

Hybrid Deep Resnet With Inception Model For Optical Character Recognition In Gujarati Language

^{*1}Mr. Sanket B. Suthar, ²Dr. Amit R. Thakkar

^{*1}Department of Information Technology,
Chandubhai S. Patel Institute of Technology (CSPIT),
Faculty of technology & Engineering (FTE),
Charotar University of science and Technology (CHARUSAT), Changa, Anand, India.

*Email: sanketsuthar.it@charusat.ac.in

²Department of Information Technology,
Chandubhai S. Patel Institute of Technology (CSPIT),
Faculty of technology & Engineering (FTE),
Charotar University of science and Technology (CHARUSAT), Changa, Anand, India.
Email: amitthakkar.it@charusat.ac.in

Abstract

In the Optical Character Recognition (OCR) system, achieving high recognition performance is important. OCR and visual perception are affected by the inclined characters in each language. Deep learning methods play an important role in the OCR field, which can outperform humans with higher recognition performance. So, in this research, a hybrid deep learning technique is applied to recognize the Gujarati language characters. Initially, Gujarati characters collected from different sources are pre-processed using different techniques. Adaptive Weiner Filter (AWF) is used for noise removal, Binarization, and contrast enhancement is done by Contrast Limited Adaptive Histogram Equalization (CLAHE) method. Finally, a hybrid deep ResNet with Inception model (GoogleNet) is suggested to perform character recognition in the Gujarati language. This hybrid architecture also performs feature extraction tasks, considered a major task in OCR. Python tool is utilized to illustrate the proposed methodology and solve the mathematical model. Scanned documents containing Gujarati characters are engaged to evaluate the robustness of the proposed methodology. Using various performance parameters, the influence of the proposed methodology is examined and its results compared with various deep learning algorithms.

Index terms: Gujarati language, Optical character recognition (OCR), pre-processing, adaptive filtering, Hybrid deep learning algorithms.

I. Introduction

Demand for handheld gadgets is increasing quickly with time in the digitization world. Handheld gadgets also require an efficient and easy tool to input the data. Input using a standard keyboard obliges time and determination, mostly for Indian scripts. Because, they have a huge complex character set that makes the input system difficult to use a normal keyboard [1]. With a simple keyboard, tiny handheld gadgets have a lot of benefits using online words identification system. In India, the Gujarati language belongs to the Devnagari family of languages, and it is instigated in the western state of India

that is Gujarat. Fifty million people of the state speak this language. It is a widely spoken language and has inherited rich cultural and literature properties, but few types of research only concentrate on identifying Gujarati characters from handwritten documents [2, 3]. There are several numbers of handwritten and printed documents available in Gujarati script, and it is essential to preserve those pamphlets in digital format from an efficient distribution and legal and historical perspective [4].

Scanning is considered one of the best methods to transform pamphlets into a digital layout. Still, searching, retrieving and editing the information in a scanned document is considered another difficult task [5]. It is an important task to retrieve the data from the scanned pamphlet. Recognition based and recognition free methods are two important techniques used to retrieve any information from the document [6]. OCR system [7] is a recognition-based method that transforms document images into readable text format. For various Indian regional scripts like Tamil, Malayalam, Telugu, Kannada, Oriya, Bangla, Devanagari and Gurumuki, a small amount of research related to character recognition has been carried out [8]. Researchers are looking forward to inventing new methods for accurately identifying the characters due to the demand for low-cost OCR systems [9, 10]. This paper concentrates on the identification of optical characters in the Gujarati language.

There are only very few works in the literature on the recognition of Gujarati language scripts [11]. In previous years, machine learning methods and conventional pattern recognition approaches have been used in the OCR framework. But, it requires an efficient method to satisfy the need and requirements of users to enhance the marketability of the OCR system using efficiency and economy [12, 13]. Several methods and techniques are used to find the best OCR system to achieve a better recognition rate. Since the last few years, machine learning and deep learning methods have emerged as promising solutions for these OCR problems. Due to the satisfactory results in this area, researchers are making several efforts to extend deep learning architectures [14, 15].

A major objective of this study is to make a structure that automatically identifies the optical characters from a set of scanned documents. Major contributions of this research are defined as follows:

- To perform the different pre-processing methods to make character recognition tasks very simple.
- To design a hybrid deep learning technique that can efficiently recognize a Gujarati character with maximum recognition accuracy.
- To reduce the computational complexity of this entire system by performing a recognition process using a hybrid deep learning algorithm.

The remaining section of this research is ordered as follows: Section 1 introduces the Gujarati language and the advantage of using the OCR framework. Section 2 depicts recent work related to our research methodology and problems in previous research. The proposed methodology is elaborated in section 3. Section 4 illustrates simulation outcomes and considerations of the proposed methodology, and Section 5 provides the conclusion and future enhancement of our research work.

II. Literature review

Generally, the electronic document analysis framework widely uses OCR for character identification. This method has been very useful for extracting text from a scanned document or image and is used in image processing, Natural language processing and pattern recognition. Rakesh Kumar Sethi and Kalyan Kumar Mohanty [16] developed a deep learning technique for optical Odia character classification. There was little progress in handwriting character recognition (HCR) for a small vocabulary for neatly hand-typed characters and new line isolated words. Moreover, a small amount of research work had been done on Odia's character recognition process. In this article, different transfer learning methods like VGG16 and ResNet 50 were utilized to perform the character recognition process, and the performance was compared with existing CNN based techniques.

Ambadas Shinde and Yogesh Dandawate [17] developed a convolutional neural network architecture for handwritten based Marathi text identification. Different authors performed character recognition by different techniques. In computer vision, deep learning algorithms were considered an important technique that correctly predicts the scanned document's characters. In general, handwritten Devanagari characters were considered difficult to identify, which was overcome with the help of deep learning methods. In this article, manually written Marathi words have been precisely identified using an OCR system based on a Convolutional Neural Network (CNN). Also, manually written Devanagari text written in the Marathi language was obtained by developing a character segmentation free technique that replicates the perceived words in printed form.

The Gujarati language normally contains many confusing characters that lead to misclassification. Vishal A. Naik and Apurva A. Desai [18] for online handwritten Gujarati character recognition. So, the classification accuracy of confusing characters was increased with the multi-layer classification method. Initially, training was performed by the polynomial kernel using Support Vector Machine (SVM) in the first classification layer. In the second layer, SVM with linear kernel was utilized to classify confusing letters when the first layer returns a letter with some letters on training data. Finally, both layers perform classification using features obtained from a hybrid feature set that contains dominant point and zoning features based on regularized chain code features.

Optimized Self-Organizing Map (SOM) network was developed by Om Prakash Jena et al. [19] to recognize printed Odia characters and digits. For Odia language, the SOM network was created to build up an OCR framework that effectively performs the character identification task. Some characteristics like shape, different content styles, subordinate conditions of characters and their context make challenges in the Odia character recognition framework. The proposed SOM network was advanced with certain structural features like height, cross-section, width, and end points to obtain 97.55% recognition accuracy.

Dibyasundar Das et al. [20] developed a multi-objective Jaya Convolutional Network (MJCN) for handwritten OCR. This technique tries to learn significant features directly from the images. This MJCN technique contained a convolution layer, an activation layer, a multiplication layer and had a multi-objective Jaya Optimizer (MJO). Over a local neighbourhood connection, the convolution layer explores significant patterns in an image, and the multiplication layer develops the convolutional response to a more compact feature space. MJO algorithm was utilized to optimize the initial weight value in the network. Minimizing intra-class variance and maximising inter-class distance were the main objectives of the MJO algorithm. Standard classifiers were used to recognize the characters from different datasets.

III. Proposed methodology

In the script of Gujarathi, nearly 34 consonants and 12 vowels are available, and such consonants are termed as Vyanjan and Vowels are called Swar. In this proposed work, the own datasets are collected from the vowels and consonants of the Gujarathi language. In which, OCR is necessary for understanding such data to both machinery and humans. Besides, OCR is considered superior, and this is because; process controlling during data production is not required in it. This also suppresses the issue of the identification of optically processed characters. Both the printed and the handwritten characters are verified by OCR. However, the input data quality mainly decides its overall performance. To classify a similar set of characters from the wide data varieties, a classifier is necessary. The ResNet and Inception model is considered a superior method for classifying similar sets. The Gujarati characters are classified in this research work by hybridising such approaches. Figure 1 represents the phases tangled in the proposed technique.

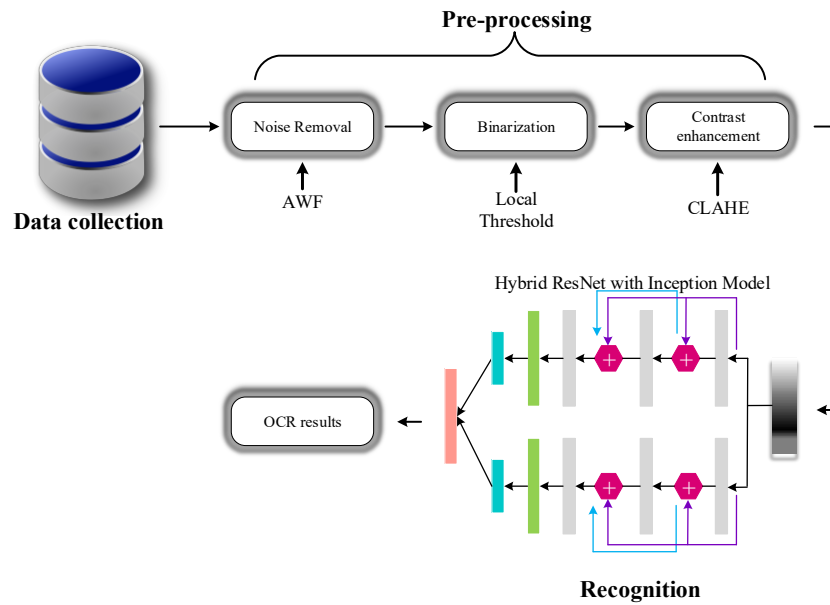


Figure 1: Workflow of the proposed methodology

This research focuses on offline OCR using a collection of Printed (laser and Machine printed) Gujarati characters from different sources like magazines, newspapers and books, etc. Initially, the dataset is pre-processed using different methods like noise removal technique using Adaptive Wiener Filter (AWF), Binarization, and Contrast Limited Adaptive Histogram Equalization (CLAHE) technique for contrast improvement. After that, the classification approach is done with the help of hybrid deep ResNet with Inception model (GoogleNet) architecture, which significantly achieves better recognition accuracy on poor quality text images. This hybrid algorithm can perform better classification results than any other conventional technique.

I. Pre-processing

After data acquisition, it should be properly pre-processed with different techniques. In some cases, the collected data is of poor quality due to the blurred image. So, pre-processing phase is essential to remove noise and variability in the input image. Different kinds of pre-processing methods like noise removal, Binarization, skew detection and correction, and image contrast enhancement are done with the help of different techniques. The character recognition task becomes simpler with a pre-processed image that organizes the image in a correct format [21].

A. Noise removal

Wiener filtering is assumed as one of the best methods to eliminate noise from digital images. Based on a local variance of the image, AWF [22] modifies the output of the filter. The main goal of this method is to minimize the mean square error between the original image and the reconstructed image. Compared to previous filtering techniques, this filtering is very useful to preserve the edges and high-frequency area of the images. In this filtering, some adjustments are created to make the image better. Various window alternatives are applied to deal with different situations and automatically pick the best one. At smooth areas, center sample in the moving window must be ignored to suppress intuitively annoying singularities, but properly utilized in uneven areas. Before performing the AWF method, the images from the datasets are initially transformed into grey scale images.

The AWF is given in equation (1), for a particular pixel location (a_1, a_2)

$$AWF[I(a_1, a_2)] = \mu + \frac{\sigma^2 - \sigma_n^2}{\sigma^2} (I(a_1, a_2) - \mu) \quad (1)$$

where, input image is denoted as I , mean μ and variance (σ^2) are calculated from the set \mathfrak{N} of $(N * M)$ local neighborhood of every single pixel. Hence,

$$\mu = \frac{1}{MN} \sum_{a_1, a_2 \in \mathfrak{N}} I(a_1, a_2) \quad (2)$$

$$\sigma^2 = \frac{1}{MN} \sum_{a_1, a_2 \in \mathfrak{N}} I^2(a_1, a_2) - \mu^2 \quad (3)$$

In addition, the variance of noise is denoted as σ_n .

B. Binarization

In the character recognition task, binarization is considered an important phase. Several amounts of binarization methods are available in previous research in which most of them are utilized for a particular image type. The major objective of this technique is to preserve significant data and reduce the amount of information present in the image. Overall threshold and local threshold are two different classes presented in the gray scale images. In overall threshold, single threshold is utilized in a whole image to create background class and text, while threshold values are determined locally (pixel-by-pixel or area-by-area) in local threshold. Given below expression is used to compute threshold Th of every single pixel locally.

$$Th = (1-k)m + k.m + k \frac{\sigma}{R(m-M)} \quad (4)$$

Here, minimum image grey level is mentioned by M , a standard deviation of all pixels in window is denoted as σ , an average of whole pixels in the window is mentioned by (m, R) mentions the maximum deviation of grayscale on all windows and k is set to 0.5 [23].

C. Contrast enhancement

An enhancement function is offered to all the neighborhood pixels, and a transformation function is acquired from that corresponding pixels. CLAHE [24] method is exploited to maximize the image's contrast. The stages of CLAHE method is explained below:

Stage 1: Input image

Stage 2: Clip limit, distribution parameter type, dynamic range (number of bins in histogram transform function), and number of regions from column and row direction are considered as input data.

Stage 3: Original image is separated into a number of regions.

Stage 4: In tile (i.e. contextual region), apply the process.

Stage 5: Clipped histogram and Gray level mapping are created. Whole pixels of contextual regions are equally distributed in every single gray level. The average number of pixels in a gray level image is represented by the given below expression.

$$M_a = \frac{M_x * M_y}{M_g} \quad (5)$$

Here, M_a mentions the average number of pixels, M_g represents the contextual area in the total gray level, total pixels in direction X and Y of the contextual region is mentioned by M_x , and M_y respectively, finally, M_{CL} and M_{clip} denotes the clip limit and the total number of clips respectively. Given below expression is used to compute the actual clip limit.

$$M_{CL} = M_{clip} * M_a \quad (6)$$

Stage 6:For creating an enhanced image, introduce the gray level mapping. This process exploits four different pixel clusters and applies a mapping function to overlap every single drawing slate over the partially sliced images. This entire process is replicated to accomplish the desired result, which gives improved pixels on image.

II. Recognition

In this research, the Gujarati character identification method is done with the help of a hybrid algorithm, which integrates two different deep learning algorithms like CNN. Different convolution and subsampling layers are tracked by more than one fully connected layer. Normally, a fully connected layer is considered a usual multilayer neural network, and it clasps output which is defined as class score. The input image is convolved by the convolutional layers, which utilize several filters (learnable weights) to convolve the image and the pooling layer down samples the image. Average pooling and max pooling are two different kinds of functions in the pooling layer. CNN converts input images over different stacked layers from original pixels to obtain the final class score. Also, CNN structures are used as building blocks for different semantic segmentation models. In our work, two different deep learning algorithms named ResNet [25] and Inception (GoogLeNet)[26]model are hybridized to perform character recognition in the Gujarati language. Description of ResNet and Inception model are explained in the next section.

A. ResNet

In 2016, Microsoft researchers developed ResNet model, which achieved 96.4% classification accuracy and won the ImageNet Large Scale Visual Recognition Competition (ILSVRC). This network contains 152 deep layers and contains a unique structure that presents residual blocks as shown in figure 2.

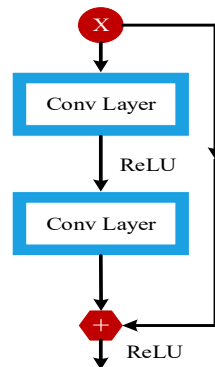


Figure 2: Basic structure of a deep residual network

It also utilize identity skip connections to discourse the problem of training a deep structure. The residual block's function is to copy the input of layers and forwards them into a subsequent layer. The vanishing gradient problem is exceeded with the help of identity skip connection in which an upcoming layer learns something different from the familiar input.

B. Inception model

In 2014, Google researchers developed the GoogleNet architecture, also known as the Inception model. This design won ILSVRC as a top-five with 93.3% classification accuracy. It contains 22 layers and introduces a building block named as Inception model. It does not follow the usual consecutive procedure. However, it exploits the network layer, pooling layer, and large and small convolution layers that are calculated in parallel. For dimensionality reduction, a 1x1 convolution operation is

performed by the convolution layers. By saving memory and computational cost, dimensionality reduction and parallelism process significantly minimize the number of parameters and operations. The building blocks of the core Inception model is displayed in the given below figure 3.

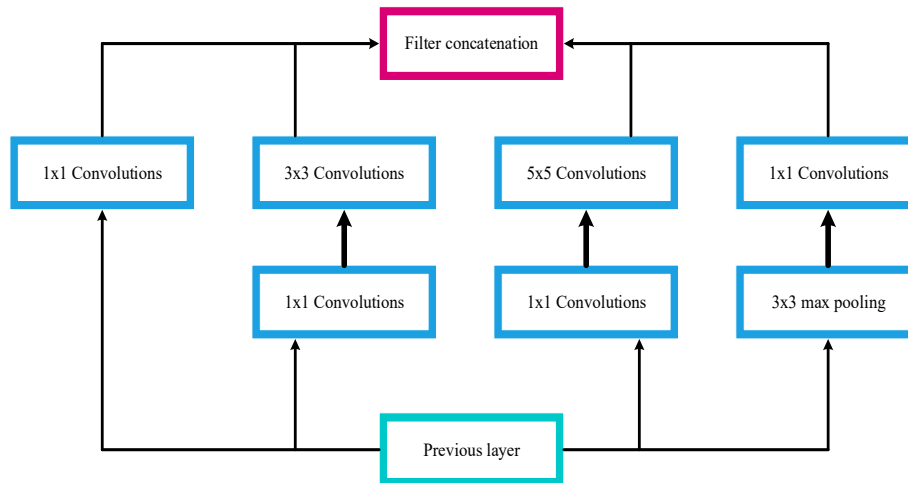


Figure 3: Core blocks in Inception module

C. Hybrid ResNet with Inception model for character recognition

This research combines the benefits of both ResNet and Inception models to improve the classification accuracy of Gujarati characters recognition with this proposed hybrid model. The inception model and residual network demonstrate their capability to increase thousands of layers by offering better performance as well as enhanced efficiency. Many residual blocks with identity mapping are presented in the residual network, and several convolution layers are presented in the deep convolution network named as Inception model. This Gujarati character recognition aims to recognize the character type by allocating and tagging separate pixels with several frequency bands into separate modules. A deep hybrid network structure is developed in this research to absorb deep features of Gujarati characters also offer better recognition accuracy performance without many pre-processing steps. The hybrid structure contains three convolutional layers and one average pooling layer. Outputs of every single layer form the input to every single, consecutive layer. Also, only one fully connected cascaded residual block is presented in the network, which is shown in figure 4.

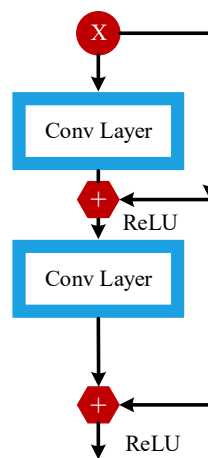


Figure 4: Residual block with fully connected cascaded layers (changed type)

Every convolutional layer accepts inputs from all preceding convolutional layers in the residual model. For our work, three convolutional layers are enough. Generally, the convolution operation is applied to the input data by convolutional layers, and the pooling operation is applied by the average pooling layer in a hybrid model. Also, these operations are performed before feeding data to the classifier. The Adam optimisation algorithm optimises the network model for its faster convergence speed. Also, this technique is a computationally effective one and less complex to sound. For our collected data, the batch size is set to 17, and the initial learning rate is 0.001. Figure 5 demonstrates the general scheme for combined Inception-ResNet modules.

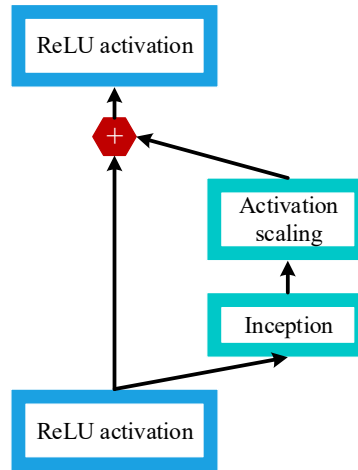


Figure 5: The general schema for scaling combined Inception-ResNet modules.

Convolutional layers

After performing convolution operations to the input image, it transforms the image by rectified linear unit (ReLU) function. Three convolutional layers are utilized in which everyone has nine filters making nine feature maps. Given below, expression (7) expresses each kernel's operation.

$$X^i = \varphi(W^i * X^{i-1} + \beta^i) \quad (7)$$

Here, $*$ is a convolution operator that convolves the filter W^i with the input data X^{i-1} adds the bias term β^i and then applies the rectifier function φ and yields the feature map X^i . The size of the convolutional filter is sixteen units, in which every single layer uses nine filters in the proposed hybrid model. Input is padded by each filter where the output has the same dimensions as the input tensor. The length of the convolution stride is 1. The glorot uniform weight initialization method is utilized in the convolution layers to initialize the weight, and the bias terms are initialized with 0. An element-wise operation is applied by ReLU activation function φ on the input data x , which is defined in the given below expression.

$$\varphi(x) = \max(x, 0) \quad (8)$$

In this work, 1D convolutional kernels are used where each and every pixel is represented as one vector with only one label. The final structure of hybrid ResNetInception architecture is presented in figure 6, and it contains only two residual blocks.

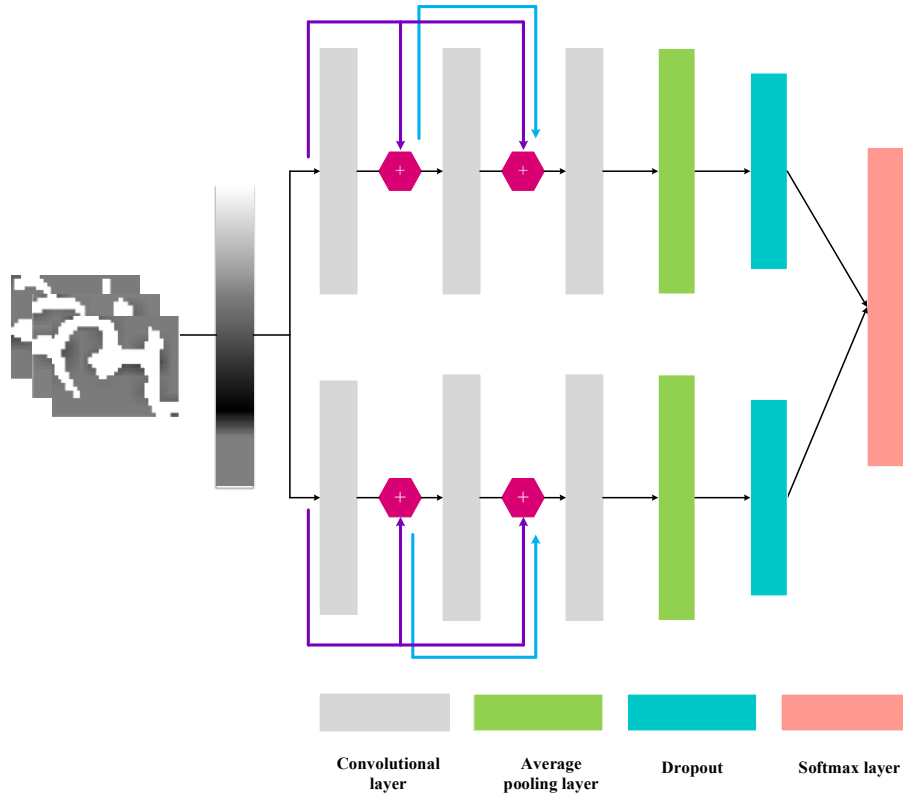


Figure 6: Hybrid deep ResNet-Inception architecture

Two residual models are ultimately connected in this network in which the given below expression express the function of the upper residual model.

$$\begin{aligned}
 X^1 &= \varphi(W^1 * X^0 + \beta^1) \\
 X^2 &= \varphi(W^2 * (X^0 + X^1)\beta^2) \\
 X^3 &= \varphi(W^3 * (X^0 + X^1 + X^2)\beta^3) \\
 X^4 &= AvgP(X^3)
 \end{aligned} \tag{9}$$

Equation 10 defines the function of the lower residual model.

$$\begin{aligned}
 X'^1 &= \varphi(W'^1 * X^0 + \beta^1) \\
 X'^2 &= \varphi(W'^2 * (X'^0 + X'^1)\beta^2) \\
 X'^3 &= \varphi(W'^3 * (X'^0 + X'^1 + X'^2)\beta^3) \\
 X'^4 &= AvgP(X'^3)
 \end{aligned} \tag{10}$$

The parallelism feature of the Inception component is stimulated by the Inception module so that lower and upper residual models work in parallel. Convolution of the operation is done by the first three lines in the equation and third convolutional layer X^3 and X'^3 feeds the output to the average pooling layer, then apply dropout technique.

Pooling layers

Average pooling is performed by only one pooling layer with a stride and filter size of 2. The below expression express this average pooling function. Equation 11 defines the average pooling function.

$$X^i = AvgP(X^{i-1}) \tag{11}$$

Here, $AvgP$ mentions the average pooling function and the input data from the previous convolutional layer is mentioned as X^{i-1} . In neural networks, dropout is performed to minimize interdependent learning between neurons. A dropout technique with a probability of 0.25 is directly

applied in the final stage after performing maximum pooling. Because, there is no fully connected layer apart from the Softmax classifier.

Softmax classifier

In a multi-class problem, Softmax assigns decimal probabilities to each class. Those decimal probabilities must add up to 1.0. The softmax activation function in the output layer obtains the probabilities of each input element belonging to a label and represents a categorical distribution over class labels. In this research, totally 34 classes are assigned to recognize 34 characters in the Gujarati language.

IV. Simulation results and analysis

This section deals with implementing OCR in the Gujarati language using a hybrid deep learning algorithm in the Matlab tool. Performance of the proposed methodology is estimated in terms of different parameters like detection accuracy, precision, recall, F-1 measure and character error rate (CER). Different kinds of existing pre-trained deep learning algorithms like AlexNet, ResNet, and GoogleNet architectures are implemented to equate the performance of the suggested technique.

I. Dataset explanation

To implement this work, the dataset containing Gujarati characters is collected by ourselves from different sources. The prepared dataset contains 10,200 characters containing 34 Gujarati consonants of 300 characters for each. The structure of dataset is displayed in table 2. Printed (laser and Machine printed) Gujarati characters from different sources like magazines, newspapers and books etc., are utilized to collect the dataset. The samples for Gujarati numerals were poised from 300 persons of dissimilar age groups, professional backgrounds and genders. Sample images from the dataset are displayed in figure 7.

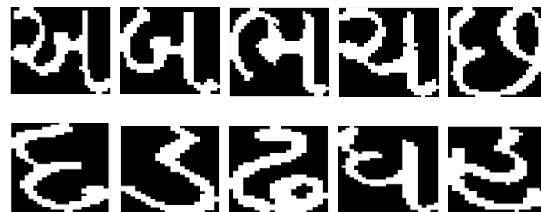


Figure 7: Sample images from the dataset

II. Experiment on dataset

The collected data is divided into 80% for training and 20% for testing in this research. Approaches closely related to our proposed methodology are implemented to equate the performance of our hybrid model. Conventional techniques like AlexNet, ResNet, and GoogleNet are implemented in this work.

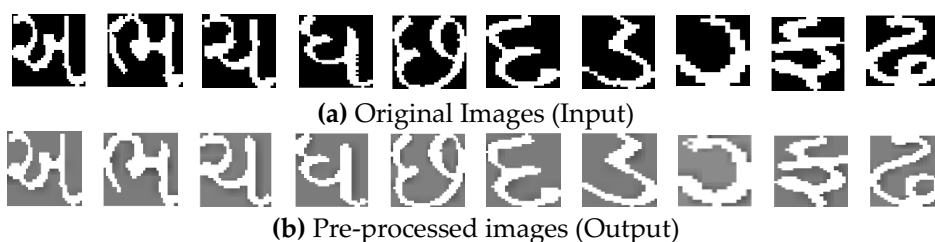


Figure 8: Output images after performing pre-processing steps

Figure 8 displays the original images as well as pre-processed images. This image only contains 10 sample characters from the collected dataset and the pre-processed images of that sample characters.

Table 1: Configuration parameters for proposed hybrid deep neural network

| Layer | Image size | Kernel size | No of filters | Stride | Activation |
|-----------------|------------|-------------|---------------|--------|----------------|
| Input | 28*38 | - | - | - | - |
| Convolution | 24*24 | 5*5 | 9 | 1 | Relu-Inception |
| Convolution | 12*12 | 3*3 | 9 | 1 | Relu-Inception |
| Convolution | 10*10 | 2*2 | 9 | 1 | Relu-Inception |
| Max pooling | 5*5 | 3*3 | 2 | 2 | Relu-Inception |
| Drop out (0.25) | 5*5 | - | - | - | - |
| Output | - | - | - | - | Softmax |

Table 1 mentions the configuration parameter for the proposed hybrid model. This hybrid deep learning algorithm uses three convolutional layers with stride size 1. Also, all the convolutional layer functions with the ReLu-Inception activation function. Drop out technique is applied with a 0.25 probability value. Finally, the output is obtained by joining the Softmax layer as the output layer. In this architecture, the initial learning rate is set as 0.001, and the batch size is 17.

III. Performance analysis

Some standard measures such as Precision, Recall, F1 measure, and accuracy are used to evaluate the performance of the hybrid model, in which they are based on a confusion matrix. The output is either a correctly recognized character or an incorrectly recognized character in the character recognition problem. True Positive (TP), True Negative (TN), False Positive (FP) and False Positive (FP) are four different categories used to estimate the performance of the proposed methodology. TP defines that the actual characters are correctly recognized as actual characters, FP explains that some other characters are incorrectly recognized as actual characters, TN describes that some other characters are correctly recognized as other characters, and FN indicates that the actual characters are incorrectly recognized as some other characters. The performance metrics like precision, recall, accuracy, and F1 measure are evaluated by these four categories. Precision (P) measure is the fraction of all recognized characters to the total number of typescripts in the dataset. The recall is a fraction between correctly recognised characters and the number of characters that should have been recognized. F1 score or balanced F-score is the harmonic mean of precision and recall. Finally, accuracy is a quantity of correctness of the character recognition. In OCR related frameworks, CER is defined as the percentage of inaccurate typescripts in the system output.

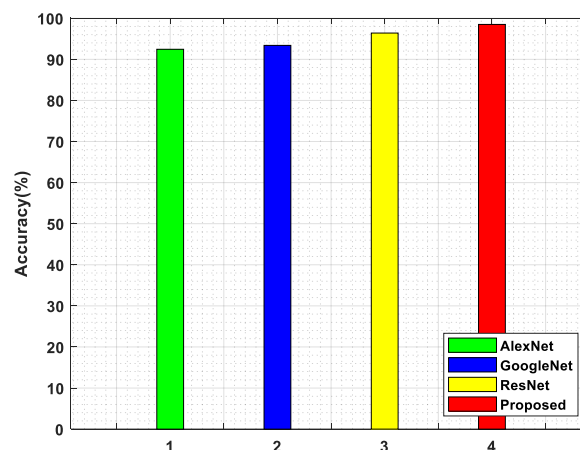


Figure 9: Performance comparison of accuracy

Figure 9 displays the performance comparison of accuracy. Additionally, the performance of the suggested classifier is compared with existing pre-trained models like AlexNet, GoogleNet and ResNet. Our proposed hybrid model exceeds all previous methods in terms of higher accuracy compared to existing methods. For instance, the proposed hybrid methodology obtains 98.5% accuracy, and the existing methods like GoogleNet obtain 93.4%, ResNet obtains 96.4%, and AlexNet obtains 92.45%. The higher accuracy is that the proposed hybrid model only uses three convolution layers for further processing, while the others use a different number of convolution layers. In general, the quality of the results is decreased with more convolutional layers.

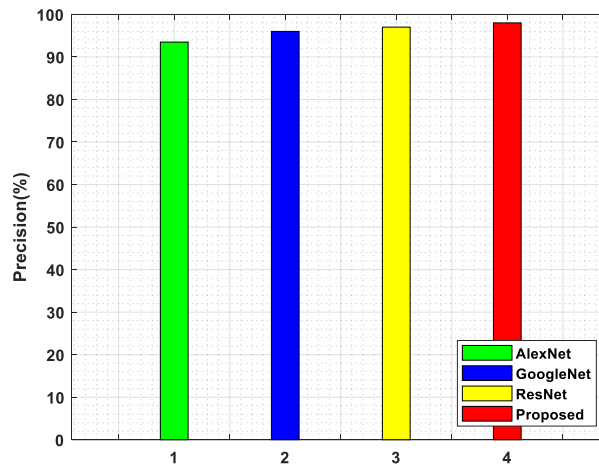


Figure 10: Performance comparison of precision

The comparative analysis of precision metrics is displayed in figure 10. The performance of the proposed hybrid structure is compared with existing methodologies like AlexNet, GoogleNet and ResNet architecture. The figure analysis shows that the proposed methodology obtains 98% accuracy in which GoogleNet obtains 96%, ResNet attains 97%, and AlexNet achieves 93.48%. The proposed methodology exceeds all conventional methods using higher precision value.

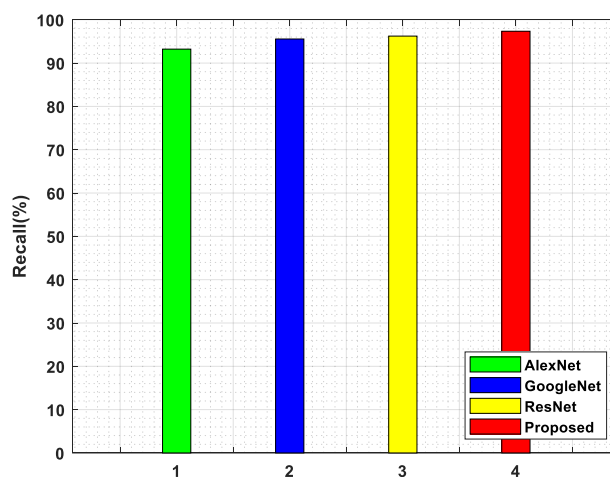


Figure 11: Performance comparison of Recall

Figure 11 demonstrates the performance analysis of recall metric. From the figure analysis, it is shown that the proposed hybrid model obtains 97.35% recall in which existing methods like GoogleNet

obtains 95.56% recall value, ResNet attains 96.23% recall. Finally, AlexNet achieves 93.22% recall value. Compared to existing methodologies, our proposed method obtains high recall in our collected dataset.

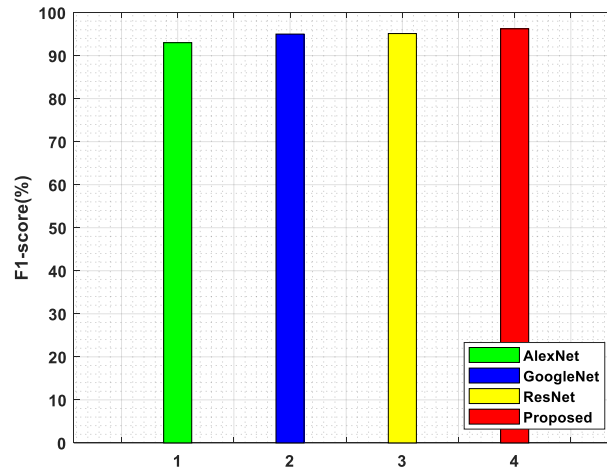


Figure 12: Performance comparison of F1-score

Figure 12 displays the performance comparison of the F1-score. The proposed hybrid model attains a 96.23% score while existing methodologies like AlexNet, GoogleNet and ResNet obtain 93%, 95% and 95.12% F1-score. From the figure analysis, it is clearly shown that the proposed methodology beats all conventional methodologies using a high score value.

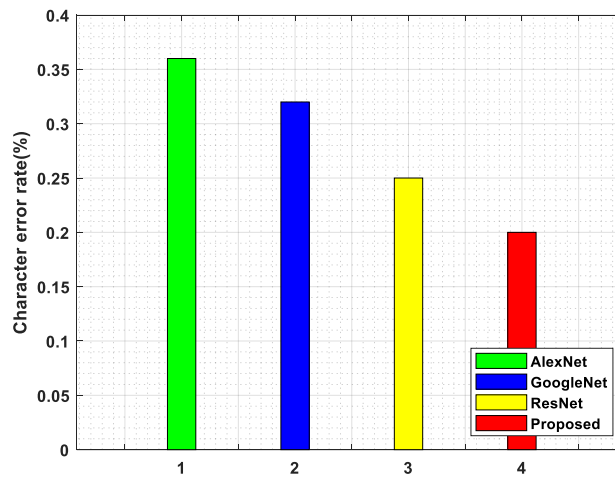


Figure 13: Performance comparison of Character error rate

Figure 13 shows the Character error rate evaluation of the proposed hybrid model with conventional techniques like GoogleNet, ResNet as well as AlexNet classifiers. When comparing the performance of classifiers, the character error rate is minimum for the proposed hybrid model with 0.20%. Existing methodologies like AlexNet, GoogleNet, ResNet obtain 0.36%, 0.32% and 0.25% error which is comparatively higher than the proposed method. The reason for less error is that both feature selection and character recognition process are done by hybrid model, which shows less character recognition rate in the proposed methodology. Moreover, the hybrid model has advantages like robustness, speed learning, and generalization to the same input. These are considered the significance of the hybrid model, which minimizes the error rate in character recognition compared to the existing pre-trained models.

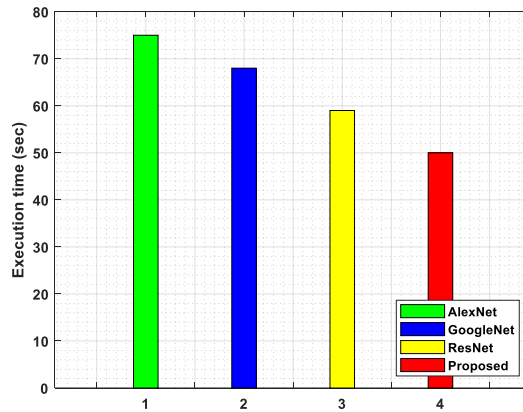


Figure 14: Execution time analysis

Figure 14 shows the execution time analysis of the proposed hybrid model and existing pre-trained models like AlexNet, GoogleNet and ResNet. Compared to the existing pre-trained models, the proposed hybrid model takes 50 seconds to complete the recognition process. This is due to the hybrid model's fast learning process that helps speedily identify the characters.

V. Conclusion

In today's world, most people are e-readers. But, very few e-books are available in the Gujarati language, and most pamphlets are in the form of hardcopy. It needs a digitization method to change those hardcopies into editable text format. OCR is a technique for converting scanned documents into digitized text. So also, for the Gujarati language, the OCR system is required. Plenty of researchers is making efforts to create an efficient OCR system for Indian languages like Marathi, Gujarati and many more. So, in this article, a hybrid ResNet-Inception model is proposed to detect the characters from the collection of scanned documents. Simulation is carried out using the Python tool, and the performance of hybrid methodology is calculated by means of different parameters. Also, it is compared with diverse deep-learning procedures to show the effectiveness of the proposed methodology. The simulation result indicates that the proposed deep hybrid architecture achieves 98.5% accuracy for character recognition, 3.73% higher than ALEXNet, 3.94% greater than GoogleNet and 1.51% superior to ResNet.

References

- [1] Islam, N., Islam, Z., Noor, N. (2017 Oct 3). A survey on optical character recognition system. *arXiv preprint arXiv:1710.05703*.
- [2] Joshi, D.S., Risodkar, Y.R. (2018 Feb 8). Deep learning based Gujarati handwritten character recognition. In 2018 *International Conference On Advances in Communication and Computing Technology (ICACCT) IEEE* 563-566.
- [3] Pareek, J., Singhanian, D., Kumari, R.R., Purohit, S. (2020 Jan 1). Gujarati Handwritten Character Recognition from Text Images. *Procedia Computer Science*. 171: 514-23.
- [4] Naik, V.A., Desai, A.A. (2017 Jul 3). Online handwritten Gujarati character recognition using SVM, MLP, and K-NN. In 2017 *8th International Conference on Computing, Communication and Networking Technologies (ICCCNT) IEEE* 1-6.
- [5] Sharma, A.K., Thakkar, P., Adhyaru, D.M., Zaveri, T.H. (2019 Apr). Handwritten Gujarati Character Recognition Using Structural Decomposition Technique. *Pattern Recognition and Image Analysis*. 29(2): 325-38.

- [6] Sharma, A.K., Adhyaru, D.M., Zaveri, T.H. (2018). A novel cross correlation-based approach for handwritten Gujarati character recognition. *In Proceedings of First International Conference on Smart System, Innovations and Computing. Springer, Singapore.* 505-513.
- [7] Chaudhuri, A., Mandaviya, K., Badelia, P., Ghosh, S.K. (2017). Optical character recognition systems. *In Optical Character Recognition Systems for Different Languages with Soft Computing.* Springer, Cham. 9-41.
- [8] Avadesh, M., Goyal, N. (2018 Apr 24). Optical character recognition for sanskrit using convolution neural networks. *In 2018 13th IAPR International Workshop on Document Analysis Systems (DAS).* IEEE. 447-452.
- [9] Sharma, R., Mudgal, T. (2019). Primitive feature-based optical character recognition of the Devanagari script. *In Progress in Advanced Computing and Intelligent Engineering.* Springer, Singapore. 249-259.
- [10] Bebartta, H.N., Mohanty, S. (2017 Jul). Algorithm for segmenting script-dependant portion in a bilingual Optical Character Recognition system. *Pattern Recognition and Image Analysis.* 27(3): 560-8.
- [11] Audichya, M.K., Saini, J.R. (2017). A study to recognize printed Gujarati characters using tesseract OCR. *Int. J. Res. Appl. Sci. Eng. Technol.* 5: 1505-10.
- [12] Jain, A.A., Arolkar, H.A. A Survey of Gujarati Handwritten Character [1] Recognition Techniques. *International Journal for Research in Applied Science & Engineering Technology (IJRASET),* ISSN. 2321-9653.
- [13] Vakharwala, P., Chhabda, R., Painter, V., Pawar, U., Dastoor, S. (2020 May 15). Performance Analysis of Various Trained CNN Models on Gujarati Script. *In International Conference on Information and Communication Technology for Intelligent Systems.* Springer, Singapore. 483-492.
- [14] Althobaiti, H., Lu, C. (2017 Mar 22). A survey on Arabic Optical Character Recognition and an isolated handwritten Arabic Character Recognition algorithm using encoded freeman chain code. *In 2017 51st Annual Conference on Information Sciences and Systems (CISS).* IEEE 1-6.
- [15] Ahmad, I., Wang, X., Li, R., Rasheed, S. (2017 Feb 2). Offline Urdu Nastaleeq optical character recognition based on stacked denoising autoencoder. *China Communications* 14(1): 146-57.
- [16] Sethi, R.K., Mohanty, K.K. (2020 July 07). Optical Odia Character Classification using CNN and Transfer Learning: A Deep Learning Approach.
- [17] Shinde, A., Dandawate, Y. Convolutional Neural Network Based Handwritten Marathi Text Recognition.
- [18] Naik, V.A., Desai, A.A. (2019). Multi-layer classification approach for online handwritten Gujarati character recognition. *In Computational Intelligence: Theories, Applications and Future Directions- Volume II.* Springer, Singapore 595-606.
- [19] Jena, O.P., Pradhan, S.K., Biswal, P.K., Nayak, S. (2020 Mar 13). Recognition of Printed Odia Characters and Digits using Optimized Self-Organizing Map Network. *In 2020 International Conference on Computer Science, Engineering and Applications (ICCSEA) IEEE.* 1-6.
- [20] Das, D., Nayak, D.R., Dash, R., Majhi, B. (2020 Nov). MJCN: Multi-objective Jaya Convolutional Network for handwritten optical character recognition. *Multimedia Tools and Applications* 79(43): 33023-42.
- [21] Bui, Q.A., Mollard, D., Tabbone, S. (2017 Nov 9). Selecting automatically pre-processing methods to improve OCR performances. *In 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR).* IEEE 1: 169-174.
- [22] Suresh, S., Lal, S., Chen, C., Celik, T. (2018 Apr 10). Multispectral satellite image denoising via adaptive cuckoo search-based wiener filter. *IEEE transactions on geoscience and remote sensing.* 56(8): 4334-45.

- [23] Kaundilya, C., Chawla, D., Chopra, Y. (2019 Mar 13). Automated text extraction from images using OCR system. In 2019 6th *International Conference on Computing for Sustainable Global Development (INDIACom)*. IEEE 145-150.
- [24] Garg, D., Garg, N.K., Kumar, M. (2018 Oct). Underwater image enhancement using blending of CLAHE and percentile methodologies. *Multimedia Tools and Applications*. 77(20): 26545-61.
- [25] He, K., Zhang, X., Ren, S., Sun, J. (2016). Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770-778.
- [26] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A. (2015). Going deeper with convolutions. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1-9.