classifier and achieved 95.45% accuracy.

Anoop M. Namboodiri [18] have presented work on Malayalam and Telugu language. They have used normalization, resampling using a Gaussian low-pass filter and an equidistant resampling to remove variations in writing speed. They have used moments of the stroke, direction, curvature, length, an area of the stroke, aspect ratio as features. They have used SVM using a Decision Directed Acyclic Graph (DDAG) and discriminative classifier. They have achieved an accuracy of 95.78% on Malayalam and 95.12% on Telugu.

Primekumar K.P. and S. Idiculla [19] have used duplicate point elimination, smoothing, normalization, resampling as preprocessing methods. They have used x-y coordinates, angular features, direction, and curvature are extracted. Using HMM classifier, they have used k means using Euclidean distance for training and using SVM classifier, they have used discrete wavelet transform for training. They have achieved an accuracy of 97.97% using SVM and 95.24% using HMM.

III. Materials and Methods

In this study various existing and commonly used techniques are listed below by surveying many research papers for image acquisition, pre-processing, segmentation, feature extraction, classification, and post-processing.

Table 1. Major Stages of the Text Recognition System

Stage	Definition	Techniques
Image Acquisition	Acquiring or Capture the image	Binarization, Digitalization Resizing, Compression
Pre-processing	Enhance the quality of an image	Noise removal, Filtering Skew, Thinning, Edge detection and correction, Morphological operation
Segmentation	Splitting image into characters or words	Character based, Word-based, Sentence based
Feature Extraction	Extracting characteristics of an image	Statistical and geometrical features

Classification	Extracting characters are in a category	Decision tree, SVM, Nearest neighbor, Distance-based methods
Post processing	Increase the performance accuracy of text prediction	Confusion matrix, Contextual approaches Dictionary-based approaches

Table 2. Merits and Demerits of the Text recognition system

S.No.	Algorithm	Merits	Demerits
Regression algorithms			
1	Linear Regression	Space complexity is very low it just needs to save the weights at the end of training. Hence it's a high latency algorithm. It’s very simple to understand Good interpretability Feature importance is generated at the time model building.	The algorithm assumes data is normally distributed in real they are not. Before building model multicollinearity should be avoided. prone to outliers
2	Logistic Regression	It is more robust: the independent variables don’t have to be normally distributed, or have equal variance in each group. It may handle nonlinear effects	It can’t solve non-linear problems with logistic regression since it’s decision surface is linear. Prone to overfitting
3	Autoregressive Integrated Moving Average (ARIMA)	The solid underlying theory, stable estimation of time-varying trends and seasonal patterns, relatively few parameters.	No explicit seasonal indices, hard to interpret coefficients, the danger of overfitting or misidentification if not used with care.
4	Multivariate Adaptive Regression Splines	Works well even with a large number of predictor variables Automatically detects interactions between variables Efficient and fast Robust to outliers	Difficult to understand Prone to overfitting Model is vulnerable to missing data
Instance-based algorithms			
5	K-Nearest Neighbor (KNN)	The simple technique that is easily implemented Building model is cheap An extremely flexible classification scheme Well suited for Multi-modal classes, Records with multiple class labels	Classifying unknown records are relatively expensive. Accuracy can be severely degraded by the presence of noisy or irrelevant features
6	Kernel Regression	It is nonparametric	Prone to bias if the independent variables are not uniformly distributed
7	Support Vector	SVM's can model non-linear decision boundaries, and there are many kernels to choose from. They are also fairly robust against overfitting, especially in high-dimensional space.	SVM's are memory intensive, trickier to tune due to the importance of picking the right kernel, and don't scale well to larger datasets.
Decision tree algorithms			
8	Classification and Regression Trees (CART)	They are robust to outliers, scalable, and able to naturally model non-linear decision boundaries thanks to their hierarchical structure.	Unconstrained, individual trees are prone to overfitting, but this can be alleviated by ensemble methods.
9	Iterative Dichotomiser 3 (ID3)	Understandable prediction rules are created from the training data. Builds the fastest tree. Builds a short tree.	Data may be over-fitted or over-classified if a small sample is tested. Only one attribute at a time is tested for making a decision.
10	C 4.5	Builds models that can be easily interpreted Easy to implement Can use both categorical and continuous values Deals with noise	The small variation in data can lead to different decision trees (especially when the variables are close to each other in value) Does not work very well on a small training set
Bayesian algorithms			
11	Naive Bayes	NB models actually perform surprisingly well in practice, especially for how simple they are. They are easy to implement and can scale with the dataset.	Due to their sheer simplicity, NB models are often beaten by models properly trained and tuned using the previous algorithms listed.
12	Bayesian Network (BN)	Have a rigorous probabilistic foundation Reasoning process is semi-transparent	Information theoretically infeasible Computationally infeasible Unautomatic
Clustering algorithms			
13	K-Means	It's fast, simple, and surprisingly flexible if you pre-process your data and engineer useful features.	The user must specify the number of clusters, which won't always be easy to do. In addition, if the true underlying clusters in your data are not globular, then K-Means will produce poor clusters.

S.No.	Algorithm	Merits	Demerits
14	Expectation Maximization (EM)	The likelihood is guaranteed to increase for each iteration. Is a derivative-free optimizer. Is fast if analytical expressions for the M-step are available. Parameter constraints are often dealt with implicitly.	Requires both forward and backward probabilities (numerical optimization requires only forward). Significant implementation effort required compared to numerical optimization.
15	Hierarchical Clustering	The main advantage of hierarchical clustering is that the clusters are not assumed to be globular. In addition, it scales well to larger datasets.	Much like K-Means, the user must choose the number of clusters (i.e. the level of the hierarchy to "keep" after the algorithm completes).
Artificial neural network algorithms			
16	Perceptron	The stochastic nature of the learning process reduces the possibility of getting stuck in local minima Easily takes advantage of redundant data Easy to implement	Cannot be parallelized
17	Back-Propagation	Relatively simple implementation Mathematical Formula used in the algorithm can be applied to any network. Computing time is reduced if the weights chosen are small at the beginning.	Slow and inefficient. A large amount of input/output data is available, but you're not sure how to relate it to the output.
18	Hopfield Network	Massive parallel computation	Computational efficiency is not consistent
Ensemble algorithms			
19	AdaBoost	Easy to implement Not prone to overfitting	Sensitive to noisy data and outliers
20	Random Forest	Reduction in overfitting Less variance	More complex Hard to visualize

IV. CONCLUSION

In this paper, an overview of various text recognition techniques, methods and recognition algorithms has been presented. Based on the literature review various text recognition algorithms accuracy are discussed. The detailed steps and flow of the text recognition techniques by surveying that image acquisition, preprocessing, feature extraction, classification, and post-processing from many research articles. Merits and demerits of text recognition algorithms are discussed. The paper presents a brief survey of the applications in various fields along with experimentation into a few selected fields. This paper will serve as a good survey of researchers who have begun work in the field of character recognition.

REFERENCES

[1] C. Patel and A. Desai, *"Segmentation of text lines into words for Gujarati handwritten text,"* Proc. 2010 Int. Conf. Signal Image Process. ICSIP 2010, pp. 130–134, 2010.

[2] C. Patel and A. Desai, *"Zone identification for Gujarati handwritten word,"* Proc. - 2nd Int. Conf. Emerg. Appl. Inf. Technol. EAIT 2011, pp. 194–197, 2011.

[3] C. Patel and A. Desai, *"Gujarati Handwritten Character Recognition Using Hybrid Method Based on Binary Tree-Classifier And K-Nearest Neighbour,"* Int. J. Eng. Res. Technol., vol. 2, no. 6, pp. 2337–2345, 2013.

[4] A. Desai, *"Segmentation of Characters from old Typewritten Documents using Radon Transform,"* Int. J. Comput. Appl., vol. 37, no. 9, pp. 10–15, 2012.

[5] A. A. Desai, *"Handwritten Gujarati Numeral Optical Character Recognition using Hybrid Feature Extraction Technique,"* Int. Conf. Image Process. Comput. Vision, Pattern Recognition, IPCV, 2010.

[6] A. A. Desai, *"Gujarati handwritten numeral optical character reorganization through neural network,"* J. Pattern Recognit., vol. 43, no. 7, pp. 2582–2589, 2010.

[7] A. a. Desai, *"Support vector machine for identification of handwritten Gujarati alphabets using hybrid feature space,"* CSI Trans. ICT, vol. 2, no. January, pp. 235–241, 2015.

[8] Mayil S. and Vanitha M, *"A Survey on privacy Preserving Data Mining Techniques"*, International Journal of Computer Science and Information Technologies. Vol.5 (5), pp. 6054-6056. ISSN: 0975-9646, 2014.

[9] M. Maloo, K. V Kale, and I. Technology, *"Support Vector Machine Based Gujarati Numeral Recognition,"* Int. J. Comput. Sci. Eng. ({IJCSE}), {ISSN} 0975-3397, vol. 3, no. 7, pp. 2595–2600, 2011.

[10] M. B. Mendapara and M. M. Goswami, *"Stroke identification in Gujarati text using directional feature,"* Proceeding IEEE Int. Conf. Green Comput. Commun. Electr. Eng. ICGCCEE 2014, 2014.

[11] N. Rave and S. K. Mitra, *"Feature extraction based on stroke orientation estimation technique for handwritten numeral,"* in Eighth International Conference on Advances in Pattern Recognition (ICAPR), 2015.

[12] Manimaran R. and Vanitha M, *"An Efficient Study on Usage of Data Mining Techniques for Predicting Diabetes"*, International Journal of Advanced Research Trends in Engineering and

Technology (IJARTET) Vol.3 (20), pp.268-272 ISSN: 2394-3785, 2016.

[13] A. N. Vyas and M. M. Goswami, *"Classification of handwritten Gujarati numerals,"* 2015 Int. Conf. Adv. Comput. Commun. Informatics, ICACCI 2015, pp. 1231–1237, 2015.

[14] Y. M. Prutha and S. G. Anuradha, *"Morphological Image Processing Approach of Vehicle Detection for Real-Time Traffic Analysis,"* Int. J. Comput. Sci. Int. J. Comput. Sci. Eng., vol. 3, no. 5, pp. 88–92, 2014.

[15] M. A. Abuzaraida, A. M. Zeki, and A. M. Zeki, *"Online recognition system for handwritten hindi digits based on matching alignment algorithm,"* in International Conference on Advanced Computer Science Applications and Technologies, ACSAT 2014, 2014, pp. 168–171.

[16] S. Belhe, C. Paulzagade, A. Deshmukh, S. Jetley, and K. Mehrotra, *"Hindi handwritten word recognition using HMM and symbol tree,"* Proceeding Work. Doc. Anal. Recognit. - DAR '12, p. 9, 2012.

[17] S. Joseph and A. Hameed, *"Online handwritten malayalam character recognition using LIBSVM in Matlab,"* in National Conference on Communication, Signal Processing and Networking, NCCSN 2014, 2015, pp. 1–5.

[18] A. Arora and A. M. Namboodiri, *"A hybrid model for recognition of online handwriting in Indian scripts,"* in International Conference on Frontiers in Handwriting Recognition, ICFHR 2010, 2010, pp. 433–438.

[19] K. P. Primekumar and S. M. Idiculla, *"On-line Malayalam Handwritten Character Recognition using HMM and SVM,"* Int. Conf. Signal Process. , Image Process. Pattern Recognit. [ICSIPR], pp. 1–5, 2013.

[20] A. Sampath, C. Tripti, and V. Govindaru, *"Online Handwritten Character Recognition for Malayalam,"* ACM Int. Conf. Proceeding Ser., pp. 661–664, 2012.

[21] G. S. Reddy, P. Sharma, S. R. M. Prasanna, C. Mahanta, and L. N. Sharma*, "Combined online and offline assamese handwritten numeral recognizer,"* in National Conference on Communications, NCC 2012, 2012.

[22] A. Bharath and S. Madhvanath*, "HMM-based lexicon-driven and lexicon-free word recognition for online handwritten indic scripts,"* IEEE Trans. Pattern Anal. Mach. Intell., vol. 34, no. 4, pp. 670–682, 2012.