Text Extraction and a Deep CNN Based Model for Character Classification in Kannada Documents

Sachin Bhat, Seshikala G

Abstract*: Pattern analysis in documents is one of the most interesting issues in the current research because of its wide are of of applications. It has leveraged its potential in reducing the manual work of converting the documents containing handwritten characters to machine-readable texts. The Deep Convolutional-Neural-Networks (DCNN) are successfully implemented for the recognition of characters in various languages. But due to high noise, degradation over a long time period, low contrast and intensity to separate the foreground text plays a spoiler in the extraction of characters from the document images. This paper proposes covers both the aspects including preprocessing of Kannada documents and a DCNN based architecture for the classification of Kannada language characters. Kannada is one of the 22 official languages in India spoken by more than 60 million people across the globe. This model is mainly developed to assist the character recognition of Kannada documents. A total of 84000 characters including both vowels and consonants have been included in the dataset. This architecture is showing a satisfactory test accuracy of 98.87% for the classification of 42 handwritten characters.

***Index Terms*: CNN, Document Analysis, Image Enhancement, Optical Character Recognition**

I. INTRODUCTION

Writing information on papers, palm leaves, copper plates, stones existed from several centuries. This method was followed by hundreds of years not only in India, but all over the world. These types of writings are generally called as manuscripts. This is the main tool used in history to study the life of ancient time. Manuscripts became the important tool in transmitting the information and traditions from one generation to another. Document analysis is a technique to upgrade the calibre of a document to improve the human perception and to help later automated processing of images. This is also a major preprocessing step in the Optical Character Recognition(OCR). OCR is a process of converting the documents containing printed/handwritten characters into machine-readable format. In recent times, it has shown its potential of cutting down the manual work of digitizing the images of printed or handwritten text. Both preprocessing and OCR have become very interesting research fields helping to improve the calibre of documents and thereby recognising and classifying the text from images. Various conventional algorithms are used for OCR like template matching, hidden markov model etc. With advancing technology and processing power, machine learning algorithms are taking over the traditional methods with improved accuracy and high speed. But, OCR of handwritten text in a document is still a complex problem for researchers because to its poor quality, indifferentiable foreground and background and variety of handwriting styles. It is particularly true for Indian languages due to a vast character set and complex writing style. OCR with high accuracy is reported in English and other western languages which have a less number of characters and minimum structural complexity. But character recognition of Indian scripts is comparatively acute coz of its compound structure and similar nature of characters.Remaining part of the paper has been arranged in the below format. Part II lists some of the notable works accomplished in the domain of document image binarization and character recognition. Part III depicts the methodology developed in detail. Experimentation with result evaluation is shown in part IV. Part V will be the conclusion

II. REVIEW OF LITERATURE

In this part, we briefly depict some of the text extraction and character recognition methods used by earlier researchers. Generally, the techniques used for BZ can be either local or global. The global binarization techniques allot a single threshold for the entire image whereas threshold for individual or group of pixels in the document image will be identified in local binarization. Histogram shape based global binarization methods [1][2] tries to estimate a global threshold to minimize intra-class variance. It requires a bimodal histogram pattern and therefore, cannot handle the document images with high variation in background. Though local binarization methods comparatively yield a better result, it is still an unsolved problem in case of ancient manuscripts. This is mainly due to different variety of noise, degradation and unclear foreground. Adaptive thresholding methods like Sauvola[3] which is an improvement over Niblack's[4] and Bernsen[5] will either generate a certain quantity of noise or fail to identify the text with a low contrast[6]. All these algorithms use mean, variance or standard deviation and contrast information of local region to calculate different thresholds. They have also failed to handle the images with light texture background.

Revised Manuscript Received on June 07, 2019.
Sachin Bhat, School of ECE, Reva University, Bengaluru/ SMVITM, Udupi, India
G Seshikala, School of ECE, Reva University, Bangalore, India

Feng[7], sets a standard in contrast adaptive thresholding by dynamically calculating the threshold depending on the gray-scale average and variance of current pixel in the neighborhood. Paper [8] demonstrates a real time adaptive thresholding using integral image of the input. This is robust to illumination changes in the image which makes it worthy for the processing of video applications. Lu in [9] used an adaptive image contrast combining the local image contrast and gradient which is tolerant to background and foreground variation caused by different types of degradations. This has been tested on 3 public databases achieving an overall accuracy of 90%. Fast ICA and NGFICA have been used for enhancement of historical documents by maximizing text layer information[10].High intensity variations and low contrast of both text and background caused by the degradation makes it difficult to design a uniform binarization model that can separate foreground from the background. Two examples can be seen from Figure 1(a) and (b). Different types of degradations make algorithms like Otsu's, Niblack and Savoula to generate bad results as shown in Figures 1(c) to (h).

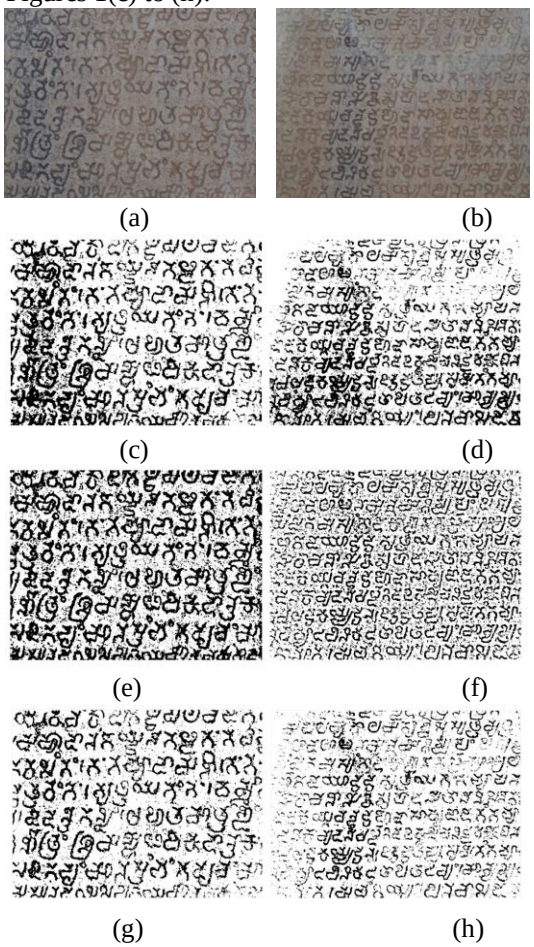


Fig 1. (a), (b) Input image, (c), (d) Ostu's binarization results, (e), (f) Niblak's binarization results, and (g), (h) Savoula's binarization results.

Also, there are several models reported in the area of character classification for online datasets. Different CNN architectures are developed for ICDAR2003, Chars74K and IAM database. He et al[11] designed a deep RNN for text classification where CNN was used to construct an ordered sequence from the word and LSTM to recognize these sequences. Some notable works are also available for text classification of Indian scripts like Sanskrit and Bangla. 3 databases of Bangla script are available online namely Banglalekha, CMARTERdb and SUST-BHND with 90000, 15000 and 100000 characters respectively. BornoNet[12]: a lightweight CNN based architecture was developed to classify these characters. Different CNN models like Resnet, VGG and Densenet are used to check the performance analysis of Bangla literals. Magnanimous amount of work is reported in Bengali compared to any other Indian language. There are few works reported in other languages as well like Devanagari and Malayalam. Umapada[13] has proposed quadratic classifier based method for off-line handwritten character recognition of Hindi script.

III. PROPOSED METHODOLOGY

Proposed methodology is split into two parts. First one deals with the preprocessing of Kannada documents and second one with the recognition of Kannada characters extracted from these documents. Flow of which is showed in Fig 2

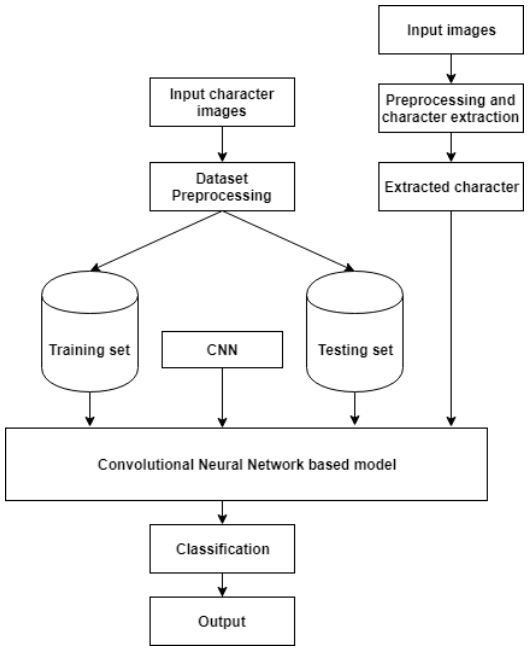


Fig 2. Proposed methodology for preprocessing and character recognition

A. Preprocessing

Phase information in frequency domain always outweighs the amplitude information of an image in spatial domain. Points with maximum Fourier components are considered as pixels of interest. Phase coherence (PC)[14][15][16] 2D is the ratio of weighted alignment of Fourier components to the sum of Fourier components which is given by

$$PC_2(x) = \frac{\sum W_r(x)\left[A_{po}(x)\cos\left(\phi_{po}(x) - \phi'_{po}(x)\right)\right]}{\sum A_{po}(x)} \quad (1)$$

Where, $W(x)$ is PC weighting mean function, $A_n(x)$ is local amplitude of nth component with scale(p), orientation(p) and $\cos(\phi_{po}(x)-\phi'_{po}(x))$ is phase deviation function. Non-orthogonal logarithmic wavelets are used to preserve the phase information of image. Local phase and amplitude information of every point in the image are extracted by this. Noise threshold of each level is estimated so that magnitudes of filter response vector can be shrunk while phase information will be untouched. As this PC is sensitive to noise, Gaussian distribution is applied. Noise shrinking threshold is calculated and is removed using below equation with mean μ_G and variance σ_G.

$$T_G = \mu_G + k.\sigma_G \quad (2)$$

Gaussian mixture(GM) model is employed to estimate and remove the background pixels. Here, each pixel is compared with each Gaussian and is classified according to its corresponding Gaussian. Probability of observing the value of any pixel P(x) for k number of Gaussian components is considered by using

$$P(x)=\sum_{i=0}^{k} W(k)\eta(X(t)), M_k, V_k), \quad (3)$$

Here, $W(k)$ - estimated weights of GM, M_k - estimated mean vectors of GM, V_k - estimated covariance matrices of GM. And η is Gaussian probability density function

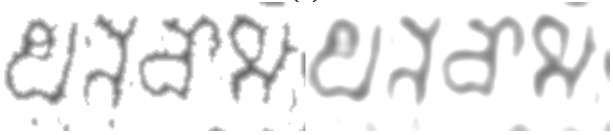
$$\eta(X(t)), M_k, V_k)=\frac{1}{\sqrt{2\pi V_k^2}} e^{\frac{-1}{2}(X-M_k)V_k^{-1}} \quad (4)$$

At each pixel location, corresponding GM parameters are determined standard expectation-maximization (EM). It is an iteration which starts with an initial estimation of above said parameters and updates them till convergence is detected. Expectation computes the weights for all pixels and maximization uses them to calculate a new parameter values. Once all the parameters are initialized, foreground pixels are estimated and parameters are updated. K value is set to 3 according to [17] and *K* Gaussian followed the value of the ratio. Background pixels correspond to high weight with weak variance. If Gaussian distribution (B) exceeds a threshold (T) are treated as background distributions and are eliminated

$$B = \arg\min\left(\sum_{i=1}^{b} W(k) > T\right) \quad (5)$$

Binarized form of input image fig.3(a) using PC is given in fig 3(b). As this is a weak denoise technique, misclassified objects below threshold T are removed as 3(c). EM algorithm eliminates background pixels as in 3(d) and (e). Anisotropic filtering is used to smooth the binarized image in 3(f).

(a)



(b) (c)

(d) (e)

(f)

Fig 3. Steps involved in binarization (a)input image (b)-(c): binarization using phase features (d)-(e): Background pixel removal with EM algorithm (f): Smoothing using anisotropic filtering.

B. Database

Database is a collection of the information that is organized so that it can be easily accessed, compared and updated. We have created 84000 samples of characters belonging to 42 classes which are subdivided into train, test and validation set in a ratio of 60:20:20. 40 students are asked to write 50 samples of each class on an A4 sheet. These sheets are scanned and saved in JPG format. Images are binarized and noise removals are done in the earlier defined method. Binarized characters are segmented and resized to 32X32 pixels and are extracted into a folder. These characters are manually labeled and assigned to different classes. Some of the random characters taken from 6 different classes of the dataset are shown in Fig.4.

Fig 4. Randomly chosen character images

C. Character Classification

Inspired by [18], our model too has various sequential small sized convolution layers and fully connected(FC) layers. But dimension of the images are reduced to 32X32 pixels to make it able to get the trade-off between size and training time. Five convolutional blocks are arranged one after another consisting of 2, 2, 3, 3, 3 convolutional layers and one pooling layer respectively. Extracted features after convolution are fetched to customized FC layers. Pretrained weights of VGG16 on ImageNet database are utilized to train our dataset. Top FC layers have been removed and four customized layers are added. This includes two FC dense layers with a flatten and a dropout layer. Dropout ratio is set to 0.5 which means 50% of the nodes are turned of in each operation. Finally a softmax classifier of 42 classes is included in relevance to our problem. Architecture of this model and the hyperparameters chosen to implement this model are shown in Fig 5 and Table I respectively.

Table I: Tuning of hyperparameters

Deep learning model framework used	Keras
Input image size	32X32X3
Pooling layer	Maxpooling
Padding	Zero padding of size (k-1)/2 where k is size of filter
Receptive field	3X3
Convolution stride	1
Activation function	ReLu
Classifier	Softmax with 46 classes
Dropout ratio	50%
Optimizer	Adadelta
Learning rate	0.01
Momentum	0.1

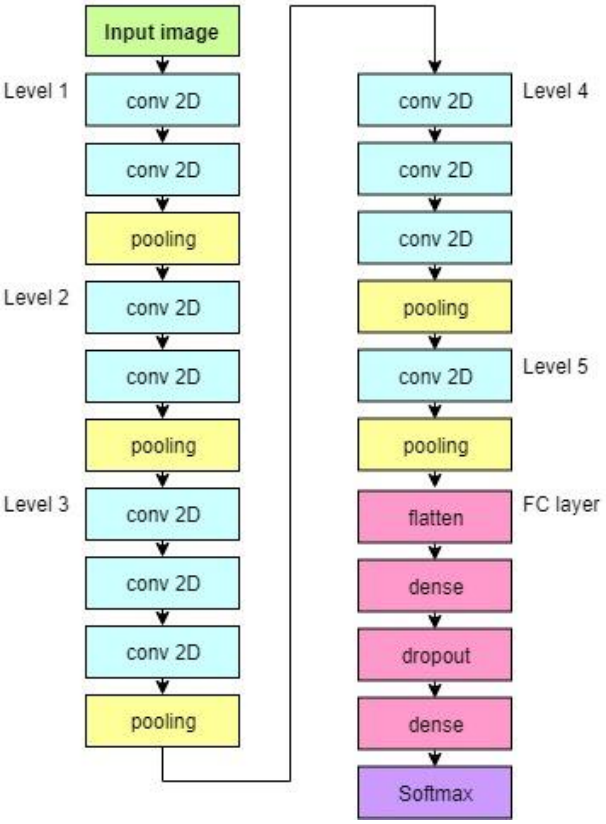


Fig 5: Architecture of CNN based classification model

Initially ReLu activation function is used on training and validation set. Activation function is fixed and accuracy of different optimizers are observed. It is noted that Adadelta optimizer is giving the highest accuracy for our data. This experiment is shown in Table II and Fig.6

Table II: Accuracy in different types of Optimizers.

Activation function	Optimizers	Training accuracy	Validation accuracy
RELU	Adadelta	98.58%	98.67%
	Adagard	80.79%	96.72%
	Adam	92.64%	97.8%
	Adamax	76.9%	96.81%
	Nadam	96.64%	98.25%
	RMSprop	98.55%	98.25%
	SGD	89.21%	94.46%

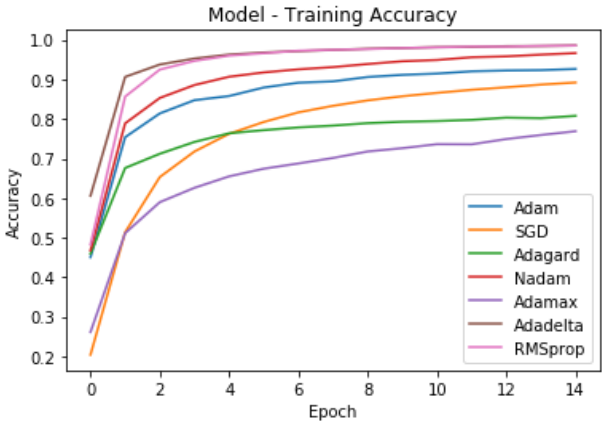


Fig 6: Training Accuracies of different optimizers for ReLu activation

IV. EVALUATION OF RESULTS

This model is trained with training and validation sets for 10 epochs with a batch size of 500. Summary of this model is as given in Table III.

Table III: Architectural summary of the proposed model

Model	Parameters
32X32 image	-
Con-64	1792
Con-64	36928
2X2 Maxpool	0
Con-128	73856
Con-128	147584
2X2 Maxpool	0
Con-256	295168
Con-256	590080
Con-256	590080
2X2 Maxpool	0
Con-512	1180160
Con-512	2359808
Con-512	2359808
2X2 Maxpool	0
Con-512	2359808
Con-512	2359808
Maxpool	0
Flatten-512	0
Dense-1024	525312
Dropout-1024	0
Dense-45	47150
Softmax	45
Total params:	12927342
Trainable params:	12927342
Non-trainable params:	0

For the customized database of 84000 images, after 10 epochs model is predicts 97.87% accuracy for validation set. This is given in Fig 7.

It is neither underfitting nor overfitting as train accuracy follows same path as validation accuracy. Performance of the CNN architecture is evaluated by plotting confusion matrix. Fig 8 shows the misclassified points using the same.

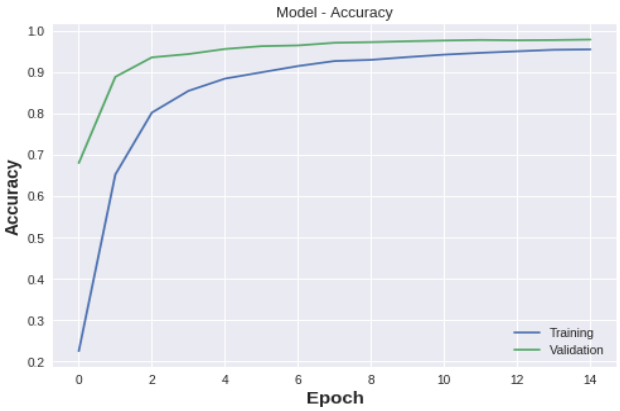


Fig 7: Training and validation accuracy plot

Different state of art machine learning algorithms are applied on the same dataset. The test accuracies are calculated for each model and are compared with the proposed method which are shown in Fig 9. This model is executed in python3 supported by Google Colab GPU. It is compared with some of the traditional machine learning algorithms. The test accuracies and score times are calculated for each model and are compared with the proposed method which are shown in table 2. Classifiers considered for comparison are KNN(k=5), Ridge classifier, Naive Bayes, Logistic regression, Extra tree classifier, SVM, Decision-tree with depth 42 and Random-forest with depth 42.

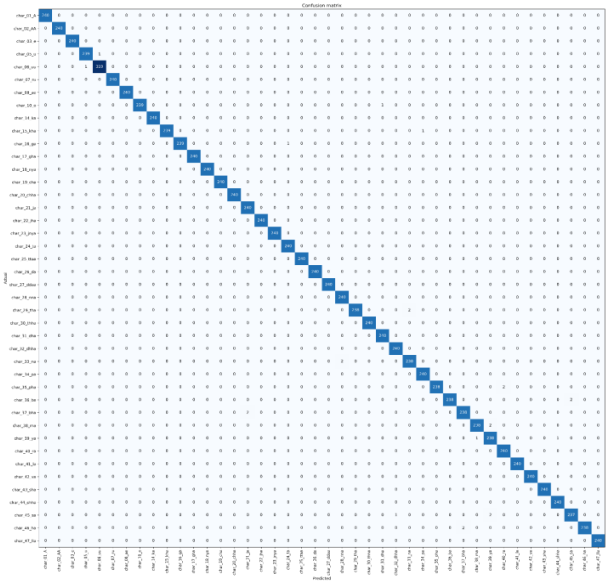


Fig 8: Confusion matrix to show the misclassified points

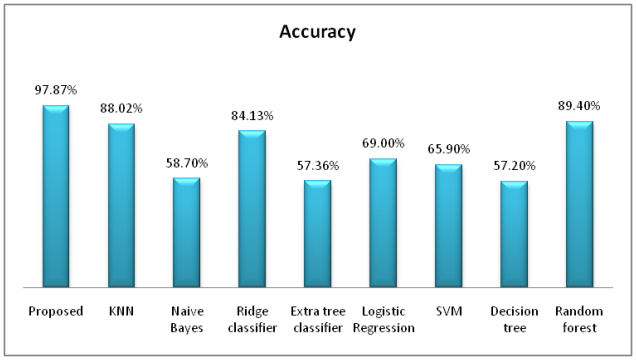


Fig 9: Comparison of accuracies obtained by various machine learning algorithms.

V. CONCLUSION

Recognition of handwritten text is a challenging task due to its inter and intra class variation of character patterns. Lack of benchmark datasets is one of the major problems encountered while addressing the issue of document analysis for all the Indian language scripts. This paper deals with the creation of handwritten Kannada character database and its classification using CNN approach. 42 classes of characters including vowels and consonants have been considered for the dataset creation. 84000 samples have been created based on the orthographic shape of the characters. VGGNet based CNN model is designed which achieved a recognition accuracy of 97.87%. This is evaluated against many machine learning algorithms and their accuracies with execution times are noted down. This is an inception model for the recognition of Kannada script. Future work will focus on the expansion of dataset by including the compound characters and improving the system accuracy by introducing new deep learning models.

REFERENCES

1. N. Chaki, S. H. Shaikh, and K. Saeed, “Exploring image binarization techniques,” *Studies in Computational Intelligence*, Springer 2014
2. N. Otsu, “ A threshold selection method from gray-level histograms,” *IEEE transactions on systems, man, and cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.
3. J. Sauvola and M. Pietikainen, “Adaptive document image binarization,” Pattern recognition, vol. 33, no. 2, pp. 225–236, 2000.
4. W. Niblack, “An introduction to digital image processing”, vol. 34, Englewood Cliffs: Prentice-Hall, 1986.
5. J. Bernsen, “Dynamic thresholding of grey-level images,” *In Proc. 8th Int. Conf. on Pattern Recognition*, 1986, pp. 1251–1255
6. B. Gatos, K. Ntirogiannis, and I. Pratikakis, “ICDAR 2009 document image binarization contest (DIBCO 2009),” *International Conference on Document Analysis and Recognition*, pp. 1375–1382, July 2009.
7. Feng, Meng-Ling, and Yap-Peng Tan, “Contrast adaptive binarization of low quality document images,” *IEICE Electronics Express* 1, no. 16 (2004): 501-506.
8. Bradley, Derek, and Gerhard Roth. "Adaptive thresholding using the integral image." *Journal of graphics tools* 12, no. 2 (2007): 13-21.
9. S. Lu, B. Su, and C. L. Tan, “Document image binarization using background estimation and stroke edges,” Int. J. Document Anal. Recognit., vol. 13, no. 4, pp. 303–314, Dec. (2010).

10. Sachin Bhat, Avinash N J, "ICA algorithm for image enhancement and improvement of word and character recognition in epigraphs", International Journal of Current Engineering and Scientific Resrarch", vol.4, no.5, pp. 72-76, 2017
11. P. He, W. Huang, Y. Qiao, C. C. Loy, and X. Tang, "Detecting oriented text in natural images by linking segments," in AAAI Conference on Artificial Intelligence (AAAI), 2016
12. P. C. Tsui, Em algorithm for Gaussian mixture model, Tech. Rep., PAMI Research Group, Department of Electrical and Computer Engineering, University of Waterloo, 2006.
13. Pal, Umapada, et al. "Off-line handwritten character recognition of devnagari script." Ninth International Conference on Document Analysis and Recognition (ICDAR 2007). Vol. 1. IEEE, 2007.
14. P. Kovesi, Phase Preserving Denoising of Images, Signal 4(1) (1999), Pages.
15. Sachin Bhat, Seshikala G, " Restoration of Characters in Degraded Inscriptions using Phase Based Binarization and Geodesic Morphology", International Journal of Recent Technology and Engineering, Volume-7, Issue-6, March 2019
16. Sachin Bhat, and G. Seshikala. "Preprocessing and Binarization of Inscription Images using Phase Based Features." *2018 Second International Conference on Advances in Electronics, Computers and Communications (ICAECC)*. IEEE, 2018.
17. P. C. Tsui, Em algorithm for Gaussian mixture model, Tech. Rep., PAMI Research Group, Department of Electrical and Computer Engineering, University of Waterloo, 2006.
18. Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556(2014)

AUTHORS PROFILE

Sachin Bhat received his M.Tech degree in Digital Electronics and Communication from NMAMIT, Nitte. He is a research scholar in the School of Electronics and Communication Engineering, Reva University, Bengaluru and working as a senior Assistant Professor in SMVITM, Udupi. His areas of interest are Document Image Processing, Deep Learning, Remote Sensing and Object Oriented Programming. He has published over 30 research papers in national and international journals.

Dr. G. Seshikala has 25 years of teaching and research experience. She holds PhD degree in Biomedical Signal Processing from JNTU-A. BE in ECE and and ME in Digital Electronics. She has published over 20 research papers in international journals and conferences. Her areas of specialization are Communication Engineering, Image Processing, Pattern Recognition and Signal Processing.