



e-ISSN: 2319-8753 | p-ISSN: 2347-6710

# IJIRSET

International Journal of Innovative Research in  
**SCIENCE | ENGINEERING | TECHNOLOGY**

# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 10, Issue 9, September 2021

**ISSN** INTERNATIONAL  
STANDARD  
SERIAL  
NUMBER  
INDIA

Impact Factor: 7.569

9940 572 462

6381 907 438

[ijirset@gmail.com](mailto:ijirset@gmail.com)

[www.ijirset.com](http://www.ijirset.com)



# Hybrid Optical Character Recognition for Mobile Device & WebApp with ANN Model

**B Likhitha Madhuri, DR.S.Adinarayana M.E.,PH.D.**

M.Tech Scholar, Department of Computer Science and Engineering, Raghu Institute of Technology, Visakhapatnam, AP, India

Professor, Department of Computer Science and Engineering, Raghu Institute of Technology, Visakhapatnam, AP, India

**ABSTRACT:** The objective of this work is to convert printed text recorded offline using either scanning equipment or cameras into a machine-usable text by stimulating a neural network so that it would improve the process of collecting and storing data by human workers. Noise has been considered as one of the major issues that degrades the performance of character recognition system. Common optical character recognition tasks involve identifying simple edge detection and matching them with predefined patterns. In this research, characters are recognized even when noise such as inclination and skewness present, by training the network to look for discrepancies in data. The entire recognition system is divided into two sections such as training and recognition section. Both sections include image acquisition, pre-processing and feature extraction. Preprocessing involves digitization, noise removal, binarization, line segmentation and character extraction. Images are also masked in multiple ways and processed individually to increase the confidence level of prediction.

## I. INTRODUCTION

**Machine Learning** is the field of study that gives computers the capability to learn without being explicitly programmed. It's one of the most exciting technologies that one would have ever come across. As it is evident from the name, it gives the computer that which makes it more similar to humans: The ability to learn.

Machine learning is actively being used today, perhaps in many more places than one would expect. Machine learning algorithms are used in a wide variety of applications, such as email filtering and computer vision where it is infeasible to develop an algorithm of specific instructions for performing the task. Machine learning is closely related to computational statistics which focuses on making predictions using computers. The study of mathematical equalization delivers methods, theory and application domains to the field of machine learning. Its application across business problems, machine learning is also referred to as predictive analytics. We probably use a learning algorithm dozens of times without even knowing it.

A mechanism for learning - if a machine can learn from input then it does the hard work for us. This is where Machine Learning comes in action. The algorithms in ML allow software applications to become more accurate in predicting outcomes without being explicitly programmed. The basic premise of machine learning is to build algorithms that can receive input data and use statistical analysis to predict an output while updating outputs as new data becomes available.

### 1.1. SUPERVISED LEARNING

These algorithms are trained using labelled examples, in different scenarios, as an input where the desired outcome is already known. An equipment, for instance, could have data points such as "F" and "R" where "F" represents "failed" and "R" represents "runs".

Advancement in deep learning frameworks, several pre-trained models such as VGG\_Face, VGG16, VGG19, MT-CNN, Face-net etc., is used to extract the face embeddings, which can be later trained with different machine learning and deep learning networks.

Hence in this paper, we have undergone different case studies and perceived diverse real time problems such as false detection, motion and Gaussian blurriness in faces, illumination variation etc., and designed a real time framework in which we are extracting the face with our custom universal face detector and solved most of the above problems using a hybrid preprocessing network and extracted the face embeddings using our own Deep learning frameworks and classified those embeddings using different machine learning and deep learning frameworks in which KNN and SVM gives best solution in classification for real time problems. In this paper, the below section represents previous case



studies of different researchers which is followed by our proposed architecture with mathematical equations and results in the A supervised machine learning model is given instances of data specific to a problem domain and an answer that solves the problem for each instance. When learning is complete, the model is able not only to provide answers to the data it has learned on, but also to yet unseen data with high precision. Neural networks are learning models used in machine learning. Their aim is to simulate the learning process that occurs in an animal or human neural system. Being one of the most powerful learning models, they are useful in automation of tasks where the decision of a human being takes too long, or is imprecise.

### 1.1.2 UNSUPERVISED LEARNING

This method of ML finds its application in areas where data has no historical labels. Here, the system will not be provided with the "right answer" and the algorithm should identify what is being shown. The main aim here is to analyze the data and identify a pattern and structure within the available data set. Transactional data serves as a good source of data set for unsupervised learning.

For instance, this type of learning identifies customer segments with similar attributes and then lets the business to treat them similarly in marketing campaigns. Similarly, it can also identify attributes that differentiate customer segments from one another. Either ways, it is about identifying a similar structure in the available data set. Besides, these algorithms can also identify outliers in the available data sets. Some of the widely used techniques of unsupervised learning are -

- k-means clustering
- self-organizing maps
- value decomposition
- mapping of nearest neighbour

### 1.1.3 SEMI-SUPERVISED LEARNING

This kind of learning is used and applied to the same kind of scenarios where supervised learning is applicable. However, one must note that this technique uses both unlabelled and labelled data for training. Ideally, a small set of labelled data, along with a large volume of unlabelled data is used, as it takes less time, money and efforts to acquire unlabelled data. This type of machine learning is often used with methods, such as regression, classification and prediction. Companies that usually find it challenging to meet the high costs associated with labelled training process opt for semi-supervised learning.

### 1.1.4 REINFORCEMENT LEARNING

This is mainly used in navigation, robotics and gaming. Actions that yield the best rewards are identified by algorithms that use trial and error methods. There are three major components in reinforcement learning, namely, the agent, the actions and the environment. The agent in this case is the decision maker, the actions are what an agent does, and the environment is anything that an agent interacts with. The main aim in this kind of learning is to select the actions that maximize the reward, within a specified time. By following a good policy, the agent can achieve the goal faster.

Hence, the primary idea of reinforcement learning is to identify the best policy or the method that helps businesses in achieving the goals faster. While humans can create a few good models in a week, machine learning is capable of developing thousands of such models in a week.

following Sections which is followed by references.

## II. METHODOLOGY

### 2.1 PRE-PROCESSING

In pre-processing phase, there is a sequence of operations that are performed on the scanned image. It improves the image rendering making it suitable for segmentation of grayscale character image into a window sized. The scanned image may have some noise due to some unnecessary details present in the image. After noise reduction, the image is then saved for later use. All the pre-designed templates of the alphabets are loaded into the system. One of the techniques used for pre-processing is thresholding that converts the image into binary format based on some threshold value. That threshold value can be set at local or global level. Different types of filters such as averaging, max and min filters can be used. One more important part of pre-processing includes finding out the skew in the document. Different techniques used for skew estimation are: projection profiles, Hough transform, nearest neighbourhood methods etc. In some cases, thinning of the image is also performed before later phases are applied. Finally, the text lines present in the document can also be found out as based on clustering or projections of the pixels.



Block diagram of ocr with ANN

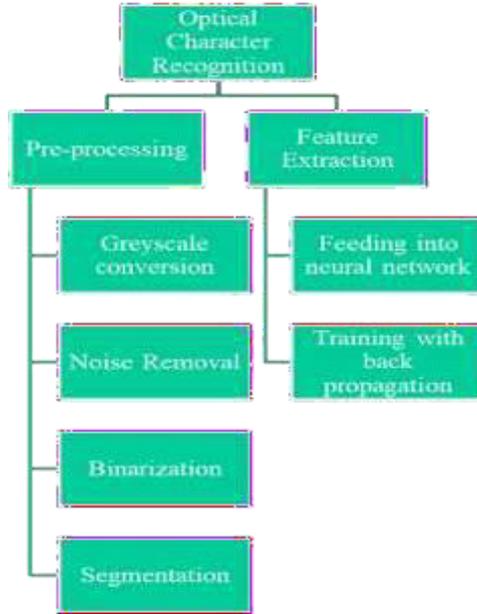


fig.1. Testing phase of OCR System

**2.2.GRAYSCALE CONVERSION**

Grayscale images have many shades of gray. Grayscale images is result of measuring intensity of each pixel. For achieving accuracy input, document should be grey scaled. If each colour pixel is described by a triple (R, G, B) of intensities for red, green, and blue. TheGIMP image software has three algorithms.

The lightness method averages the most prominent and least prominent colours:

$$(\max(R, G, B) + \min(R, G, B)) / 2.$$

The average method simply averages the values:  $(R + G + B) / 3$ .

The luminosity method is a more sophisticated version of the average method. It also averages the values, but it forms a weighted average to account for human perception. We're more sensitive to green than other colours, so green is weighted most heavily.

The formula for luminosity is:

$$Y = 0.2126R+0.7152G+0.0722B$$

**2.2 SKEW ANGLE CORRECTION**

The detection of the skew angle of a document is a common pre-processing step in document analysis. The document might have been slightly rotated during its digitalization.(fig 3 and 4) It is therefore necessary to compute the skew angle and to rotate the text before going further in the processing pipeline (i.e. words/letters segmentation and later OCR). We will work on 4 different images of the same short (but famous) text.

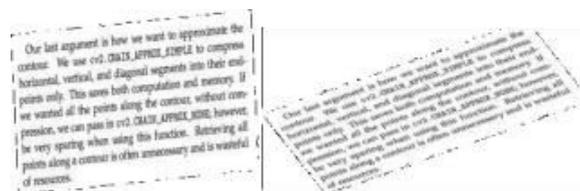
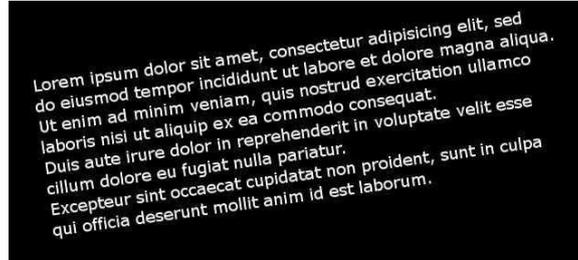


figure2:skew correction Document



We assume here that the noise introduced by digitalization has been removed by a previous preprocessing stage. We also assume that the text has been isolated: no images, horizontal or vertical separators, etc. In image processing, objects are white and the background is black, we have the opposite, we need to invert the colours of our image. And the result looks like this



### 2.3.BOUNDING BOX

The minimum bounding box of a set of 2D points set is the smallest rectangle (i.e. with smallest area) that contains all the points of the set. The points of our 2D set are all the white pixels forming the different letters.

If we can compute the bounding box of this set, it will be possible to compute the skew angle of our document. Given a bounding box of this set, it will be possible to compute the skew angle of our document. Given a bounding box of our set, it will also be easy to extract our text from the image and rotate it. Our text is small but we have a large number of points, indeed the resolution of our image is large, we have many pixels per letter. We have several possibilities here: We can down strangle the image in order to reduce its resolution, we can use mathematical morphology (i.e. erosion), etc. There are certainly other solutions, you will have to choose one depending on what are the previous or next stages in your processing pipeline.

Here, we used a 5x3 rectangle shaped structuring element, that is to say we want to keep pixels lies in a region of white pixels of height 3 and width 5.

The result on the previous image (fig 6):



fig 3: Result on the previous image

This erosion was really aggressive, we removed most of the pixels and in fact only some letters “survived” the operation. Calculating the bounding box will be really fast but we may have stripped too much information, it might cause problems on certain images. However, as we will see, there are still enough pixels to get decent results. And the result is(fig 7)

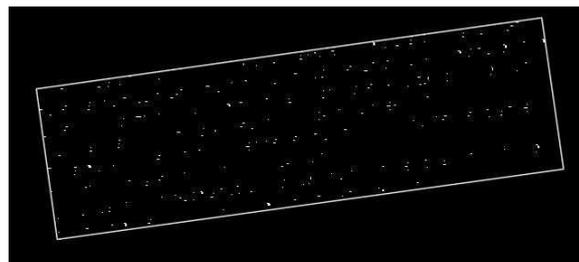


Fig 4: Result after bounding box

We can now use the OpenCV function in order to compute the minimal bounding box, this function is called `minAreaRect`, That's it, we have our minimal bounding box! We can now have access to the angle: there are several ways to compute the rotation angle.

In our case, if the angle is less than  $-45$  degrees, the angle was computed using a "Vertical" rectangle, we must therefore correct the angle by adding 90 degrees. Note that the bounding box obtained is not the bounding box of our whole text. The erosion removed a lot of information, if we try to match the bounding box on the initial text, we will lose for example the upper part of letters with a larger height than other letters (eg: f, t, h, d, l...). Compute the bounding box on the unprocessed text (or use a smaller structuring element) if you want the bounding box of the whole text.(fig 8)

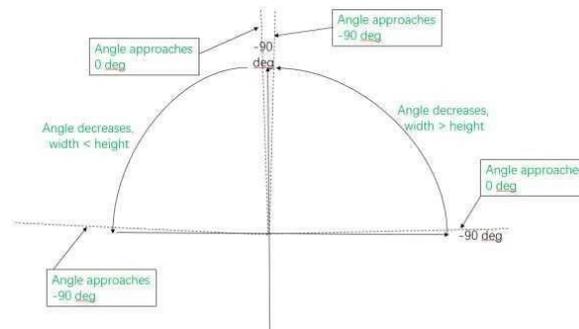


fig5: Process of Computation of Bounding box on unprocessed text

## 2.4.NOISE REDUCTION

Scanned documents often contain noise that arises due to printer, scanner, print quality, age of the document, etc. Therefore, it is necessary to filter this noise before we process the image. Commonly used approach is to process the image through a low-pass filter and use it for later processing. The objective in the design of a noise- filter is that it should remove as much of the noise as possible while retaining the entire signal. Noise reduction improves the quality of the document. Image noise is random vibration of brightness or colour information in images and is usually an aspect of electronic noise. It can refer to the electronic signal corresponding to acoustic noise (in an audio system) or the electronic signal corresponding to the (visual) noise commonly seen as 'snow' on a degraded television or video image. Image processing allows a much wider range of algorithms to be applied to the input data and can avoid problems such as build-up of noise & signal distortion during processing of images. Wavelet transforms have become a very powerful tool for denoising an image. One of most popular method is Wienerfilter in this 4 types of noise used (Gaussian noise, salt & pepper noise, speckle noise & Poisson noise) is used and the image denoising performed for different noise by mean filter, median filter & Wiener filter.

### 2.4.1MEAN FILTER

Mean filtering is a simple intuitive (easy to use & understand) and easy to implement method of smoothing images i.e, Reducing the amount of intensity variation between one pixel and next one. it is often used to reduce noise in images. The idea of mean filtering is simply to replace each pixel value in an image with mean value of its neighbours & itself. The problem with mean filtering, it is single pixel with a very unrepresentative value can significantly affect the mean value of all the pixels in its neighbourhood.

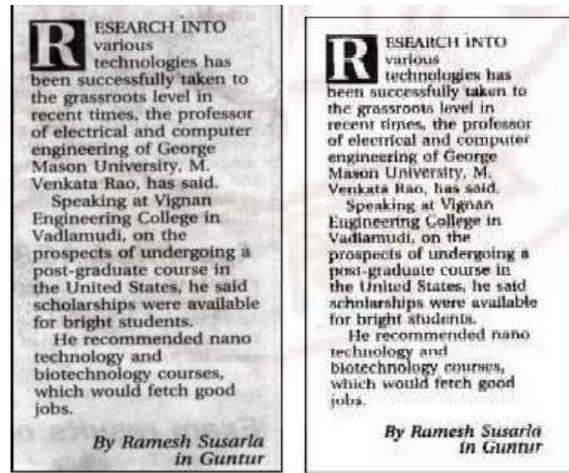
### 2.4.2.MEDIAN FILTER

Median filter is a non-linear digital filtering technique. It is used to remove noise from an image or signal. It preserves edges while removing noise which is better than mean filter instead of taking mean of neighbouring pixel values, it takes median of those values median is calculated by first sorting all the pixel values from the surrounding neighbourhood into numerical order & then replacing the pixel being considered with the middle pixel value. Median filter is more robust than the mean and so a single very unrepresentative pixel in the neighbourhood will not affect the median value. Using spatial median filter and mean filter we can reconstruct the noise image by using mean filter total noise not reduced but median filters using total noise reduced. A median filter is more effective than

when the goal is to simultaneously reduce noise & preserve edges. Median filters are particularly effective in the presence of impulse noise also called " salt & pepper noise " because of its appearance as white and black dots superimposed on an image. For every pixel, a  $3 \times 3$  neighbourhood with the pixel as centre is considered. In median filtering, the value of the pixel is replaced by the Median of the pixel values in the  $3 \times 3$  neighbourhood.

### 2.5.BINARIZATION

A printed document is first scanned and is converted into a grey image. Binarization is a technique by which the grey scale images are converted to binary images. In an image analysis or enhancement problem, it is very essential to identify the objects of interest from the rest. In a document image this usually involves separating the pixels forming the printed text or diagrams (foreground) from the pixels representing the blank paper (background). The goal is to remove only the background, by setting it to white, and leave the foreground image unchanged. Thus, binarization separates the foreground and background information. This separation of text from background is a prerequisite to subsequent operations such as segmentation and labelling. The most common method for binarization is to select a proper intensity threshold for the image and then convert all the intensity values above the threshold to one intensity value, and to convert all intensity values below the threshold to the other chosen intensity. In order to deal with this, the conversion of a grayscale image into binary image (fig 12) where a pixel can have any of the two values (0 or 1). This process works on the basis of threshold value. The threshold value is calculated by taking the average of all pixel intensities.



(a) Gray image

(b) Binary image

### 2.6.SEGMENTATION:

Pattern based recognition requires matching of generated binary format with the existing template for this purpose the binary has been divided into 5 tracks and each track subdivided into 8 sectors.

This can be done using following procedure

1. Identify centre of matrix.
2. Calculate radius say  $rad$  by finding pixel with maximum distance from center using distance formulae.

$$Dist = ((2-1) + (2-1))$$

3. Perform  $(rad \div 5)$  to identify size of each imaginary track.
4. Identify imaginary sectors.
5. Generate track- sector matrix by calculating number of 1's in each intersection of sector and track.

### 2.6.FEATURE EXTRACTION

Feature extraction is the process of getting information about an object or a group of objects in order to facilitate classification. This is an important part in our system.

Now each character is to be identified for the row obtained earlier. This is done by scanning the row vertically from top to bottom, (fig 9) the first darker pixel detected is the leftmost (left) pixel of character. (fig 10) Now if all pixels are found to be blank then this is right of character.

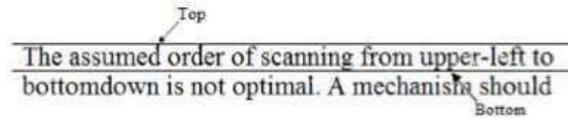


fig 5: Scanning the row vertically from top to bottom

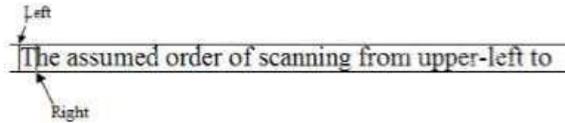


fig 6: Darker pixel detected is leftmost

III.NEURAL NETWORK ARCHITECTURE

Tesseract 4.00 includes a new neural network subsystem configured as a text line recognizer. It has its origins in OCR Python-based LSTM implementation, but has been totally redesigned for Tesseract in C++. The neural network system in Tesseract pre-dates Tensor Flow, but is compatible with it, as there is a network description language called Variable Graph Specification Language (VGSL), that is also available for Tensor Flow.

The idea of VGSL is that it is possible to build a neural network and train it without having to learn a lot of anything. There is no need to learn Python, Tensor Flow, or even write any C++ code. It is merely required to understand the VGSL specification language well enough to build syntactically correct network descriptions. Some basic knowledge of what the various neural network layer types are and how they are combined will go a very long way. Tesseract 4.00 includes a new neural network

3.1.BASICS OF IMPLEMENTATION OF NETWORK:

The accuracy of an LSTM network is heavily dependent on the training data.(fig 14) The training data used in the new Tesseract LSTM included a significant amount of degraded images produced by cameras. If Tesseract's LSTM recognizer fails on a particular character sequence, it can "fall-back" to its generic static shape classifier to make the determination. The amount of computation required for LSTM network character recognition is about 50 times greater than for character recognition performed using a static classifier. To help speed up the processing, we are utilizing for the inner neural network calculations. We have also achieved a significant performance increase by making

All network layer types are derived from the Network base-class. The Plumbing sub-class is a baseclass for layers that manipulate other layers in some way, e.g. by reshaping their input/output or organising a group of layers. The input/output data "Tensor" is Network IO and the weights are stored in a Weight Matrix, both of which contain a Tesseract GENERIC\_2D\_ARRAY to hold the data. LSTM Recognizer provides the higher-level abstraction of converting an image of a text line to a sequence of tesseract WERD\_RES classes. LSTM Trainer likewise handles the abstraction of training a network on an image of a text line that has a UTF-8 string 'truth'. Network Builder takes responsibility for converting the VGSL specification language to a graph of network elements.

3.1.2.LSTM ARCHITECTURE:

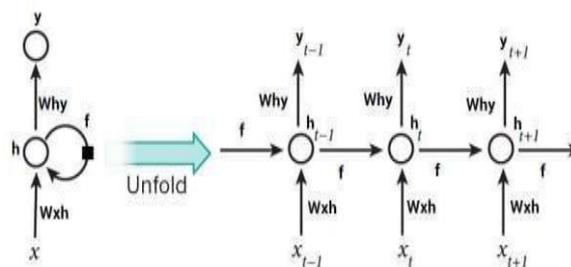
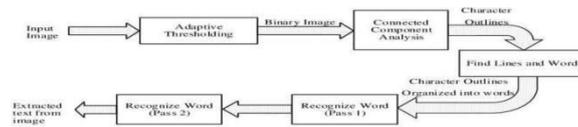


Fig:7 LSTM ARCHITECTURE



**3.1.3. TRAINING THE NEURAL NETWORK WITH BACKPROPAGATION:**

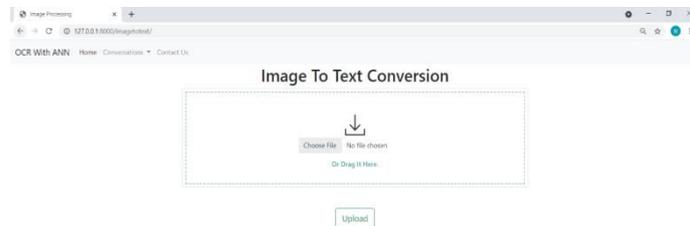
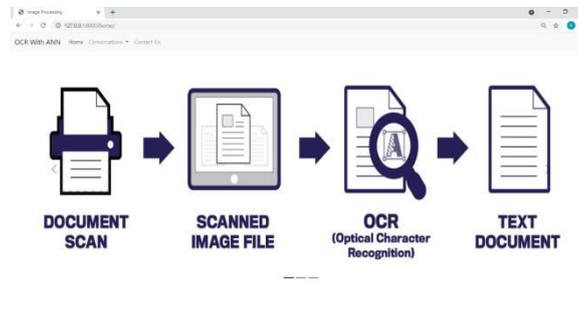
A neural network model is described by a string that describes the input spec, the output spec and the layers spec in between. Example: [1,0,0,3 Ct5,5,16 Mp3,3 Lfys64 Lfx128 Lrx128 Lfx256 O1c105]. The first 4 numbers specify the size and type of the input, and follow the TensorFlow convention for an image tensor: [batch, height, width, depth]. Batch is currently ignored, but eventually may be used to indicate a training mini-batch size. Height and/or width may be zero, allowing them to be variable. A non-zero value for height and/or width means that all input images are expected to be of that size, and will be bent to fit if needed. Depth needs to be 1 for greyscale and 3 for colour. As a special case, a different value of depth, and a height of 1 causes the image to be treated

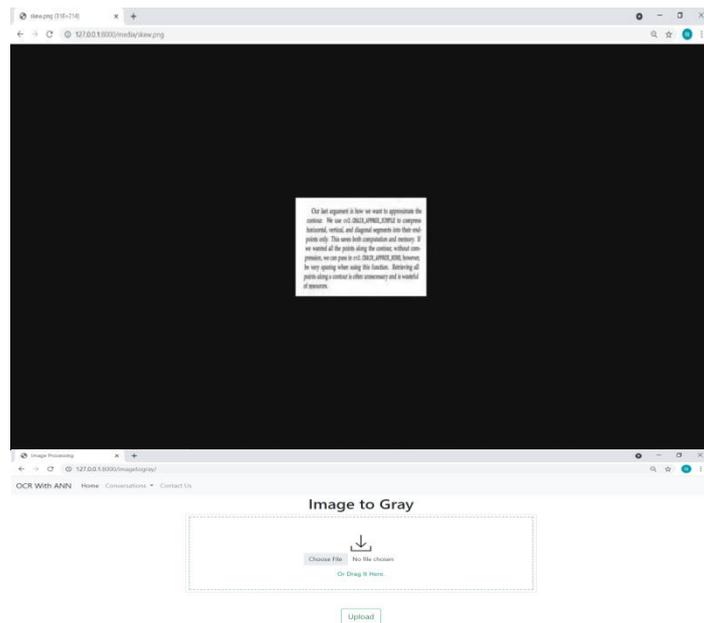
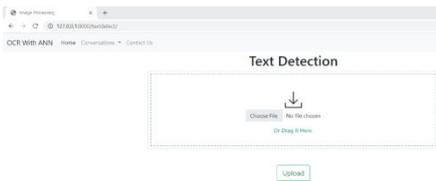
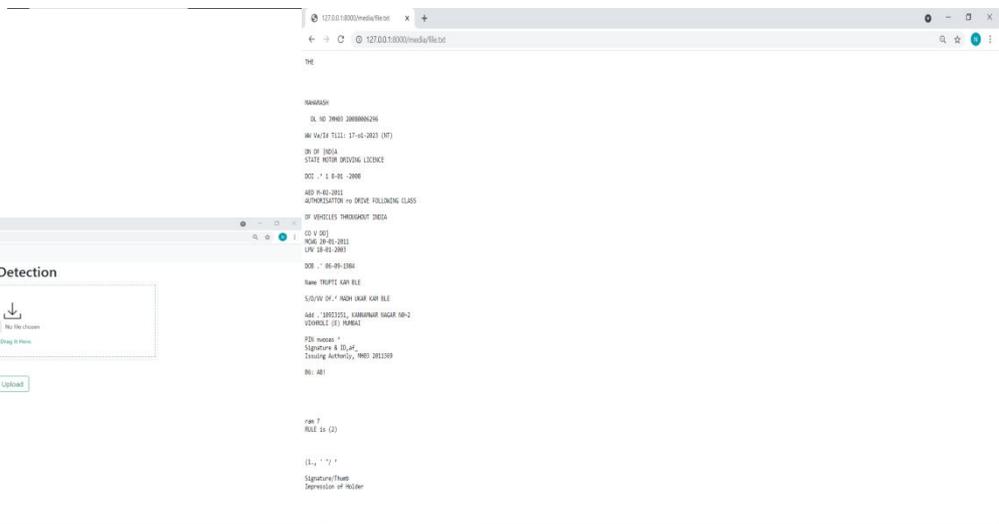
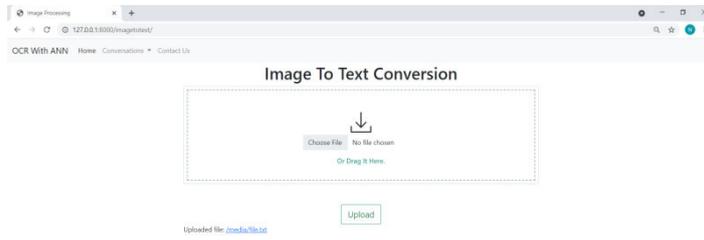


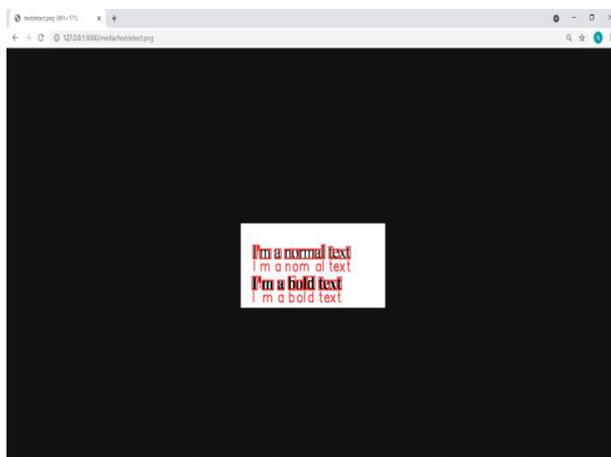
**Fig 15: ANN for identification of characters**

from input as a sequence of vertical pixel strips. Note that throughout, x and y are reversed from conventional mathematics, to use the same convention as TensorFlow. The reason TF adopts this convention is to eliminate the need to transpose images on input, since adjacent memory locations in images increase x and then y, while adjacent memory locations in tensors in TF, and Network IO in tesseract increase the rightmost index first, then the next-left and so-on, like C arrays.

**IV.OUTPUTS**







## V. FUTURE WORK

OCR supports a large number of useful applications. In the early days, OCR has been used for sorting the mails, reading bank cheques and verifying signatures. Further, OCR has been used by various for automated form processing in places where an enormous data is available in printed form. Another valuable application of OCR is in helping blind and visually impaired people to read text. Other uses of OCR include passport validation, processing utility bills, pen computing and automated number plate recognition.

OCR of complex documents intermixing with text, graphics, tables and mathematical symbols, unconstrained handwritten characters, color documents, low-quality noisy documents, etc. Among the commercial products, postal address readers, and reading aids for the blind are available in the market. Nowadays, there is much motivation to provide computerized document analysis systems. OCR contributes to this progress by providing techniques to convert large volumes of data automatically. A large number of papers and patents advertise recognition rates as high as 99.99%; this gives the impression that automation problems seem to have been solved.

Failure of some real applications show that performance problems still exist on composite and degraded documents (i.e., noisy characters, tilt, mixing of fonts, etc.) and that there is still room for progress. Various methods have been proposed to increase the accuracy of optical character recognizers. In fact, at various research laboratories, the challenge is to develop robust methods that remove as much as possible the typographical and noise restrictions while maintaining rates similar to those provided by limited-font commercial machines. Thus, current active research areas in OCR include handwriting recognition, and also the printed typewritten version of non-Roman scripts (especially those with a very large number of characters).

## VI. CONCLUSION

In this project, we have proposed an artificial neural network based character recognition system for recognition of English alphanumeric characters with special symbols even with different font styles. Our system produces very good results for individual classes but produces satisfactory results when the classes are combined together. Optical characters can be recognized in multiple ways, however using something that mimics the nature and characteristic of human brain is bound to succeed better than any other system. The proposed method uses neural network approach for the task of optical character recognition.

## REFERENCES

1. Shyla Afroge, Boshir Ahmed, Firoz Mahmud, "Optical Character Recognition using Back Propagation Neural Network".
2. T. K. Das, Asis Kumar Tripathy and Alekha Kumar Mishra, "Optical Character Recognition using Artificial Neural Network".
3. Rajesh Kumar Subudhi, 2 Bibhuprasad Sahu, 3 Pratyush Rn. Mohapatra, "A Novel Noise Reduction Method For OCR System".



4. Youlian Zhu, Cheng Huang, "An Improved Median Filtering Algorithm for Image Noise Reduction".
5. Colby McKibbin "Optical Character Recognition Using Artificial Neural Networks".
6. Faisal Mohammad, Jyoti Anarase, Milan Shingote, Pratik Ghanwat "Optical Character Recognition Implementation Using Pattern Matching"
7. Puneet Naresh ,Kumar Garg "Binarization Techniques used for Grey Scale Images".
8. Pranob K Charles , V.Harish , M.Swathi , CH. Deepthi "A Review on the Various Techniques used for Optical Character Recognition".
9. Ray Smith "Hybrid Page Layout Analysis via Tab-Stop Detection"
10. Y. Zhao, —On-line neural network learning algorithm with exponential convergence rate, | Electron. Lett., vol. 32, no. 15, pp. 1381–1382, July 1996.
11. D. E. Rumelhart, B.Widrow, and M. A. Lehr, —The basic ideas in neural networks, | Commun. ACM, vol. 37, pp. 87–92, 1994.
12. R. P. Lippmann, —An introduction to computing with neural networks, | IEEE Acoust. Speech Signal Process. Mag., vol. 4, no. 2, pp. 4–22, Apr. 1987.
13. I. Qadri, M.T., & Asif, M, 2009, Automatic Number Plate Recognition System for Vehicle Identification Using Optical Character Recognition presented at International Conference on Education Technology and Computer, Singapore, 2009. Singapore: IEEE.
14. Guohong Liu, Wenming Guo, "Application of improved arithmetic of median filtering denoising, Computer Engineering and Applications," 2010, vol.46, no.10, pp.187-189
15. Xiaokai Wang, Feng Li, "Improved adaptive median filtering," Computer Engineering and Applications, 2010, vol.46, no. 3, pp. 175-17.



**Impact Factor:  
7.569**



# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 9940 572 462  6381 907 438  [ijirset@gmail.com](mailto:ijirset@gmail.com)



[www.ijirset.com](http://www.ijirset.com)

Scan to save the contact details