Hybrid networks for handwritten word recognition in JPEG compressed domain

Moksh Grover[1], Abhishek Kumar Gupta[2], Ayush Raj[3], Utkarsh Priyam[4]
[1]Department of IT, Indian Institute of Information Technology, Allahabad, U.P - 211012, India
Email: [1]iit2018186@iiita.ac.in [2]iit2018187@iiita.ac.in [3]iit2018188@iiita.ac.in [4]iit2018197@iiita.ac.in

***Abstract*—The ability for an automated system to receive and interpret messages from a handwritten format has been a field of interest for the past few decades.The concept discussed in this paper uses a hybrid approach combining the aspects of both RNN and CNN to achieve the same in the case of images in JPEG compressed image domain**

***Index Terms*—compressed domain, hybrid networks, handwritten , word recognition**

I. INTRODUCTION

Handwritten Word Recognition is one of the most intriguing areas of research in computer sciences domain but due to the vast amount of differences in the writing styles of various people it tends to be a bit difficult too [1]. Satisfactory results have been found in isolated word recognition and character identification, though the current science is still very inept from a stage where we have a really good performing automation for the texts which are stored and then fed altogether,i.e., texts which are offline in nature and hence which have no constraints imposed upon them [2].

General approaches in handwritten word recognition use the concept of preprocessing to reduce the vast amount of variation that are present in different ways in which the words are present. The common methods of preprocessing depends upon the normalization of size of characters and correction of the slopes of various letters available in the given image [3]. As the images in the compressed domain are somewhat different to view as they do not contain the words that are generally present and can be understood by the human eye, but the features are very difficult to be fed to the model individually. So, we rely on RNN and CNN for the feature extraction to the output classification after some steps of preprocessing had already been done to perform the word segmentation.

A. Literature Survey

A lot of papers were included in the research comprising various methodologies mixed with various advantages and disadvantages such as, in paper [1] they used an ANN trained with scaled conjugate gradient alongside Resilient Back Propagation to get a final accuracy score of 95% on manually created dataset. Preprocessing comprises of various operations on images as tilting. Then the tilted images were further segmented. As the method suggested fewer features it took slightly less time for computation. In [2] an approach was made using a CNN-RNN along with a CTC layer along with a spell checker to suggest possible options thus achieving an accuracy of 90.3% on the IAM dataset. In [3] the model preciously discussed was tweaked a little by applying some modifications such as the use of a CNN-RNN model.The model used dummy data and was utilized for achieving good efficacy by keeping some key points into consideration such as smart initial weight assignment to the architecture, images tilted were normalized and respectively for various domains the data was changed along with some extortion for finding some key changes to be further utilized in the model.In paper [4] a hybrid novel approach was proposed which used a one dimensional LSTM at its core for one of the basic components. It tooka number of lines for it as its input with some preprocessing, that is the lines were normalized on the basis of their geometry. The output thus generated was fed into CTC layer for final layer. The paper thus concluded on a note that the simple one dimnensional LSTM is outperformed by approach where a collection of various layers of convolution, 1D LSTM along with some max pooling between consecutive layers were used.In [5]a method which uses word beam search decoding was used. This approach greatly affected the results as the words were constrained to be those which were already present in the set of known words. Though, it allowed some results between the words from the dictionary which were not present in the dictionary. As it used a model for the language used the time used was significantly less than the time taken in the method which utilized token pass at its core.For the proposed WBS a prefix tree was used to query the characters . In [6] a brief overview of various methodologies was carried out for study of content based image retrieval uncompressed domain, all on the MPEG dataset and hence concluded that CBIR possesses approximate 15% less computational complexity in compressed image domain as compared to normal images.

II. PROBLEM STATEMENT AND OBJECTIVES

In the domain of Handwritten Word recognition, recently it was proved that using hybrid architecture as convolutional recurrent architecture where the convolutional layers are for extracting the features , which are further provided to a RNN layer with CTC loss, work better than other hybrid model,but main pitfall is the high time complexity of neural networks, but since there is lesser amount of data in the compressed domain than in the original uncompressed domain there is great scope for reducing overall time complexity of the model. So, our main objective is train the CNN-RNN hybrid network on compressed images

III. PROPOSED METHODOLOGY

a) Handwritten Word Recognition in DCT compressed Domain: Handwritten Word Recognition (HWR) model first convert the scanned image into DCT compression then detect that image into text as shown $Fig.$ 1.
We trained a Nueral Network(NN) in DCT compressed image of used Dataset(IAM).

Fig. 1. prediction of word

b) Model Overview: We are using a hybrid network to achieve handwritten text recognition. As shown in $Fig.$ 2, it mainly consists of 4 parts

- preprocess and compress the all images in the Dataset
- Multi-scale feature Extraction: using CNN(5 layer)
- Sequence Labeling: using RNN (2 LSTM layer)
- Transcription: Decoding the output of the RNN.

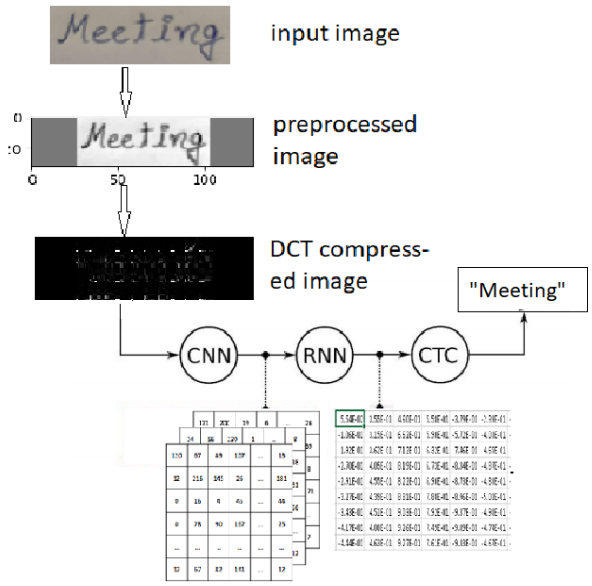

Fig. 2. Overview of current model.

c) Operations:

- Cropping:cropped into centre.
- Gaussian blur: by averaging value.
- Increasing contrast: increasing contrast and applied threshold give good results.
- Morphological operations: give better and more accurate results.
- JPEG compression: includes steps such as splitting and the application of DCT (Discrete Cosine Transform).
- CNN: Now,compressed image fed to 5 layer of CNN in which top two layer have 5x5 kernel and last 3 layer have 3x3 kernel alongside with relu and maxpool operation.
- RNN: The result on performing RNN is then fed to a 2D array of dimension (32,80). The dataset which we are using(IAM) has 79 element or chars ,but we need one more character i.e (blank character label in case time-step have no character in it). So, there is total 80 element. For each time step and highest probability among 80 element will select as predicted character.
- CTC: When we train our hybrid network, the CTC would be given to the RNN output vector along with the original truth word and it calculates the CTC loss amoubt. The CTC layer is then provided the output vector which then translates it to resultant text. The length of input and output text are considered to be less than 2^5.

IV. MODEL ARCHITECTURE

The proposed model has different layers as per given $Fig.$ 3 below:

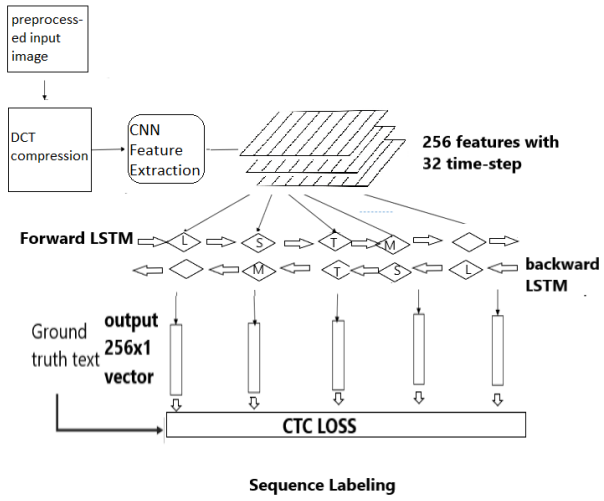

Fig. 3. Architecture of proposed Model

A. Pre-Processing

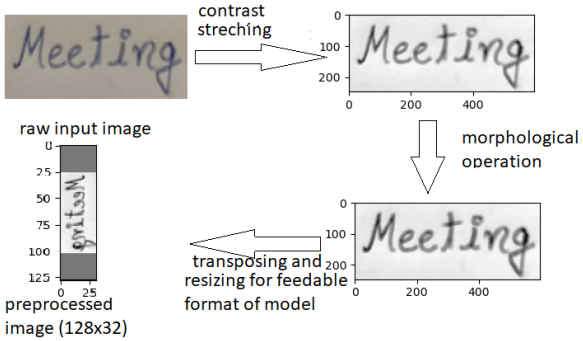

Fig. 4. Pre-Processing

a) Contrast Stretching: Contrast stretching generally stretches the intensity (higher pixel increases its intensity and lower pixel value would be decreases its intensity). Hence, improving the image quality as shown in first step of $Fig.$ 4.

$$g'(x,y) = \text{INT}\left\{\frac{255}{GL_{max} - GL_{min}}\Big[g(x,y) - GL_{min}\Big]\right\}$$

where,

$g(x,y)$ is pixel value and GL is the grey level value.

b) Morphological operation: Erosion is the process by which an image is eroded by the help of a kernel which strides over the image and with some other operations provides us with the eroded image.$Fig.$ 4.

c) Resizing and Transpose : Resizing and Transpose of the input image to 128x32 is done,which would be fed to the model as shown in last step of $Fig.$ 4.

B. DCT Compression

The Discrete Cosine Transform is generally used for reduction in bandwidth of image in image pre-processing.It is also known as block compression.we can use 8x8 block (standard) with varied block size of 4x4 and 32x32. In our case both 8x8 and 4x4 block compression would be considered.

$$DCT(i,j) = \frac{1}{\sqrt{2N}}C(i)C(j)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}f(x,y)$$

where,

$$f(x,y) = pixel(x,y)\cos\frac{(2x+1)i\pi}{2N}\cos\frac{(2y+1)i\pi}{2N}$$

$$C(x) = \frac{1}{\sqrt{2}} \text{ if } x \text{ is } 0, \text{ else } 1 \text{ if } x > 0$$

C. CNN

CNN can successfully learn the extraction of 2D elements and visualize the human prepossessing. In specific, the max-pool layer used to detect the variation in image(detect the brighter pixel of image in that case) and the small reference weight of the straps allows CNN to include fewer parameters than a other method like fully connected layer. More specific CNN trained over gradient based algorithm which avoid direct error in the model.That is why CNN can make highly efficient machine and give better all over performance. In our case CNN gets a compressed image as input which it processes to provide a set of 256 features to RNN.

D. RNN

RNN (Recurrent Neural Networks) is a case of artificial neural networks where the connection between various nodes follow a temporal order and which can also be used for processing images of variable length images.Here 32 stamps achieved from the previous functions are mapped with the ground text which will be then sent to the LSTM for further ado. The mathematical foundations for this are bounded by the equations such as: Current state:

$$h_t = f(h_{t-1}, x_t)$$

applying activation function:

$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t)$$

calculation of output:

$$y_t = W_{hy}h_t$$

E. CTC

CTC (Connectionist temporal classification) can be considered as a loss function for the purpose of mentoring the model which works on by adding the probabilities of all probable alignment between the label and the input and is used in our model for the same.
The CTC loss calculates as follow:

$$p(\text{Y}|\text{X}) = \sum_{\text{A}\in\Omega_{\text{X,Y}}}\prod_{\text{t=1}}^{\text{T}}p_t(a_t|\text{X})$$

, where X and Y are input and output sequence lengths T and U respectively. Set Ω consists all possible length sequences with length T and a_t is the current state.

V. DATASET DESCRIPTION

The dataset used here was taken from a journal disclosed by the International Conference on Document Analysis and Recognition(ICDAR) and the name of the dataset s IAM Handwriting Dataset. The dataset primarily comprises of images of handwritten words in English. The dataset procurement consisted of tedious contribution of 657 people who gave their specimens of handwritten text. The text accumulated to 1539 pages after being scanned. The text pages when segmented amounted to 5685 sentences which were discrete and had labels. Further segmentation broke it into 13353 lines which were discrete and had labels which further provided us with 115,320 words which were discrete and had labels. For the purpose of breaking words and for their extraction an automatic segmentation method was used and the results were checked one by one by people.The Institute as mentioned in[7] contributed to the birth of the segmentation technique and its approval

VI. EXPERIMENTS

To measure effectiveness of our proposed model we did experiment on standard benchmarks and compare the results and techniques. Section 6.A give implementation of proposed model and Section 6.B give comparative evaluation with results with other model based on handwritten recognition.

A. IMPLEMENTATION

Input: It is a gray-scale image of dimension (128,32). Images in the database do not possess this same size of (128,32), so it is resized until the same dimension is achieved. After that image is put on a white area of 128 x 32 dimension. ,as explained in $Fig.$ 5 At the end, the gray values of the image will be easily processed in order to simplify the work of neural network. On selecting different random coordinate, data enrichment technique can also be added here by copying the image there, instead of left-aligning it. *DCT compression:* This

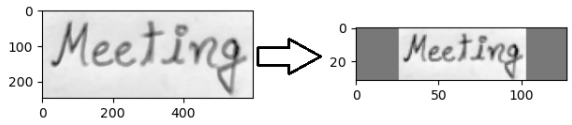

Fig. 5. preprocessing of image

a

operation transform the preprocessed image into compressed image after applying 8x8 block compression as shown in $Fig.$ 6 below. *CNN:* CNN layers that can be in sequence

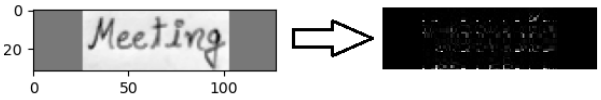

Fig. 6. DCT compression of Image

length 32. Each layer installation has 256 elements these feature subsequently given to BLSTM layer.out of 256 feature some have high correlation with properties of image.
RNN: The score of characters, with a blank space at end as denoting end character is represented in the matrix. These scores are calculated by Connectionist Temporal Classification. The matrix transpose has input as these letters: "!"#&'()*,./ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789:;? " ,when seen from up to down.
CTC: The true text is encoded in the form of tensor. On calculation of the loss, the true text in both cases are fed as shown in Fig 5. upon which the required operations follow. The sequence length is fed in these CTC functions, which is shown in $Fig.$ 7

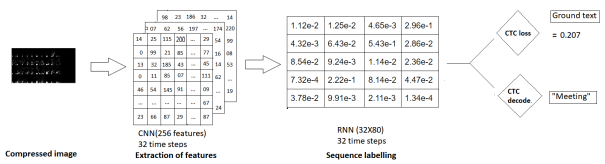

Fig. 7. feature extraction and sequence labelling

B. COMPARATIVE EVALUATION

The comparative evaluation of the CRNN model used in [3] and the implementations consisting of CRNN in pixel domain and CRNN in compressed image domain using 8*8 and 4*4 block size in dct compression is depicted in Fig 8.

From the results one can easily see that the method implementation used by us outperforms the base paper implementation which is trained on the same IAM dataset.

The main reason for performing better owes to the usage of usage of bi-directional LSTM (Long Short Term Memory) which uses sequence labelling which helps in the detection of a word by using some knowledge, which words are feasible and which are not thus helping in the elimination of impossible words and which when used along with the CTC loss function which is used for a proper evaluation of how the retraining is to be done by the use of losses calculated using the CTC function which also helps in the alignment problem generally faced.

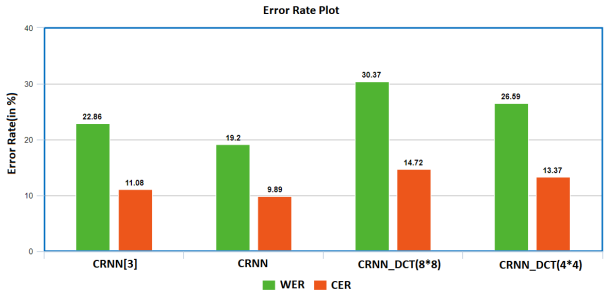

Fig. 8. Error Rate Plot

Table 1 and Table2 is comparative model configuration summary of our proposed model and previous work done in [3]. BLSTM followed by word beam search CTC decoder perform better than MDLSTM with best path search.

TABLE I
MODEL CONFIGURATION OF OUR PROPOSED MODEL

Type	**Description**
Input	gray-value line-image (128 x 32)
Conv+Pool	#map 32 kernel 5 x 5, pool 2 x 2
Conv+Pool	#map 64 kernel 5 x 5, pool 2 x 2
Conv+Pool+BN	#map 128 kernel 3 x 3, pool 1 x 2
Conv+pool	#map 128 kernel 3 x 3,pool 1 x 2
conv+pool	#map256 kernel 3 x 3, pool 1 x 2
Collapse	remove dimension
Forward LSTM	256 hidden unit
Backward LSTM	256 hidden unit
Project	project into 80 classes
CTC	decode or loss

TABLE II
MODEL CONFIGURATION OF CRNN MODEL[13]

Type	Description
Input	W × 32 gray-scale image
Conv+Pool	#map 64 kernel 3 × 3, poo1 2x2
Conv+Pool	#map 128 kernel 3 x 3 , pool 2 x 2
Conv	#map 256 kernel 3 x 3
Conv+Pool	#map 256 kernel 3 x 3, pool 1 x 2
Conv+BN	#map 512 kernel 3 x 3
Conv+Pool+BN	#map 512kernel 3 x 3, pool 1 x 2
Conv	#map 512 kernel 2 x 2
Collapse	remove dimension
MDLSTM	256 hidden unit
Project	project into 80 classes
Transcription	–

VII. RESULTS

A detailed study of papers was done to understand the concepts of image compression and a brief study of papers also helped in selecting the hybrid network (CNN-RNN) to achieve our goal. We have done our experiment without using compressed domain images and then extended the work to compressed domain with block 8x8 and 4x4 kernel as shown in $Table3$.

TABLE III
STUDY OF THE CNN-RNN ARCHITECTURE

Method	WA	WAF	CER
CNN-RNN (simple)	80.08	92.8	9.89
CNN-RNN-DCT (8*8)	69.63	85.76	14.72
CNN-RNN-DCT (4*4)	73.41	89.05	13.37

We used the character error rate (CER) to analyse our model. CER is defined as (where GT:ground truth and PT:predicted text):

$$\text{CER} = \frac{\sum_{i \forall \text{samples}} \text{EditDistance}(GT_i, PT_i)}{\sum_{i \forall \text{samples}} \#\text{Chars}(GT_i)}$$

edit distance is algorithm to check the similarity between two string.
We use word accuracy to analyse our model. WAF and WA is the Word Accuracy without and with Flexibility used.
We assume a word is predicted correctly if value of edit distance between two word is less than 3 where the edit distance function is the function which takes two strings as an input and finds the minimum number of insertions, deletions or changes in one of the string to transform it into the other string, where the cost of each of the aforementioned operations is 1 and the sum of all the costs is returned to the user and is used as the edit distance.

Predicted text for few images with random handwriting are shown in $Fig.$ 9.

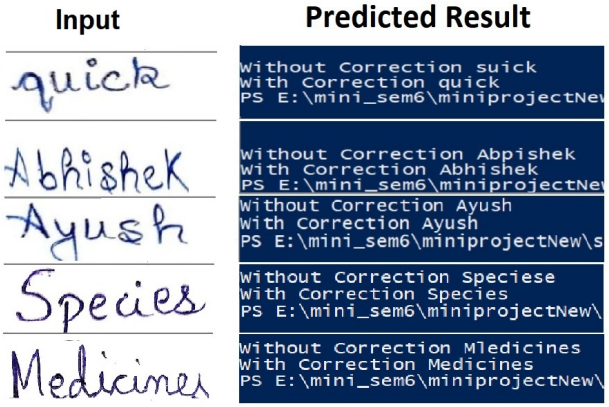

Fig. 9. Predicted Results

VIII. CONCLUSION

In this work, we presented a CNN-RNN hybrid architecture trained on images that are in DCT compressed domain.The accuracy was achieved as expectations and the computational time required decreased when compared to HWR without DCT compression thus saving a whole lot of valuable time.When DCT compression was applied by using (4*4) lesser collision of blocks containing letters occur as compared to the DCT compression using (8*8) kernel, hence an increase in accuracy was seen when a kernel size of (4*4) was considered.

REFERENCES

[1] Obaid, A. M., et al. "Handwritten text recognition system based on neural network." Int. J. Adv. Res. Comput. Sci. Technol.(IJARCST) 4.1 (2016): 72-77.
[2] Manchala, Sri Yugandhar, et al. "Handwritten text recognition using deep learning with Tensorflow." International Journal of Engineering and Technical Research 9.5 (2020).
[3] Dutta, Kartik, et al. "Improving cnn-rnn hybrid networks for handwriting recognition." 2018 16th international conference on frontiers in handwriting recognition (ICFHR). IEEE, 2018.
[4] Breuel, Thomas M. "High performance text recognition using a hybrid convolutional-lstm implementation." 2017 14th IAPR international conference on document analysis and recognition (ICDAR). Vol. 1. IEEE, 2017.
[5] Scheidl, Harald, Stefan Fiel, and Robert Sablatnig. "Word beam search: A connectionist temporal classification decoding algorithm." 2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR). IEEE, 2018.
[6] D. Edmundson and G. Schaefer, "An overview and evaluation of JPEG compressed domain retrieval techniques," Proceedings ELMAR-2012, Zadar, Croatia, 2012, pp. 75-78.
[7] U. Marti and H. Bunke. A full English sentence database for off-line handwriting recognition. In Proc. of the 5th Int. Conf. on Document Analysis and Recognition, pages 705 - 708, 1999.
[8] Espana-Boquera, Salvador, et al. "Improving offline handwritten text recognition with hybrid HMM/ANN models." IEEE transactions on pattern analysis and machine intelligence 33.4 (2010): 767-779.
[9] Kim, Gyeonghwan, Venu Govindaraju, and Sargur N. Srihari. "An architecture for handwritten text recognition systems." International Journal on Document Analysis and Recognition 2.1 (1999): 37-44.
[10] Ingle, R. Reeve, et al. "A scalable handwritten text recognition system." 2019 International Conference on Document Analysis and Recognition (ICDAR). IEEE, 2019.