



Received: 26 July, 2021
Accepted: 03 August, 2021
Published: 04 August, 2021

***Corresponding author:** Hrishikesh Viswanath, Department of Computer Science and Engineering, People's Education Society University, Bangalore, India, Tel: +91 78998 56053; E-mail: hrishikeshv@pesu.pes.edu, hrishi.vish@outlook.com

ORCID: <https://orcid.org/0000-0001-6788-2392>

Keywords: Feed forward neural network; Convolutional neural network; Support vector machine; Epigraphical scripts; Segmentation; Sliding window; CNN; SVM

<https://www.peertechzpublications.com>



Research Article

Study of using hybrid deep neural networks in character extraction from images containing text

P Preethi, HR Mamatha and Hrishikesh Viswanath*

Department of Computer Science and Engineering, People's Education Society University, Bangalore, India

Abstract

Character segmentation from epigraphical images helps the optical character recognizer (OCR) in training and recognition of old regional scripts. The scripts or characters present in the images are illegible and may have complex and noisy background texture. In this paper, we present an automated way of segmenting and extracting characters on digitized inscriptions. To achieve this, machine learning models are employed to discern between correctly segmented characters and partially segmented ones. The proposed method first recursively crops the document by sliding a window across the image from top to bottom to extract the content within the window. This results in a number of small images for classification. The segments are classified into character and non-character class based on the features within them. The model was tested on a wide range of input images having irregular, inconsistently spaced, hand written and inscribed characters.

Introduction

Study of ancient inscriptions is important and vital in reconstructing history. The scripts used in these inscriptions may belong to different eras and are classified based on the dynasty that ruled during these periods. Scripts from southern Indian kingdoms form the subject of this study. All South Indian scripts are derived from Brahmi script, which dates back to the 3 century B.C and the script has continuously evolved over the years. These scripts can be seen on temple walls, copper plates, palm leaves, coins and pots. Epigraphers and paleographers are responsible for drawing conclusions from these inscriptions. However, the study of inscriptions takes a lot of time. The automation of this process and construction of an optical character recognizer is the need of the day [1]. Despite the work on inscriptions having improved significantly, period prediction and recognition is still difficult because of the degraded source material and an absence of experts to interpret the obsolete scripts. Traditional ways of decoding these inscriptions comprise creating estampages, manually reading & storing them for further study.

Some of the challenges that are prevalent in the field of historical document restoration are that the text found on inscriptions and estampages does not follow a standard typeset, with fixed character dimensions and regular letter spacing. The same character may be inscribed in different ways by different sculptors. The text is not in a straight line and is found to be degraded through weathering and physical damages inflicted by humans over the course of history. Furthermore, the characters cannot be processed by any character recognition algorithms because the languages being studied are obsolete or have evolved to use a different script. Therefore, models which extract information from inscriptions using techniques that are not dependent on the language or the script are required in these scenarios.

Machine learning models are widely used in solving real world problems. Although they were introduced a long time ago, due to lack of computational power and availability of significant amount of data, usage of these models was limited. Today however, due to the availability of high end machines and Cloud support, layer wise training has taken leaps in every domain. In the recent times, many researchers have found

ways to adopt machine learning and its variants in document image analysis, text detection, object recognition and face recognition etc.

Segmentation is the process of dividing an image into its meaningful constituents comprising multiple segments, which can later be used for recognition. In our previous work, nearest neighbor algorithm, water shed algorithm and connected component algorithm were applied and we could conclude that connected component algorithm was better suited for segmentation of epigraphical images. The output also contains unwanted elements, prominently, noise and partially cropped incomplete characters. Segmenting such characters is the main challenge of the models. In this paper, we explore the use of text and non-text classifiers, which are modeled to differentiate noise and half cut characters from legible characters. They are trained to extract and write legible segments onto disk.

The table below summarizes the shortcomings of some of the existing techniques that have been used to segment text Table 1.

Literature survey

Script extraction or character spotting from images of stone inscriptions, palm leaves, degraded historical documents and estampages plays a key role in building automated script recognition. There are several research papers describing segmentation procedures applied on images of printed epigraphical scripts, ancient historical documents and handwritten documents. A survey reveals that input considered in the proposed methods have homogeneous background with equal character spacing and these are the primary reasons for achieving high accuracy. We consider images of Estampages (derived from stone inscriptions) as a type of document. Spotting the characters from these images for the purpose of recognition and prediction remains an unresolved problem. This section will review various sources of literature on character segmentation from epigraphical images along with the challenges encountered.

A.S Kavitha, et al. [5] proposed text segmentation on degraded historical images obtained from Indus Valley

civilization remains. Experimentation includes enhancement of input image using a combination of Laplacian and Sobel filters. Closed proximal components of the image are grouped using nearest neighbor method into two clusters - text and non-text. Classifying text and non-text clusters are based on the features of the text. This results in text segmentation of the input image and claims to achieve 90% accuracy.

Printed historical documents that belong to 16th -20th century CE periods are of interest in the paper by Laurence and zahour, et al. [6]. They propose a survey indicating the work done in the field and also explain the importance of epigraphy. A document image undergoes preprocessing to remove noise and is then subjected to segmentation methods like projection profile, bounding boxes and connected components. The methods are applied to images of ancient Arabic documents. The input dataset however dates back to eras as recent as the 16th century, this implies that the data isn't as degraded as older documents and is less complex to interpret.

A novel segmentation algorithm was designed to work on Greek inscriptions. SLIC super pixel with region merging was adopted to work on geometry of the surface. Based on the unscripted surface, strokes and breaks on the surface of the rock, surface points are classified [7]. Murthy, et al. [8] proposed a nearest neighbor clustering approach in segmenting the characters. Epigraphical images used in the proposed method are of printed scripts with irregular spaces, skewed, making the lines intersect in certain regions. The authors focus on character, word and line segmentation. In [9] connected component analysis segments the input image into lines and characters. The segmented characters are subjected to recognition and period prediction. Rui Hu, et al. [10], Propose a method to extract Maya codices from degraded images using region based segmentation. Super pixels having multi resolutions are extracted and classified into foreground or background pixels based on SNM classifier. A fully connected conditional random field model is used to improve label consistency. The model helps in retrieval of characters from the Maya script images.

Mohana, et al. [11] propose character extraction from stone inscriptions. The methods applied include cropping the characters, application of morphological operators to enhance script and finally, extracting SIFT features. The cropped reference image is matched with characters of the input image. A reference image is matched with the query image to calculate the correspondence between foreground pixels and matched pixels. The matching factor calculates the ratio of difference in hit and miss count while scanning to total number of foreground pixels. A threshold is set to identify the matching characters in the stone inscriptions. These methods claim to achieve up to 88% accuracy in segmentation of specific characters.

Michele, et al. [12] proposed a method of initializing deep neural network weights with linear discriminant analysis instead of random values. These values are initialized layer wise and converge faster to yield better performance. The transformation matrix obtained from activated LDA features is used to separate text and non-text regions. Experimentation

Table 1: Summary of existing models that are used for text segmentation.

Method	Description	Shortcomings
Histogram Projection Technique [2,3]	A projection profile of points along the axes is used to split the text into lines. It can be used for handwritten documents	The model may not work well for short lines of text and fails when there's significant overlap between characters
Run Length Smoothing algorithm [2]	Consecutive black pixels that fall within a distance threshold are placed in bounding boxes.	The method only works for printed documents.
Hough Transform based methods [2,4]	These methods are used to find lines of text in an image. They work on unconstrained lines of handwritten text.	The models may perform poorly if the characters are accompanied by accents. These models have performed poorly when there were intersections between ascending and descending strokes.



was conducted on historical medieval period dataset of about 150 pages of scanned images. The author worked on small layered CNNs and advises the use of deeper network for further works.

Chen, et al. [13] describe the use of support vector machine to classify, text, borders, background and periphery in their paper. Super pixels derived from an image are used to extract features based on intensity values into classes. Pinkesh, et al. [14] proposed a bidirectional LSTM model to segment a sentence embedding using a CNN model. The text segments are classified from printed documents.

Junho, et al. [15] in their paper, have implemented convolutional neural networks and have used synthesized samples for training. The model segments handwritten text based on these features. The authors' claim is that the accuracy achieved upon segmentation lies between 75% and 90%. Al-Rawi, et al. [16], describe the use of generative adversarial network to segment text automatically and does not require pixel level annotation of the dataset by constituting reliable unsupervised text segmentation. The data set is mainly composed of scene images containing text. In M M Reza, et al. [17], table segmentation in invoices is the motto for using conditional generative adversarial network for table area localization. Signet based encoder and decoder with skip connections are used for segmentation.

Li, et al. [18], considered character segmentation under complex background and their model classifies the split images into characters and gap images. P-N Learning strategy was adopted to train the CNN, which verified the classification with handwritten text data set. In Zirari, et al. [19], connected component-based approach is used for segmentation. In Manigandan, et al. [20], a binary image was used to segment the characters based on connected components. Sowmya, et al. [21] propose drop fall and water reservoir method to segment the connected characters in historical records of varying complexity. The dataset considered for the model is the set of printed epigraphical characters and works better upon enhancement. In Abtahi, et al. [22], segmentation agent based on reinforcement learning helps in finding the appropriate paths for segmentation along with projection profiles. The input used has one to two text lines and the characters are equally spaced.

Proposed model

Epigraphical images considered for experimentation are estampages from regions covering the modern-day Indian state of Karnataka from various periods from 4th century to 15th century CE. The data has been collected from Archaeological Survey of India, Mysore. The images are inherently noisy due to degradation and the complex texture of the medium on which they are present.

Epigraphical text is harder to segment into letters due to varying and uneven character dimensions. The connected component algorithm uses a graph traversal technique, which works well on image segmentation, but generates image

segments having both text and non-text. The term "non-text" refers to noise, cuts and incomplete text. Feeding the output of the connected component technique will yield poor results by models that are designed for recognition and prediction.

To remedy this issue, input images are segmented into multiple small image segments using regular overlapping rectangular kernels of sizes comparable to text size. The dimension of the sliding window is chosen in such a way that it encompasses one character. The generated image segments are grouped into two classes as correctly segmented text and incorrectly segmented text. The process of classification is done automatically by detecting the presence of white border in a binary image. If such a border exists, it is safe to assume that there is no broken text or partially segmented characters, which if present will extend beyond the segment, thereby making it impossible for the segment to have a clean unbroken border.

This data is fed to various machine learning models for classification and correctly classified characters of the text are then written onto the disk. Machine learning models considered for experimentation are Feed Forward Neural Networks, Convolutional Neural Network, K-Nearest Neighbor Model, Support Vector Machine and a combination of CNN and SVM models (Hybrid model). Figure 1 shows the proposed model.

a. Generation of dataset for classifier models

The data set of rectangular segments is shown in Figure 2. It consists of correctly segmented and partially segmented characters from epigraphical images. The procedure to generate the data set is as follows. Firstly, A kernel is slid across the image and the contents are stored on the disk periodically. A regular sized rectangular kernel is chosen such that it fits a single character. This kernel is slid across the input image to capture a small region of the image at each iteration. The process continues until kernel reaches end of image. The generated inputs are grouped into two classes - Valid characters and Invalid characters (non-text characters), which constitutes the dataset for all the models.

b. Preprocessing

The cropped segments may contain noise and will be of different size since the text size isn't consistent among the documents. During the preprocessing stage, a median filter is applied to improve quality and the images are resized to 32*48. At a first glance each character is identified by its border and classified into one of text and nontext classes. [23].

c. Neural network

The neural network is a CNN. A feed forward neural network is attached to the convolution layers. This network is trained to classify the input dataset. The network uses RELU activation function for all of its hidden neurons and sparse activation makes the network efficient and easy for computing. Multiple hidden layers are selected based on trial and error and these layers aid in learning linear and nonlinear relationships between inputs and outputs [24].

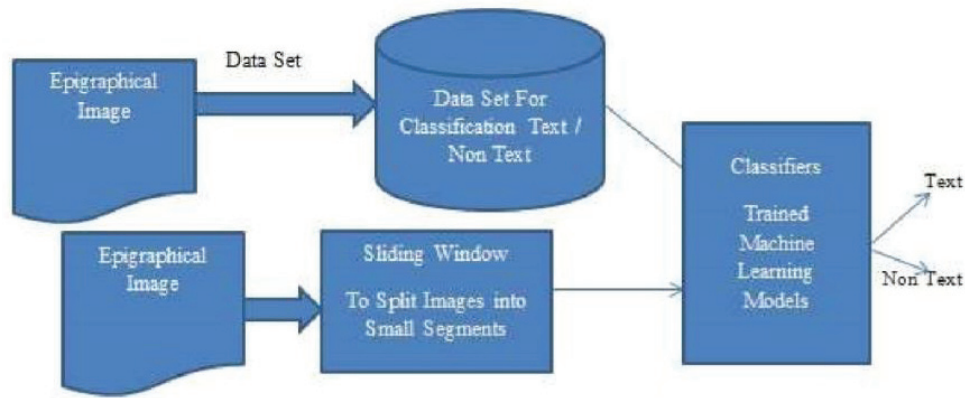


Figure 1: Logic flow of the proposed model of the study. A dataset of segmented characters is built to train the CNN model. The test image is broken down iteratively using sliding windows. Each segment is sent to the model.

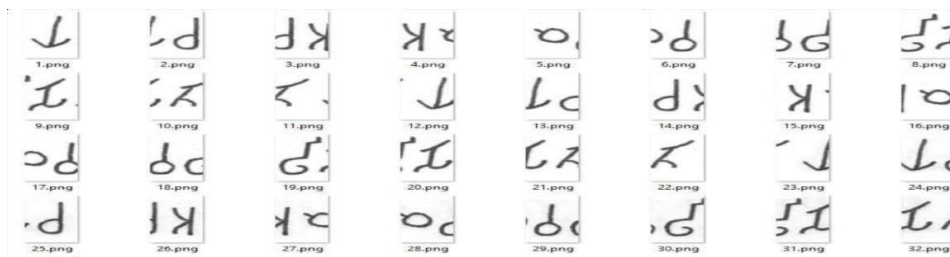


Figure 2: Small cropped segments which forms the train dataset.

The architecture adopted for our study (after the initial trial and error classification) includes an input layer which takes a segment of size 32*48. The image is flattened and passed to three hidden layers with activation function as RELU, and propagates its way to the output layer with Sigmoid activation function. The loss is calculated using binary cross entropy and the optimizer used is Adam. To evaluate the model’s efficiency, accuracy and F1 Score are used as metric.

The data set is split into training, testing and validation sets. 10 percent of the data is set aside for validation. 80 percent of the segments is used for training, while the remaining 20 percent of it is used for testing. Epochs are set such that, the network stops once the loss converges to zero.

This method works in all cases where the letters are disjoint. Languages which heavily make use of connected letters cannot be segmented using this method. Output of the model and accuracy are discussed in results section.

d. Convolutional neural network

Convolutional Neural Network was designed to have a series of 2D convolutional layers, each followed by a max pooling layer. Rearranging the layers and changing number of neurons were two things that made it pivotal in reducing overfitting and improving the validation accuracy as opposed to using L1 and L2 regularizes, which improved the training accuracy. Stochastic gradient descent was used to train the model rather than steepest ascent gradient descent [25] for obvious reasons Figure 3.

The network has 4 convolutional layers, RELU activation function is applied to each layer. The optimizer used for gradient calculation is Rmsprop and Loss is calculated using Binary Cross Entropy. Given below are the equations that depict the calculation of new weights and bias using Rmsprop optimizer. β value is assigned to 0.9 and ϵ is used to prevent the gradients from blowing up as v_{dw} can be zero.

$$v_{dw} = \beta \cdot v_{dw} + (1 - \beta) \cdot dw^2 \tag{1}$$

$$v_{db} = \beta \cdot v_{db} + (1 - \beta) \cdot db^2 \tag{2}$$

$$W = W - \alpha \cdot \frac{dw}{\sqrt{v_{dw} + \epsilon}} \tag{3}$$

$$b = b - \alpha \cdot \frac{db}{\sqrt{v_{db} + \epsilon}} \tag{4}$$

e. Support vector machine

Histogram of oriented gradients features is constructed and is used to classify the segments into character or non-character from the dataset. Principal component analysis is used for dimensionality reduction and was adopted to normalize features before the SVM algorithm is invoked. SVM, using nonlinear kernel, captures complex relationship between the correctly segmented characters and the incorrect ones. According to the observations made, the time taken is higher and is also computationally intensive [26,27].

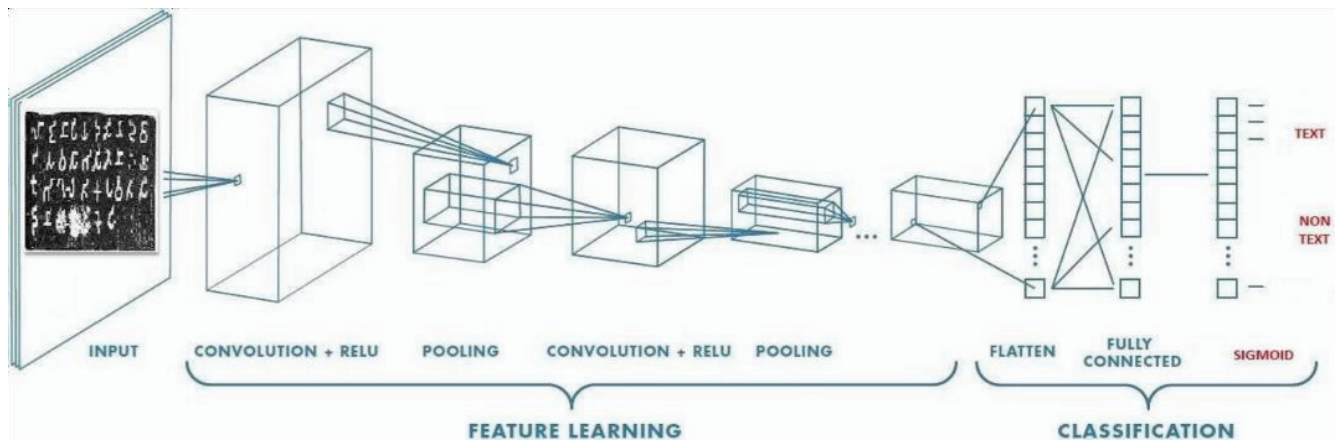


Figure 3: Convolutional neural networks.

HOG feature descriptors are best known for object detection in computer vision problems. The magnitude of gradients along the corners and edges define the object shape. These features are extracted from the input of size $32 \times 48 \times 3$. As a first step, vertical and horizontal gradients are calculated using the kernel mentioned below.

Sharp changes in intensity are triggered by magnitude of the gradients. vertical lines are used to depict intensity changes along x axis and horizontal lines along y axis. There are no data points in smooth regions. This typically shows the outline of characters in our image. By analyzing the inherent structure of HOG features and reducing of features using principal component analysis [28], they are subjected to Support vector machines.

Using these data points, SVM generates the best decision line called hyper plane, classifying text and non-text characters. Experimented linear kernel gave considerable outcome with good accuracy.

f. Combination of convolutional neural network and Support vector machine

Individually, the CNN and the SVM models performed poorly. To improve the accuracy further, a model that was composed of CNN and SVM cascaded together was proposed and implemented. On the raw dataset, Convolutional neural network was trained and the results were classified as true positives, true negatives, false positives and false negatives. True positives were written to disk while false positives and negatives were sent to Support Vector Machine model, which constructed a hyperplane to classify text and non-text. The accuracy of the model drastically increased compared to any other model discussed. The results and discussion would showcase more in the next section.

Implementation

The models were tested on Google Colab with 12GB RAM and 80 GPU NVIDIA which accelerated the execution. The code was written in Python 3.5 using Tensorflow, an openended software library used for machine learning algorithm design, as the framework for building neural networks.

To build the dataset, images were fed to a sliding window algorithm, which crops the image into equal-sized segments. It is a brute force technique to generate all possible segments having both text and nontext. The window size is based on the size of the text. The number of generated crops depends on the image size. On the generated segments, a simple border check algorithm is applied and if a border exists it is considered to be text, as text would appear in the middle of the segment. The images are resized to 32×48 in order to create a uniform dataset. Distortion of text is not an issue to the neural network since it is only concerned with whether a character exists within the segment. Nearly 5400 Samples are recorded in the dataset file having details of text and nontext labels with 1 and 0 respectively. Noisy segments may be classified as correct characters if they lie within the segment. In such cases, SSIM is used to compare the output with the set of all acceptable characters of the language to discard unwanted and yet, properly segmented characters. An example of such noise is the presence of characters from a different language Figure 4.

Results and discussion

The performance was evaluated with the help of a confusion matrix. Accuracy is a measure of correctly predicted instances over all the instances made and F1 score is the weighted average of precision and recall. It is a measure of how correctly the model classifies positive instances.

Accuracy is the ratio of correctly predicted observations to the total observations and F1 score is weighted average of Precision and Recall. The equations 5-8 describe the metric as follows:

$$\text{Accuracy} = \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad (5)$$

$$\text{Precision} = \frac{(TP)}{(TP+FP)} \quad (6)$$

$$\text{Recall} = \frac{(TP)}{(TP+FN)} \quad (7)$$



$$F1 \text{ Score} = \frac{2 * (\text{Recall} * \text{Precision})}{(\text{Recall} + \text{Precision})} \tag{8}$$

Table 2 depicts result of all the models and shows that a combination of Convolutional neural network and Support vector machine outperformed in classifying the characters. Upon completion of prediction correctly segmented characters are written on to the disk which can be further used for recognition Figure 5.

From the above table, it can be concluded that the hybrid model outperformed and learnt to classify text and non-text. The models are tested on inputs and outputs are depicted in the below Figures 6-8.

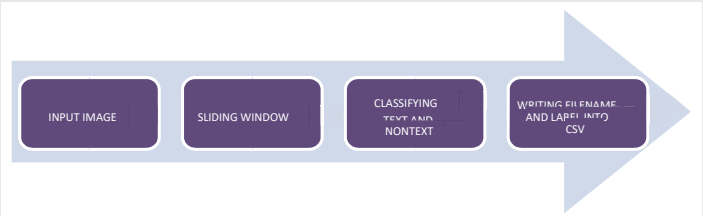
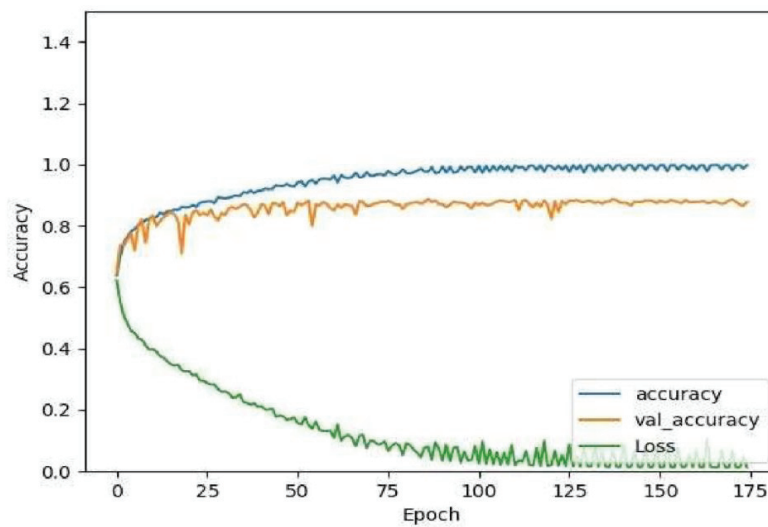


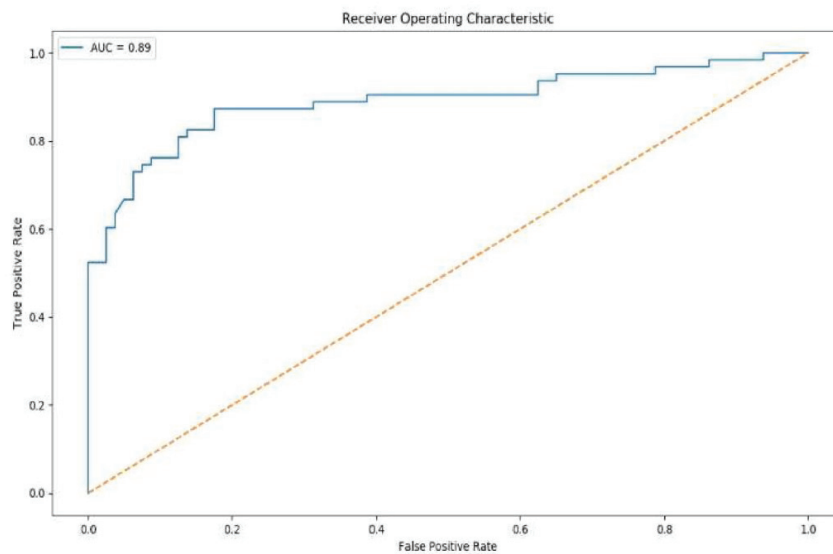
Figure 4: The steps in generation of dataset.

Table 2: Description of training accuracy of Machine Learning Models in predicting character and Non-character regions in the image segments.

Model	Test Accuracy	F1 Score
CNN	87.3%	89.5%
SVM	63.61%	67.05%
Combination of CNN & SVM	98.35%	98.34%



CNN: Plot of Accuracy and Loss against epochs during training of the model. It can be implied from this graph that the model is not overfitting since the validation accuracy remains high while the loss reduces.



SVM: True positive versus False Positive ROC curve. The dotted line corresponds to 0.5, or inability to distinguish between the two classes. The curve (The blue jagged line) is significantly higher than the dotted line, indicating that the model can very clearly discern between the two classes.

Figure 5: The training accuracy of the CNN network and the ROC curve for the SVM model.



Figure 6: Depicting clear input image and corresponding output segmented text images.



Figure 7: Depicting degraded input image and corresponding output segmented text images.



Figure 8: Depicting a severely degraded historical document and its Output having only one script. .

It was found that the Hybrid model outperformed all other models with the Accuracy of 83.758% and F1 Score of 92.06% for the test input images.

Conclusion and future work

From experimentations, Machine learning models can be used to classify segmented images into text and non-text which further can be used for recognition and prediction of time periods. Feed forward neural network, while has a good accuracy, failed to obtain high F1 scores and was remedied by using the hybrid model, which outperformed all the other models. It was both consistent and had fast convergence. With the expense of computational power, the classifiers can converge at greater speeds.

Input images used are preprocessed Estampage images and printed epigraphical scripts. The sliding window applied on these images produced training dataset and the machine learning models were tested on 8000+ segments. Acceptable results were established for epigraphical samples having equal-spaced characters and for white space delimited characters. On the Estampage images (images having noisy, overlapped characters) accuracy of the models decreased because of varying sized characters. The results of segmentation can be further used for prediction of era and its dynasty.

Trained CNN models can be set up with the SVM classifier in a pipeline to automatically perform the process of segmentation with little effort from the end users. The two models when cascaded, create a verification process, where the SVM verifies the results generated by the neural network.

The novelty of the model presented in the article is summarized as follows

1. The use of white space surrounding the black pattern as a separator is independent of alignment of text, size of the line or the dimensions of the character. Skewed text with uneven and unconstrained string of characters are easily segmented. The model performs well even in cases with minor overlap.
2. The model performs well when the document is slightly degraded. It however fails when the noise has the same texture as text.
3. The use of a verification model alongside the primary classification model reduces the need for human verification once the segmented characters are presented. There were very few wrongly classified characters, as indicated by the results table.

The shortcomings of this model are

1. The primary assumption is that characters are segmented based on white boundaries. Any scribble on the document, separated by whitespace is classified as character. The models do not know whether the language encompasses the patterns extracted from the document.
2. The model fails to extract characters from languages such as Hindi, where the characters are required to be joined to form words.

Future work lies in improving the accuracy of segmentation and classification of character/noncharacter by automatically training machine learning models to crop and classify which can build dataset for prediction of era and recognition. These can be implemented by other deep learning models like RCNN, Fast RCNN and Mask RCNN.

Acknowledgement

We thank Archaeological Survey of India, Mysore for providing the dataset for the project. All the work was done at PES University.

References

1. Manigandan T, Vidhya V, Dhanalakshmi V, Nirmala B (2017) Tamil character recognition from ancient epigraphical inscription using OCR and NLP. 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDAS), Chennai, 1008-1011. [Link: https://bit.ly/3jBsXkT](https://bit.ly/3jBsXkT)



2. Likforman-Sulem L, Zahour A, Taconet B (2007) Text line segmentation of historical documents: a survey. *International Journal of Document Analysis and Recognition (IJ DAR)* 9: 123-138. [Link: https://bit.ly/3ik02SQ](https://bit.ly/3ik02SQ)
3. Padmaprabha P, Ramappa MH (2018) A Systematic Approach in Transforming Inscriptions into Modern Text–Review. *International Journal of Signal Processing, Image Processing and Pattern Recognition* 11: 37-44. [Link: https://bit.ly/3CdeC6m](https://bit.ly/3CdeC6m)
4. Louloudis G, Gatos B, Pratikakis I, Halatsis K (2006) A block-based Hough transform mapping for text line detection in handwritten documents. In *Tenth International Workshop on Frontiers in Handwriting Recognition*. Suvisoft. [Link: https://bit.ly/3jBoHlp](https://bit.ly/3jBoHlp)
5. Kavitha AS, Shivakumara P, Kumar GH, Lu T (2016) Text segmentation in degraded historical document images. *Egyptian Informatics Journal* 17: 189-197. [Link: https://bit.ly/3lwhlwz](https://bit.ly/3lwhlwz)
6. Likforman-Sulem L, Zahour A, Taconet B (2006) Text Line Segmentation of Historical Documents: a Survey. *International Journal on Document Analysis and Recognition*. [Link: https://bit.ly/3A32LG4](https://bit.ly/3A32LG4)
7. Sapirstein P (2019) Segmentation, Reconstruction, and Visualization of Ancient Inscriptions in 2.5D. *Journal on Computing and Cultural Heritage (JOCCH)* 12: 1-30. [Link: https://bit.ly/3ChPOKO](https://bit.ly/3ChPOKO)
8. Murthy SK, Kumar HG, Shivakumar P, Ranganath PR (2004) nearest neighboring clustering based approach for line and character segmentation in epigraphical scripts. [Link: https://bit.ly/2Vo3iE3](https://bit.ly/2Vo3iE3)
9. Bhat SS, Balachandra Achar HV (2016) Character recognition and Period prediction of ancient Kannada Epigraphical scripts. *International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering* 3: 114-120. [Link: https://bit.ly/3CeSKrd](https://bit.ly/3CeSKrd)
10. Hu R, Odobez JM, Gatica-Perez D (2017) Extracting maya glyphs from degraded ancient documents via image segmentation. *ACM Journal of Computational. Cultural. Heritage* 10: 71-93. [Link: https://bit.ly/3A8ZrcF](https://bit.ly/3A8ZrcF)
11. Mohana HS, Navya K, Rajithkumar BK, Nagesh C (2014) Interactive segmentation for character extraction in stone inscriptions. *Second International Conference on Current Trends In Engineering and Technology - ICCTET*. [Link: https://bit.ly/3AmmSzx](https://bit.ly/3AmmSzx)
12. Alberti M, Seuret M, Pondenkandath V, Ingold R, Liwicki M (2017) Historical Document Image Segmentation with LDA-Initialized Deep Neural Networks. In *Proceedings of 4th International Workshop on Historical Document Imaging and Processing, Kyoto, Japan*. [Link: https://bit.ly/2TTJEnr](https://bit.ly/2TTJEnr)
13. Chen K, Liu C, Seuret M, Liwicki M, Hennebert J, et al. (2016) Page Segmentation for Historical Document Images Based on Superpixel Classification with Unsupervised Feature Learning. *2016 12th IAPR Workshop on Document Analysis Systems (DAS), Santorini* 299-304. [Link: https://bit.ly/2Vx1Gaz](https://bit.ly/2Vx1Gaz)
14. Badjatiya P, Kurisinkel LJ, Gupta M, Varma V (2018) Attention-based Neural Text Segmentation. *European Conference on Information Retrieval 2018, (ECIR-2018),Grenoble, France*.
15. Jo J, Koo H, Soh JW, Cho N (2019) Handwritten Text Segmentation via End-to-End Learning of Convolutional Neural Network. [Link: https://bit.ly/37fU2nT](https://bit.ly/37fU2nT)
16. Al-Rawi M, Bazazian D, Valveny E (2019) Can Generative Adversarial Networks Teach Themselves Text Segmentation?. [Link: https://bit.ly/3imtlUH](https://bit.ly/3imtlUH)
17. Reza MM, Bukhari SS, Jenckel M, Dengel A (2019) Table Localization and Segmentation using GAN and CNN. *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW), Sydney, Australia* 152-157. [Link: https://bit.ly/3rQhe5D](https://bit.ly/3rQhe5D)
18. Li X, Zhang X, Yang B, Xia S (2017) Character segmentation in text line via convolutional neural network. *2017 4th International Conference on Systems and Informatics (ICSAI), Hangzhou* 1175-1180. [Link: https://bit.ly/3xilxX7](https://bit.ly/3xilxX7)
19. Zirari F, Ennaji A, Nicolas S, Mammas D (2013) A Document Image Segmentation System Using Analysis of Connected Components,” *2013 12th International Conference on Document Analysis and Recognition, Washington, DC* 753-757. [Link: https://bit.ly/3A59U8Y](https://bit.ly/3A59U8Y)
20. Manigandan T, Vidhya V, Dhanalakshmi V, Nirmala B (2017) Tamil character recognition from ancient epigraphical inscription using OCR and NLP. *2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), Chennai*, 1008-1011. [Link: https://bit.ly/3jffnly](https://bit.ly/3jffnly)
21. Sowmya A, Kumar H (2015) Enhancemnet and segmentation of historical records. *Fifth intenation conference on CS and IT*. [Link: https://bit.ly/37iELCH](https://bit.ly/37iELCH)
22. Abtahi F, Zhu Z, Burry AM (2015) A deep reinforcement learning approach to character segmentation of license plate images. *2015 14th IAPR International Conference on Machine Vision Applications (MVA), Tokyo* 539-542. [Link: https://bit.ly/3ftioyY](https://bit.ly/3ftioyY)
23. Bailey DG, Klaiber MG (2019) Zig-Zag Based Single-Pass Connected Components Analysis. *J Imaging* 5: 1-26. [Link: https://bit.ly/3CfUeSn](https://bit.ly/3CfUeSn)
24. Shrruru K (2020) An Introduction to Artificial Neural Network. *International Journal Of Advance Research And Innovative Ideas In Education* 1: 27-30.
25. Khan S, Rahmani H, Ali Shah SA, Bennamoun M, Medioni G, et al. (2018) A Guide to Convolutional Neural Networks for Computer Vision,” in *A Guide to Convolutional Neural Networks for Computer Vision*. Morgan & Claypool 1-18. [Link: https://bit.ly/3lvdViP](https://bit.ly/3lvdViP)
26. Lipo W (2005) *Support Vector Machines: Theory and Applications*. Springer Science & Business Media 431. [Link: https://bit.ly/3rZI38v](https://bit.ly/3rZI38v)
27. Ni KS, Nguyen TQ (2009) An Adaptable k-Nearest Neighbors Algorithm for MMSE Image Interpolation. *IEEE Transactions on Image Processing* 18: 1976-1987. [Link: https://bit.ly/3xmPh6o](https://bit.ly/3xmPh6o)
28. Lin T (2018) PCA/SVM-Based Method for Pattern Detection in a Multisensor System. *Mathematical Problems Engineering, Hindawi*. 1-11. [Link: https://bit.ly/3yt8bKg](https://bit.ly/3yt8bKg)

Discover a bigger Impact and Visibility of your article publication with Peertechz Publications

Highlights

- ❖ Signatory publisher of ORCID
- ❖ Signatory Publisher of DORA (San Francisco Declaration on Research Assessment)
- ❖ Articles archived in worlds' renowned service providers such as Portico, CNKI, AGRIS, TDNet, Base (Bielefeld University Library), CrossRef, Scilit, J-Gate etc.
- ❖ Journals indexed in ICMJE, SHERPA/ROMEO, Google Scholar etc.
- ❖ OAI-PMH (Open Archives Initiative Protocol for Metadata Harvesting)
- ❖ Dedicated Editorial Board for every journal
- ❖ Accurate and rapid peer-review process
- ❖ Increased citations of published articles through promotions
- ❖ Reduced timeline for article publication

Submit your articles and experience a new surge in publication services (<https://www.peertechz.com/submition>).

Peertechz journals wishes everlasting success in your every endeavours.

Copyright: © 2021 Preethi P, et al. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Citation: Preethi P, Mamatha HR, Viswanath H (2021) Study of using hybrid deep neural networks in character extraction from images containing text. *Trends Comput Sci Inf Technol* 6(2): 045-052. DOI: <https://dx.doi.org/10.17352/tcsit.000039>