BPBR: Serial Number Recognition from Bengali Prize Bond

***Abstract*—Recognition of prize bond serial numbers offers something new to the field of Image Processing and Natural Language Processing (NLP). In this paper, we discussed the procedures of detecting Bengali serial number of prize bonds from image using NLP where several steps of processing of the image data is also required before the recognition process. We went through several steps processing of the images to bring them into the appropriate format for the recognition procedure. We also discussed further possibilities of practical implementation of this concept which will include the idea of automation in a minimal way in our regular life.**

***Index Terms*—Image Processing, Natural Language Processing, Automation**

I. Introduction

The applications of Natural Language Processing are increasing day by day in this modern era of technology. The definition of NLP can be given from many aspects, but the most generalized definition is given by Lutkevich B. [1]. He [1] stated that "Natural language processing (NLP) is the ability of a computer program to understand human language as it is spoken and written is referred to as natural language". It can be applied on different types of data such as voice, image, text etc. It has been used in many different areas such as email filtering, auto-correct feature, license plate detection or even in smart assistants like Siri and Alexa. NLP has been used for general purpose, research purpose or even in the field of automation. The idea of serial number detection from prize bonds comes from the aspect of automation. The images need to be pre-processed at the beginning. At first the four point transformation is needed to detect and filter out every unnecessary part of the image and keep the prize bond's information only. After that the images need to be rotated to bring each and every image in the same position so that every images share same coordinates for same regions. Then the quality of each image is improved by sharpening the images further. After these steps, the images are required to be converted into grayscale images. This process ensures that the colors which are not of interest, does not come into the act of hampering the recognition process. Following that, unnecessary noises are removed from the images. After that only the region of interest (ROI) in the images are identified and selected which are required for the next stage to operate smoothly. After these processing steps, finally the recognition procedure may begin after all the processing steps. That is why our research work incorporates the use of both applications of image processing and natural language processing.

Digit recognition using NLP has been studied in a variety of contexts and applications during the last few years. The types of character recognition systems that have evolved as a consequence of diverse research investigations are classified and elaborated. Character recognition has been one of the main topics for the past 30 years for the researchers. It has been using for recognizing the handwritten zip codes, cheque processing and many more. In the mentioned paper they compared the performances of neural networks with the k-Nearest Neighbors classifier [2].In some other cases the neural network is trained and tested using the MNIST collection of handwritten digits. They also proposed Principal Component Analysis (PCA) for feature extraction, which can increase the neural network's performance and drastically reduce training time [3]. As for recognition of handwritten digits, a character recognition technique using convolutional neural network is suggested [4]. As the digital number designs on several objects is not the same, so the method of detecting them also gets to change every now and then and those specific methods needs to develop with technology too. There has been a huge difference between the online and offline recognition of the numbers. In online they are arranged and there is no chance to manage for the movement that needs to be handled very carefully. Different works regarding serial number detection from bank notes [5] [6] was done before and many ideas were proposed regarding its practical implementation in the real world. The paper by Ji Qian, Dongping Qian and Mengjie Zhang [6] described the use of image processing for the recognition of the serial number of the Chinese currency banknotes. The image processing techniques helped them to remove the noise from the picture by mean filtration technique i.e. by reducing the intensity variation between one pixel and the next one [6]. The other techniques that is followed by is Optical Character Recognition (OCR), is a process by which we convert printed document or scanned page to ASCII character that a computer can recognize. It can also decrease the degree of human error. OCR system approaches several system for the several page structures that they would like to work on [7]. OCR can have various approaches for it's system too, Matrix matching, Fuzzy Logic, Feature Extraction, Structural Analysis and Neural Network. According to the survey of OCR applications

banking industry is one the most that uses this technology as human interaction in the core is very low. In this age of verification, we have to verify the program that we use the most in day to day life and that is CAPTCHA. As CAPTCHA always contains that distracting backgrounds and noises in the pictures also not highlighted, it can be removed using OCR technique [8]. For the banknotes, we have come to know about two processes, the Skew correction and orientation identification. Then detected the text region be in a binarized by a combined threshold technique. In recent years, the binarized approach has shown to be quite effective in extracting a large number of documents. This may be done using an Area-Ratio-Based Binarization technique, which creates a binary map using the area ratio method. Then there's Binarization based on Block Contrast. The proposed block contrast based threshold approach takes use of the stroke width information, eliminates interference such circles, wrinkles, and smears, and tends to provide a binary image without accounting for pixel intensity. Because the area-ratio technique ignores the stroke texture characteristic and the block contrast based method ignores pixel intensity, the combination approach can fix a few erroneous foreground pixels while also giving a more comprehensive solution [5].

Detecting serial number from prize bonds is similar to detecting serial number from bank notes in terms of concepts but much more different in terms of implementation. Automation decreases redundant work from daily life even if we use it in a smaller margin or for basic tasks. Automation is the key to revolutionize the industrialization where there will be lesser margin of errors. The idea of detecting serial numbers from prize bonds may sound ordinary but the idea behind its practical implementation unlocks a new scope of automation in our daily life.

II. Methodology

A. Dataset

The dataset for our work includes over two hundred images of different prize bonds. In Bangladesh, only Bangladesh Bank issues prize bonds and they are fully written in Bengali language including the serial numbers as well. That is why we needed a hefty amount of images to begin our work. Over five hundred images were captured using different mobile phones to bring variety in the qualities of the image files. From there we tried to detect the serial numbers. But then we faced some constraints during the image processing segment such as pictures containing fingers, pictures not having 4 corners pointed out. In order to fix the complexities issued by the constraints, we decided to filter out the 500 images based on them.

In **Figure** 1(a), we can see that the four corners of the prize bond are missing thereby leading to difficulties during the image processing segment. Similarly in **Figure** 1(b), it is clearly visible that the prize bond image

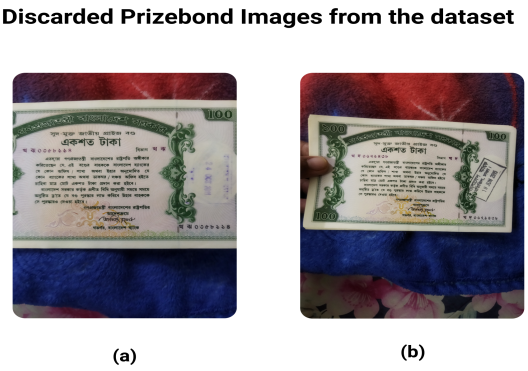

Fig. 1. Discarded prize bond images from the initial database (a) Without four corners of the prize bond (b) With a finger on the prize bond

contains a finger, which poses some difficulties during the identification. After taking these issues into consideration, we filtered our database which resulted in images as shown in **Figure** 2. We ended up with 200 images that did not have the constraints that we mentioned above. Then we tried to recognize the serial numbers from this filtered image dataset.

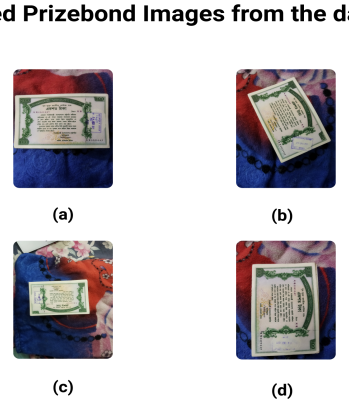

Fig. 2. Filtered prize bond images from the initial database (a) Without rotation (b) Rotated clockwise with less than 90 degrees (c) Rotated clockwise with 180 degrees. (d) Rotated clockwise with 90 degrees.

B. Procedure

From the filtered dataset, we randomly picked an image at first as shown in**Figure** 3 where the background had various objects apart from our object of concern.Then, we applied a four point transformation on the image so that our concerned object, i.e. the prize bond, is considered only in a fixed height. This is applied to all the images in the dataset so that every image will be resized according to

the already defined height. The resulting image after the transformation made is shown in **Figure** 4.

Fig. 3. Randomly picked image.

Fig. 4. After four point transformation.

This will also allow our region of interest(ROI) to always be in the same coordinate of the image. After taking our transformed image, we then rotated our image by 90, 180 and 270 degrees respectively, in order to get the correct alignment. Users may take pictures in different angles which can be seen in some of the sample images from the dataset as shown in **Figure** 2. In order to keep the images in the same alignment, the transformed image was rotated. The transformed image in **Figure** 4 is rotated by 90 degrees anticlockwise to get the image shown in **Figure** 5. After that, the image filtering was done using the UnsharpMask filter with setting radius at 2 and percent at 1000. The images were required to be sharpened so that the images looked more clear and more detailed. The resulting image is shown in **Figure** 6. After that the images were converted into gray-scale images using CV2 to bring all the images into one single layer which simplifies the images on which we can work easily. This is depicted in the image shown in **Figure** 7. Then we cropped the image according to our ROI coordinates from the gray-scale image depicted in the **Figure** 8. After all the above mentioned image processing steps, We were able to start the necessary recognition phase. In this final stage, the Optical Character Recognition (OCR) was done using the Tesseract library which is a very rich and qualitative library for OCR [9].

Fig. 5. After rotation of the image.

Fig. 6. After UnsharpMask filter was applied to the image.

Figure 9 shows the steps that we followed to reach our ultimate result. Another possible methodology could be to recognise the serial number digits simultaneously, then design a sequential ROI detection and classification system. The first CNN (the ROI detection CNN) detects the ROI for the serial number. Using the detected ROI, the second CNN (the classification CNN) classifies all characters in the ROI simultaneously. Because the size of the ROI is pre-determined depending on the kind of banknote and the input image is assumed not to be tilted, it is only necessary to detect a single point in the ROI.

III. Result and Analysis

After finishing all our image processing steps we get a gray-scale image which is depicted in **Figure** 9. From

Fig. 7. Gray-scale Image of the filtered image.

Fig. 8. Selected ROI from the gray-scale image.

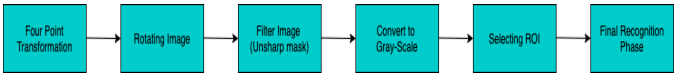

Fig. 9. Block Diagram of the Processing Step

this figure we try recognize the characters by using open source library tesseract [9] with the help of python. The characters are nine in number and the pattern of the characters is as follows. The first two characters are the bangla alphabets and the rest of the seven characters are bangla digits. Therefore, when we provide the gray-scale image to py-tesseract, it provides us the serial number. The serial number we got from the open source are **'খছ ০৩৫৮৯৯৫' , 'হছ ০৩৪৮৯৯৫','খছ ০৩৪৮১১৫', 'খছ ৩৩৫৮৯৯৫','হছ ০৩৫৮৬৬৫', 'খছ ০৩৪৮১৬৯'.** From the results we can see that this open source can not predict accurately sometimes. It mismatches bangla digit **'৯'** with **'১'** and **'৬'**. Also it miss recognizes **'৫'** as **'৪'** as well as **'০'** as **'৩'**. On the other hand while recognizing the bangla alphabets, it also seemed that it is trying to detect **'খ'** as **'হ'**, **'ঝ'** as **'ব'**, **'ক'** as **'ব'** etc. This happens very few times while recognizing the serial numbers. Our current procedure is able to recognize all the character except 1 or 2 characters for few times. This happens because OCR tesseract model training for character recognition is not suitable for prize bond serial number. It can be improved and the highest level of accuracy can be achieved in character recognition by building a model with prize bond character types. Also, while implementing it on the mobile app or website, we can ask the user to check if the detected number is correct or not and change those specific characters if any of those characters is not being recognized properly. The challenging part was recognizing the bangla alphabets. We know the bangla alphabets are just a part of the whole serial number and is not that important when matching with the results. The draw of the prize bond publishes only the bangla number and any one with similar serial bangla number can avail the award. So, the bangla alphabets are not of that much importance. Therefore, in implementation we can skip saving the bangla alphabets and save only the bangla digits and notify user that he/she has won a lottery and provide the serial number.

IV. Conclusion

In this paper, we have proposed a system to extract the serial numbers from prize bonds. Automation is one of the scopes for this work. The initial idea was to develop a mobile app which will detect the serial number from a prize bond and then it will match the retrieved serial number against the serial numbers that won the draw of the prize bonds. The app will send a notification if a match is found which means the user won a prize. It is a redundant work for the user to check each and every time if the result is published or if his or her number is there among several hundred numbers. So automating this part may cancel out all the redundant work and make life a little easier. People may want to make the detection process a bit smooth, people may want to reduce the challenges that we faced. In a word there are room for improvements but we believe future researchers will find more effective ways to solve and address those issues.

References

[1] Lutkevich B. natural language processing (NLP) . Retrieved from: https://searchenterpriseai.techtarget.com/definition/natural-language-processing-NLP

[2] Burel, G., Pottier, I., & Catros, J. Y. (1992, June). Recognition of handwritten digits by image processing and neural network. In Proc. Of the Int. Joint Conf. on Neural Networks, pp. 666-671.

[3] Dan, Z., & Xu, C. (2013, January). The recognition of handwritten digits based on bp neural network and the implementation on android. In 2013 Third International Conference on Intelligent System Design and Engineering Applications, pp. 1498-1501 IEEE.

[4] Rabby, Akm Shahariar Azad & Abujar, Sheikh & Haque, Sadeka & Hossain, Syed. (2018). Bangla Handwritten Digit Recognition Using Convolutional Neural Network. 111-122. $10.1007/978-981-13-1951-8_11$.

[5] B. Feng, M. Ren, X. Zhang and C. Y. Suen,"Extraction of Serial Numbers on Bank Notes," 2013 12th International Conference on Document Analysis and Recognition, 2013, pp. ,-702, doi: 10.1109/ICDAR.2013.143.

[6] J. Qian, D. Qian and M. Zhang, "A Digit Recognition System for Paper Currency Identification Based on Virtual Instruments," 2006 International Conference on Information and Automation, 2006, pp. 228-233, doi: 10.1109/ICINFA.2006.374117.

[7] Singh, R., Yadav, C., Verma, P., & Yadav, V. (2010). Optical Character Recognition (OCR) for Printed Devnagari Script Using Artificial Neural Network.

[8] Singh, A., Bacchuwar, K., & Bhasin, A. (2012). A survey of OCR applications. International Journal of Machine Learning and Computing, 2(3), 314.

[9] Anthony Kay. 2007. Tesseract: an open-source optical character recognition engine. Linux J. 2007, 159 (July 2007), 2.