International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 10 | October - 2023 SJIF Rating: 8.176 ISSN: 2582-3930

Handwritten Character Recognition Using Deep Learning

Sukesh N D[1], Steephan Amalraj J[2,]

[1]*Student, Department of Artificial Intelligence and Data Science,*
Bannari Amman Institute Of Technology

[2] *Assistant Professor,Department of Computer Science Engineering*
Bannari Amman Institute Of Technology

Abstract - Handwritten digit or character recognition in transforming the printed or handwritten text from an image. Optical character recognition plays an important role in documentation scanning ,text extractions from the image. Optical character recognition is used in different fields like postal services ,Ecommerce , Shipping ,Banking sector for character extraction from the images . However the existing character recognition system faces many challenges in extracting text from noisy and distortion images or complex layout and Extraction mostly limited to numbers and English alphabets . The introduction of Deep learning has changed Optical Character Recognition by using models like Recurrent Neural Networks,convolutional neural network .In this paper i am gonna compare the different models like CNN model and CRNN model with current State of art model Transformer based Optical Character Recognition

KeyWords **:**Transformers ,Convolution Recurrent Neural Network, Handwritten ,Optical Character Recognition .

1.INTRODUCTION

With the rapid development in the field of Artificial Intelligence (AI) , Computer Science and enormous amounts of data produced Daily the machine learning and Deep learning Models have made our ways easy in many ways like self driving cars ,personal voice assistants. Handwritten Character recognition also have become high interest because of its use case in multiple domains like digitalization of old documents,Reduce human errors in postal services , E-commerce and Banking domain Optical Character Recognition is the process of making computers understand human Handwritten Characters within the image .Converting the visually represented text into machine readable and digit format text. The many issues in creating a perfect Optical Character Recognition model due to variety of handwriting styles and font ,Noise and distortion in the image making it difficult to recognize the text part , developing a multi-language model to recognize different languages,Insufficient image Dataset of every language for Training the model, Insufficient Performance of special Characters that are not frequently repeated and historical image may contain letter or words that are no longer in use and doesn't have a sufficient data to train the model.In the past, OCR systems were typically based on template matching or feature extraction techniques. However, these techniques have been surpassed by deep learning-based OCR systems, which are able to learn the relationships between characters and their features.

In the past the Convolution Neural Network has been widely used for research in OCR but CNN is only able to capture patterns at a certain level but still there is lot of room for improvement in OCR.In this paper i have compared the Convolution Recurrent Neural Network architecture and Pre-trained Transformer based OCR(Minghao Li et al 2022) on Optical Character Recognition. Convolution Recurrent Neural Network is the combination of a Convolution Neural Network (CNN) and a Recurrent Neural Network (RNN), where the CNN part of the model is used for image processing. This involves extracting intricate features from the image and converting the visual input into a compact feature vector. This feature vector is then sequentially processed by the RNN's Recurrent layers, often employing structures like Long ShortTerm Memory (LSTM) or Gated Recurrent Unit (GRU). These Recurrent layers are particularly adept at recognizing temporal patterns, allowing them to decipher and identify meaningful words and text elements within the image context. Transformers architecture was first released in the paper "Attention is all you need" (Ashish Vaswani el al 2017).Transformer model consists of a decoder and encoder part. The encoder part consists of self attention layer , Feed forward layer and a normalization layer and the decoder part consists of self attention layer , feed forward layer ,normalization layer and encoder decoder attention layer. Initially, the image is divided into a sequence of patches. Positional embedding is then applied to each patch, and the output is processed by an encoder selfattention layer. A typical transformer architecture contains multiple encoder and decoder layers. A Typical transformer architecture layer contains N-number of encoder and decoder in their architecture

2.LITERATURE SURVEY

1. Trupti R. Dilhiwala et al. (2023) published a paper on handwritten digit recognition using convolutional neural networks (CNNs). The MNIST dataset, which consists of 28x28-pixel images of handwritten digits, was utilized in their study. They explained the various layers of CNNs, such as the convolution layer, which uses several filters to execute convolution operations. They used a rectified linear unit (ReLU) as an activation function. The pooling layer downsampled the output, and finally, the fully connected layers receive the flattened matrix from the pooling layer as input and produce the output. The pixel values were originally between 0 and 255, but they were normalized between 0 and 1 for improved performance. They achieved an accuracy of 99

2. k.Swetha et al. (2021) studied the uses and main challenges of implementing a model that converts images of handwritten digits into digital format.

3.They compared many algorithms, such as KNN, random forest classifiers, SVM, logistic regression, and CNN, for handwritten digit recognition. They used OpenCV to read and manipulate images, and converted the images into grayscale. The noise in the images was removed using Gaussian blur. Finally, the models were trained with this data. Their results showed that CNN performed comparatively well compared to the other algorithms they used in terms of accuracy

4. Ritik Dixit et al. (2021) studied and worked with different algorithms for handwritten digit recognition, including support vector machines (SVM), multi-layer perceptrons (MLPs), and 17 convolutional neural networks (CNNs). They compared the accuracy, error, and training and testing times of these algorithms using the MNIST dataset, which consists of 70,000 28x28 pixel images of handwritten digits. The authors first normalized the images by converting each pixel from 0.0 to 1.0. They then performed one-hot encoding on the output variable and used this data to train the different models. They found that SVM had the highest accuracy on the training dataset, but CNN achieved the highest accuracy on the testing dataset . SVM also had the shortest execution time, while CNN had the longest execution time. The training rate, testing rate, and execution time for each algorithm were as follows:

MODEL	TRAINING RATE	TESTING RATE	EXECUTION TIME
SVM	99.98%	94.005%	1.35min
MLP	99.92%	98.85%	2.32min
CNN	99.53%	99.31%	44.02min

The authors concluded that CNN is the best algorithm for handwritten digit recognition in terms of accuracy, but SVM is a better choice if execution time is a priority.

5. Fathma Siddique et al(2020) studied the accuracy of various hidden layers and number of epochs of CNN for better accuracy in building a model for HandWritten Digit recognition using the MNIST dataset.The accuracy of the CNN model is varying based on the hidden layers that model contains.The accuracy increase till the hidden layer count increases till the count become 4 and then he checked the same with number of epochs the model trained based on his paper the number of epochs increases the accuracy also increase but it may also lead to overfitting .

6. Nirmal S Guptha et al(2022) student the the Handwritten character on 64 english English class and 10 class on Arabic Language and 657 classes in Kannada character recognition on Long short term memory Recurrent neural network on two dataset chars74k and MADbase digits dataset. He proposed an optimizer named Elephant herding optimization algorithm with LSTM model he achieved a accuray of 96.67 on kannada Data ,99.66 on English 64 classes and 99.93 accuracy on 10 arabic classes.He clearly mentioned the process involved in the Optical Character Recognition is collecting data ,Line and individual character segmentation,Data pre-processing ,Feature extraction by inverse difference moment normalized (IDMN) and Enhanced Local Binary Pattern(ELBP), feature optimization using EHO algorithm , character classification by Long short term memory network and that model can classify 10 Arabic languages,657 classes in kannada language and 64 classes in English language .

6.Ashish Shetty and Sanjeev Sharma(2023) have clearly explained the use of optical character recognition on different domains like document scanning and books scanning . He tells the importance of Convolution neural networks on pattern recognition tasks. They mentioned the process involved in the Handwriting Character Recognition tasks. They clearly explained the use of different layers of CNN in OCR tasks. He have totally compared Totaly 7 models of different architectures of CNN namely ResNet , InceptionV3, DenseNet201, DenseNet121, Xception, R+I (ResNet + Inception model) and finally an ensemble model combination of DenseNEet121 , Xception and R+i model. The ensemble model outperformed the normal models on the Test Dataset.They have used Relu as an activation function on different layers of the Neural Networks and softmax for the output layer.

7. Minghao Li et al(2022) had proposed a transformer based end to end Optical Character Recognition system . Their transformer model contains an image based Transformer for extracting visual features and a text based transformer for language model . Their model obtains 96.58 F1 score whereas the CRNN got 36.09 and Tesseract Ocr got 54.57

3.METHODOLOGY

A) **Data Acquisition :**

Fig. 1. Images In MNIST Dataset

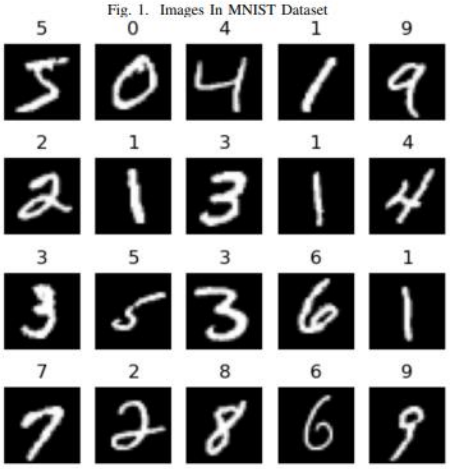


Fig. 2. Handwritten Character Image - CAROLINE

The first process of building a deep learning and machine learning models is collecting the dataset for model training here we have used MNIST Dataset for VGG-collected 3 lakhs image of handwritten character ,41k test data and 41k validation data where i used 30k training data and 30 k validation data while model training for training thee CRNN model and for TRocr is a pre-trained model available in Hugging face for research..

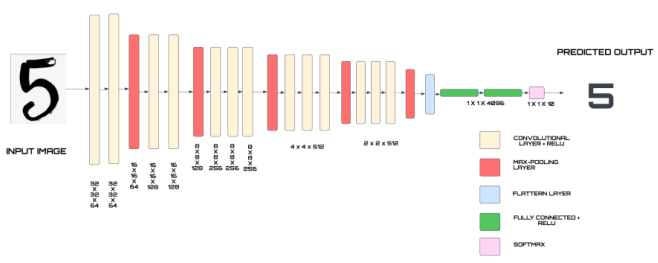
B) Data Preprocessing and Exploration:

Splitting the MNIST dataset for training(60,000) and testing (10,000) . The images in the dataset is 28*28 pixels of grayscale image which is converted into 32*32 pixels of grayscale image from model training. Normalization of pixel values in the image to the range of 0 to 1 by dividing the value by 255 In the text image dataset the dataset is loaded and removes the images from the data which are UNREADABLE. Converted the letters in the labels to the lowercase letters ,The image is reshaped into 64*256 pixels for model training Creating labels for representing the letters in the alphabets consists of a-z .

C) Building Convolution neural networks:

Build the VGG-16 model for predicting Handwritten text images trained using MNIST Dataset It consists of 16 layers where 13 is convolution layers and 3 dense layers and 23 max pooling is used as pooling layer The above figure shows the VGG-16 layer used for handwritten digit recognition. It consists of 16 layers, a combination of 13 convolution layers and 3 dense layers and max pooling layer and uses Relu as an activation function The initial two convolution layers are responsible for detecting the basic features in the input image such as corners edges of the text in the input images which is initially used for feature extraction in the images. Kernel or also know as filters are used for feature extraction there are many filter which values are initially assigned by the model while training .The convolution is the process of combining the two and gives one output .The same process is followed in the convolutional layer where the filter slide above the image part by part and one by one do matrix multiplication operation between the image and filter which is used for feature extraction process . The pooling layers are responsible for downsampling the feature map while retaining the relevant essential information . The smaller feature maps makes the models computational efficient and down sampling the reduce the risk of overfitting .There are three types of pooling method namely average pooling , max pooling and min pooling where average pooling takes average of all the values ,min pooling takes the minimum values in the particular matrix and max pooling takes max pooling .

Fig. 3. VGG -16 Architecture



The filter moving speed is controlled by a variable called Stride; the default value is 1 . The next two convolutional layers perform the same task as the first two layers and this is repeated till max pooling layer 4.The flatter layer is used to convert the 2D feature map into a 1D vector for sending for fully connected data . The last 3 layers are dense layers or fully connected layers. The first of the three dense layers is this one. Fully connected layers are another name for dense layers. Each neuron in this layer is linked to every neuron in the layer before. A linear operation is usually followed by a non-linear activation function. This layer frequently has a significant number of neurons, which enables it to detect intricate patterns in the data. This dense layer contains 4096 neurons here.This is the second layer of density, dense1 (Dense). It is totally connected, meaning that every neuron is linked to every other neuron in the preceding layer, much like the prior layer. These thick layers frequently minimize the number of dimensions in the data and extract prominent features. This dense layer in your architecture likewise has 4096 . The on final dense layer is finally used for classification task .The number of neurons in this layers is depended on number of classification classes here it is 0 to 9 so totally 10 classes 31 with of help of softmax activation functions it will produce the 10 probability value which number has the highest number of value will be the final output of the function .

Fig. 4. Formula for Relu and Softmax

$$RELU(x) = \begin{cases} 0 \text{ if } x < 0 \\ x \text{ if } x >= 0 \end{cases}$$

$$\text{softmax}(z_j) = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \text{ for } j = 1,\ldots,K$$

I have used Relu as an activation function . Relu provides solutions to vanishing gradient problems . Relu introduces non-linearity to the system. Also.Relu function is a less computational task which will also reduce the model training time but Relu also have some problems, mainly dying Relu problem where if value becomes negative the whole node will become dead .”I used the Adam optimizer during the Convolution Neural Network (CNN) training procedure for handwritten digit recognition. The selection of an optimizer is an important consideration when training deep neural networks. Adam, which stands for Adaptive Moment Estimation, is a well-liked option because of its effectiveness and capacity for a variety of learning rate plans. Adam optimizer is the combination of two types of optimizer namely “RMSprop ” and momentum. it maintains the adaptive learning rate for every parameter . The main advantages of ADAM optimizer is its property of changing learning rate which prevents the model to get stuck in the local minimum during training which is highly effective while using high dimensional data like images.

D) CRNN :

Fig. 5. CRNN Architecture

Model: "model"

Layer (type)	Output Shape	Param #
input (InputLayer)	[(None, 256, 64, 1)]	0
conv1 (Conv2D)	(None, 256, 64, 32)	320
batch_normalization (Batch Normalization)	(None, 256, 64, 32)	128
activation (Activation)	(None, 256, 64, 32)	0
max1 (MaxPooling2D)	(None, 128, 32, 32)	0
conv2 (Conv2D)	(None, 128, 32, 64)	18496
batch_normalization_1 (BatchNormalization)	(None, 128, 32, 64)	256
activation_1 (Activation)	(None, 128, 32, 64)	0
max2 (MaxPooling2D)	(None, 64, 16, 64)	0
dropout (Dropout)	(None, 64, 16, 64)	0
conv3 (Conv2D)	(None, 64, 16, 128)	73856
batch_normalization_2 (BatchNormalization)	(None, 64, 16, 128)	512
activation_2 (Activation)	(None, 64, 16, 128)	0
max3 (MaxPooling2D)	(None, 64, 8, 128)	0
dropout_1 (Dropout)	(None, 64, 8, 128)	0
reshape (Reshape)	(None, 64, 1024)	0
dense1 (Dense)	(None, 64, 64)	65600
lstm1 (Bidirectional)	(None, 64, 512)	657408
lstm2 (Bidirectional)	(None, 64, 512)	1574912
dense2 (Dense)	(None, 64, 30)	15390
softmax (Activation)	(None, 64, 30)	0

Convolution Recurrent Neural Network is the combination of convolution neural network and recurrent neural network where the the first part of the algorithm is same as convolution layers in CNN whereas instead if dense layers in the CNN model is replaced by a layer called LSTM (Long short Term memory) . LSTM is a type of recurrent neural network that are useful for sequence of data.The architecture of the CRNN models where the functionality of the convolution layers remains the same as CNN . Normalization is the process of converting the data to the particular range without disturbing the shape of the data . Batch normalization is a process to make the network fast and more stable by adding an extra normalization layer that does standardizing and normalizing operations on the input of the convolution layer .Normalization is the process of transforming the data to have a mean zero and standard deviation one First step in batch normalization is to calculate the mean for the values from that particular layer where m in the number of neutrons in the hi layer and calculate the standard deviation and for normalization the values will be subtracted from the mean and divided by the

standard deviation plus the smoothing term where the smoothing term is used for the stability of the number. The LSTM layer is a type of RNN layer that is mainly used in sequence data; LSTM can capture dependence over a long distance of input data . In Handwritten digit recognition LSTMs are used to understand the sequence of characters in a word or sentence . LSTMs function by keeping an updated state throughout time. In this condition, the LSTM is able to retain data from earlier inputs, which is critical for identifying long-term dependencies. Additionally, LSTMs feature gates that regulate how information enters and exits the state. These gates give the LSTM the ability to selectively forget or remember information, which is crucial for avoiding loops in the LSTM . We have 2 bidirectional LSTM layers after the dense layer . These layers are used to process the features extracted by the CNN layers and learn sequential patterns in the data . The bidirectional LSTM layers allow the LSTM to learn patterns in the data from both directions, which can improve the accuracy of the OCR system. Dropout layer is used for regularization techniques used in models to avoid the overfitting problems . Commonly the deep learning models are over-fitting prone so we have used drop out layers . When we add dropout layers in the network it will randomly select the subsets of neurons in that layer and set their output as Zero (0) . This randomly selecting probability is known as dropout rate where we have set it to 0.3 . The benefits of using the dropout layer is regularization ,reducing Co-Adaptation and creating an effective model.

E) Transformer-based Optical Character Recognition:

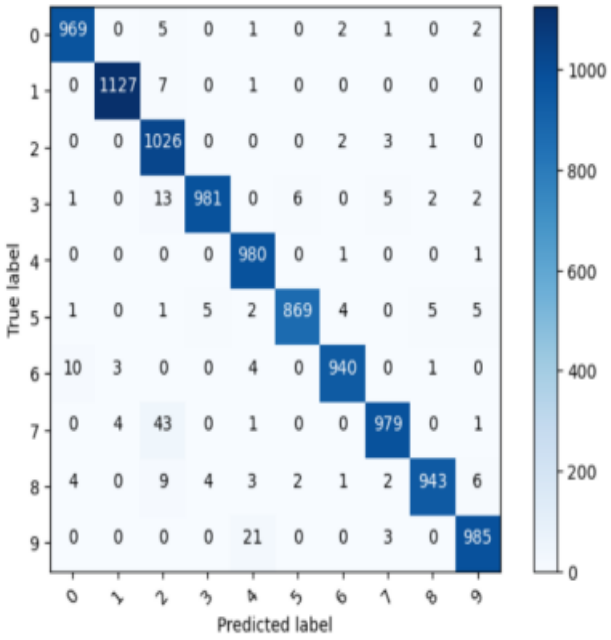
Minghao Li et al in 2022 has proposed a new Transformer based architecture for optical character recognition. The transformer model has 2 main component encoders and a decoder where encoder is used for converting the image into a feature vector and the vector is processed to the decoder where the feature vector is used for extracting the text from the image. The encoder is an image transformer and the decoder is a text transformer . The image is first resized to 384x384 pixels for model input.The TrOCR model is designed to work with images of this size, so resizing the image ensures that it will be compatible with the model. The image is then split into a sequence of 16 patches and the split images are sent simultaneously to the model. Position encoding is an essential component when applying the Transformer architecture to image data, just as it is for sequential data like text. Position encoding helps the model understand the spatial layout and relationships between different patches of the image.Each patch is passed through a Transformer encoder. The Transformer encoder is a neural network that takes an image patch as input and produces a sequence of feature vectors as output. The feature vectors represent the important features of the image patch, such as the shapes, colors, and textures of the text.After that, the feature vectors from all the patches are concatenated. This is carried out to produce a single feature vector sequence that represents the entire image. A stack of self-attention layers and feed-forward layers make up the neural network that serves as the Transformer decoder. The feed-forward layers let the decoder learn the relationships between the various portions of the sequence, while the self-attention layers enable the decoder to attend to different sections of the input sequence.The decoder starts by generating a blank token. Then, it iteratively generates the next token in the sequence, one at a time. At each step, the decoder takes the previously generated tokens as input, as well as the feature vectors from the encoder, and predicts the next token.The final text output is obtained by decoding the sequence of word piece tokens. The decoding process is performed by a language model, which is a neural network that can translate a sequence of tokens into a sequence of words.The final text output is obtained by decoding the sequence of word piece tokens. The decoding process is performed by a language model, which is a neural network that can translate a sequence of tokens into a sequence of words. Reference the above TROCR model idea is proposed by Minghao Li et al (2022).

4.RESULT AND DISCUSSION

The experiments are implemented on a regular personal computer with AMD ryzen 4000 series CPU, 8 GB RAM, and an AMD Radeon Graphics Card .

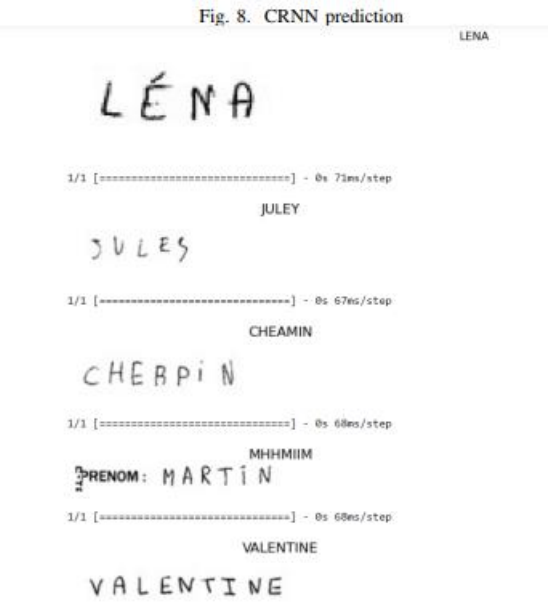
S.NO	MODEL	ACCURACY
1	CNN-VGG-16	97.99

Fig. 7. Confusion Matrix



The confusion matrix show the prediction of CNN architecture on MNIST test data which consists of 10000 images .We can see that the model didn't perform well on number 7 due to its pattern is similar with number 2 the error can be increased by increasing the number of Epoch and using techniques like data Augmentation to increase the images of 2 and 7.

Fig. 8. CRNN prediction



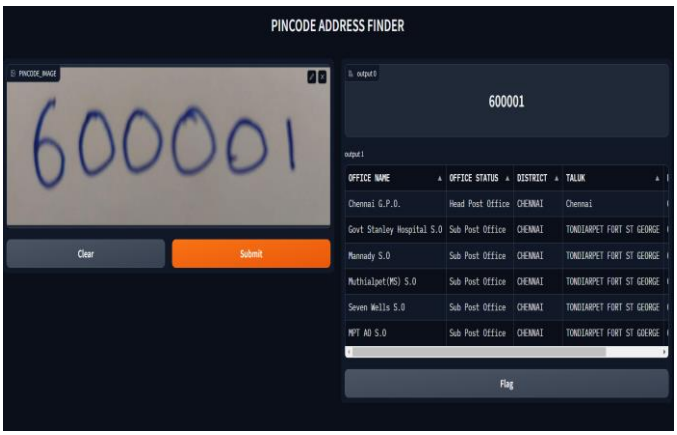
The above figure shows some prediction examples of CRNN on Character Recognition.

S.NO	IMAGE COUNT	CORRECT CHAR %	CORRECT WORD %
1	40,000	81.66%	64.80%
3	30,000	81.69%	64.70%
3	20,000	81.53%	64.69%
4	10,000	81.43%	64.92%

The model has performed well on identifying the single character but relatively low performance on finding the whole word together .For better performance the model architecture need to be improved and the number of training epochs need to be increased so the model can perform well on test data also.The CRNN model needs to be improved for using the same model in the real time OCR tasks where sometime when the character will not be the main focus in the image .The pre trained TROCR model performs well on Handwritten user input image compared to the CNN and CRNN models.

5. CONCLUSION AND FUTURE WORK

My comparison of the pre-trained Transformer based Optical Character Recognition Model outperformed both of my CNN and CRNN models . The OCR can be used in different domains like E-commerce , Postal service and baking domain etc,. The Transformer model Works better on Handwritten images given by the user at the live time compared to other models. The OCR can be used in different domains as mentioned earlier here is an example of its use in Postal Services for scanning the PIN code in the parcels



Future Work :

Increasing Accuracy : Increase the OCR model's accuracy by gathering more diverse training datasets, with a focus Fig. 9. Usage in Postal Service on handwritten text in a range of languages and styles. Look into cutting-edge pre-processing methods to improve image quality and lower noise for better recognition outcomes.

Recognizing layout and structure: Discover how to recognize document layouts and structures, such as headings, paragraphs, tables, and lists, which can be helpful for reading documents

Real-Time OCR: Develop real-time OCR capabilities for processing text from live video streams, making it suitable for applications like augmented reality and video conferencing

Multilingual OCR: This would allow OCR systems to recognize text in multiple languages. This would be useful for applications such as global document management and translation.

Intelligent OCR: This would allow OCR systems to learn and adapt to new document types and conditions. This would make OCR systems more robust and reliable.

Recognition of Handwriting : As this is a useful feature for many applications, continue to expand handwriting recognition capabilities, especially for cursive handwriting and different handwriting styles..

Integration of Natural Language Processing (NLP) : Use NLP approaches to combine sentiment analysis, named entity recognition, and document summarizing on text that has been recognized.

6. REFERENCE

- Minghao Li1,Tengchao Lv2,Jingye Chen2,Lei Cui2,Yijuan Lu2,Dinei Florencio2,Cha Zhang2, Zhoujun Li1,Furu Wei2 Title of article : TrOCR: Transformer Based Optical Character Recognition with Pre-trained Models https://arxiv.org/abs/2109.10282
- Atienza, R. 2021. Vision Transformer for Fast and Efficient Scene Text Recognition. arXiv preprint arXiv:2105.08582. Baek, J.; Kim, G.; Lee, J.; Park, S.; Han, D.; Yun, S.; Oh, S. J.; and Lee, H. 2019. What is wrong with scene text recognition model comparisons? dataset and model analysis. In Proceedings of the IEEE/CVF International Conference on Computer Vision, 4715–4723.
- Ahmed . S . S , Mehmood . Z , Awan . A . I , Yousaf . M . R (2023) . Title of article:A Novel Technique for Handwritten Digit Recognition Using Deep Learning . Hindawi Journal of Sensors Volume 2023, Article ID 2753941,15 pages https://doi.org/10.1155/2023/2753941
- Dilhiwala R. T , Patel . P , Patal . B , Mehta . H , Panchal . N . (2023) . Title of article: Handwritten Digit Recognition Using CNN . International Journal of Advances in Engineering and Management (IJAEM) Volume 5, Issue 6 June 2023, pp: 969-972 www.ijaem.net ISSN: 2395-5252
- Swetha K . Hithaishi . Y , Tejaswini .N.L , Parathasarathi . P , Venkateswara Rao . P. V (2021) . Title of article:HANDWRITTEN DIGIT RECOGNITION USING OPENCV AND CNN. International Journal of Creative Research thoughts (IJCRT) ,— Volume 9, Issue 6 June 2021 — ISSN: 2320-2882
- Siddique . F , Sakib . S , Md . Abu Bakr Siddique (2020) . Title of article: Recognition of Handwritten Digit using Convolutional Neural Network in Python with Tensorflow and Comparison of Performance for Various Hidden Layers . Publisher: IEEE DOI: 10.1109/ICAEE48663.2019.8975496
- Chandra . R . P , Karthik .S.J , Tharun . G , Bhargav . A , Rakesh . A , Makesh .U. G (2022), Title of article: HIGH ACCURACY HANDWRITTEN DIGIT RECOGNITION USING DEEP CONVOLUTIONAL NEURAL NETWORK ARCHITECTURE . International Journal For Advanced Research In Science and technology , Volume 12, Issue 11, Nov 2022 ISSN 2457-0362 Page 223 - 228
- Anchit Shrivastava, Isha Jaggi, Sheifali Gupta, Deepali Gupta, "Handwritten Digit Recognition Using Machine Learning: A Review", 2019 2nd International Conference on Power Energy, Environment and

IntelligentControl(PEEIC),978-1-7281-1793-5/19/$31.00 ©2019 IEEE

- Albahli . S , Mawaz . M , Javed .A , Irtaza . A (2021) , Title of article : An improved faster-RCNN model for handwritten character recognition ,Research ArticleComputer Engineering and Computer Science Published: 30 March 2021, DOI: 10.1007/s13369-021-05471-4.