Towards Development of Real-Time Handwritten Urdu Character to Speech Conversion System for Visually Impaired

Tajwar Sultana
Biomedical Engineering Department
NED University of Engineering & Technology
Karachi,
Pakistan

Abdul Rehman Abbasi
Design Engineering & Applied Research Laboratory
KINPOE (affiliated with PIEAS)
Karachi,
Pakistan

Bilal Ahmed Usmani, Sadeem Khan, Wajeeha Ahmed, Naima Qaseem, Sidra
Biomedical Engineering Department
NED University of Engineering & Technology
Karachi, Pakistan

***Abstract*—Text to Speech (TTS) Conversion Systems have been an area of research for decades and have been developed for both handwritten and typed text in various languages. Existing research shows that it has been a challenging task to deal with Urdu language due to the complexity of Urdu 'Nastaliq' (rich variety in writing styles), therefore, to the best of our knowledge, not much work has been carried out in this area. Keeping in view the importance of Urdu language and the lack of development in this domain, our research focuses on 'handwritten' Urdu TTS system. The idea is to first recognize a handwritten Urdu character and then convert it into an audible human speech. Since handwriting styles of different people vary greatly from each other, a machine learning technique for the recognition part is used i.e., Artificial Neural Networks (ANN). Correctly recognized characters, then, undergo processing which converts them into human speech. Using this methodology, a working prototype has been successfully implemented in MATLAB that gives an overall accuracy of 91.4%. Our design serves as a platform for further research and future enhancements for word and sentence processing, especially for visually impaired people.**

Keywords—Artificial Neural Network; Classification; OCR; Text To Speech; Urdu Handwritten Character

I. Introduction

Urdu is the national language of Pakistan and there are more than 100 million Urdu speakers worldwide1. Urdu is predominantly the combination of two languages i.e. Arabic and Persian which contains variety of features, properties, scripts and writing styles that makes it more difficult for common algorithms to work on it [1].

A TTS System is an application used to read text aloud. TTS systems take text (handwritten or typed) as an input and produce audible speech as an output. They have a wide range of application in different areas like games and education, vocal monitoring, voice enabled email and very useful for visually impaired, etc.

A TTS system is composed of two parts: Optical Character Recognition (OCR) and Speech Synthesis.

OCR is a process of converting an image into machine code. It may be classified into two categories namely online and offline [2]. In online character recognition, the recognition process requires real time data from user and in case of offline character recognition existing (stored) data is used. The complex task of performing accurate recognition is based on the nature of the text to be read and on its quality [3]. The process of OCR comprises of a series of steps that are essential for the preprocessing of input image. Usually, these steps include: Binarization, Segmentation, Feature Extraction and Classification.

The process of speech synthesis creates speech artificially on the basis of the input text. The aim of speech synthesis is to acquire speech that is easily understandable.

In this reported work, we have constructed a mechanism to develop an efficient TTS system. The paper advances with the review of the literature. Then, the proposed methodology is presented that highlights different phases of TTS construction. The major parts of algorithm are image acquisition, preprocessing, classification, speech synthesis and GUI development. The manuscript ends with the discussion of the important results with possible extension of the work.

II. Literature Review

TTS Systems have been an area of research for decades. The complete TTS system was first developed for English language by Noriko Umeda [4]. An overview of the early attempts to develop TTS systems is done by Klatt [5] that gives an extensive database of the methodologies employed to develop these systems.

In fact, the OCR systems should be efficient enough to read handwritten or typed text in different languages such as English, Sindhi, Persian and Arabic etc. M. Farhad et al. [6] proposed a novel methodology for OCR of English alphabets using ANN as the classification algorithm with curvature features of characters as input to the network. They used different seeking angles for the recognition of characters considering the predetermined features of them. This technique yields ~90% accuracy for different character seeking angles but it was computationally expensive as feature extraction proves to be time consuming.

Amit Choudhary et al. [7] presented an extensive work on offline handwritten English character recognition using multilayer feed forward neural network reporting an accuracy of 85.62%. Ganai et al. [8] focused on improving the speech quality of existing system by combining the methods of Hidden Markov Model (HMM)-based speech synthesis and waveform-based speech synthesis to develop human like speech. Despite of the different techniques used for character recognition and speech synthesis, the overall performance of TTS systems developed by both authors led to lesser accurate results and reduced overall performance. Moreover, both TTS systems were developed for English language whose alphabets are easier to detect.

Sarmad Hussain [9] presented his work on Urdu text to speech system. The work emphasized on the use of Urdu phonological processes and divided it into three stages. In the first stage, text was converted into its respective phonemes. In second stage, these phonemes were converted into numerical parameters known as text parameterization and at final stage; speech was synthesized through these parameters. However, this work was based on typed Urdu words.

Kashif Shabeeb and D.S Singh [10] developed a GUI for handwritten Urdu Text to Speech Converter. It was based on the recognition of isolated typed Urdu characters using Artificial Neural Networks (ANN). However, the accuracy of OCR was not mentioned and it did not provide facility for online handwritten Urdu character recognition.

III. Proposed Methodology

Unlike [9] & [10], our methodology is based on recognizing ‘online handwritten’ Urdu text. The framework of the proposed methodology is illustrated in Fig 1 and the details of each step are provided here:

A. *Image Acquisition*

Handwritten Urdu characters are input to the system in the form of images which are captured with the help of a webcam or smart phone as shown in Fig 2. The acquired images may be utilized to function in two modes i.e. offline mode and online mode.

In offline mode, the system uses images which are already stored in dataset whereas in online mode a webcam is used to capture images in real time. The images acquired are then preprocessed for further analysis.

B. *Preprocessing*

The steps involved in preprocessing of an input image are binarization, boundary detection, thinning, feature extraction and padding as shown in Fig 1.

1) Binarization

In the first step, the acquired colored image is converted into grayscale. In order to change any color to its gray level, it is necessary to obtain the value of its primary colors (Red, Green, and Blue). Then, by adding 30% of red, 59% of green and 11% of blue of each value together, we may get our desired grayscale value. Once the image is converted to grayscale, it is then binarized by assigning the value ‘0’ to the weaker intensity of gray showing black and ‘1’ to the stronger intensity of gray showing white.

In fact, a single threshold value is determined using Otsu’s method [11] above which a pixel value is considered as ‘1’ and below it as ‘0’. This is a global binarization technique which is very simple and efficient for separating background and foreground pixels with high accuracy in minimum time. Sample result is shown in Fig 3.

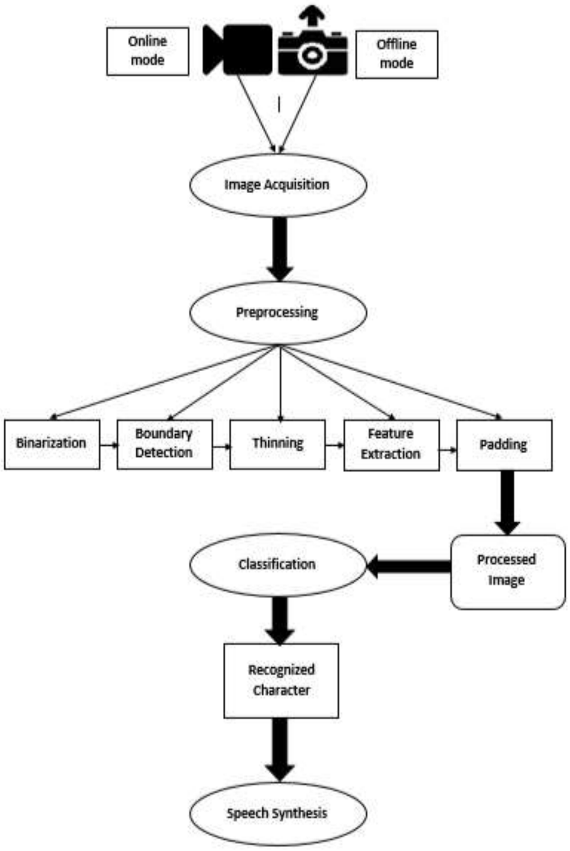


Fig. 1. Methodology of Proposed Framework

Fig. 2. Image Acquisition

2) Boundary Detection

In order to detect the character in an image, it is essential to detect its boundary. By locating regions having abrupt dark-light transfigurations and suppressing regions with homogenous intensity, the boundaries of the characters present in the image can be found. This also provides information about the number of connected elements in an image.

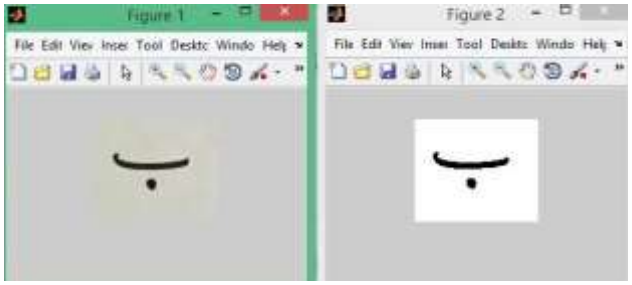


Figure 1
File Edit Viev Inser Tool Desktc Windo Helı
Figure 2
File Edit Viev Inser Tool Desktc Windo Helı

Fig. 3. Input image (left), Binarized image (right)

3) *Thinning*

When applied to binary images, image thinning is used to remove selected foreground pixels that may be presented as noise. The end result of this is another binary image that has these undesired pixels removed. The process of median filtering, which is a non-linear method is applied to perform thinning.

4) *Feature Extraction*

Features are distinctive characteristics present in the image of a character such as dots, strokes, edges, etc. These are extracted and cropped so that the task of classifying characters becomes less rigorous as shown in Fig 4.

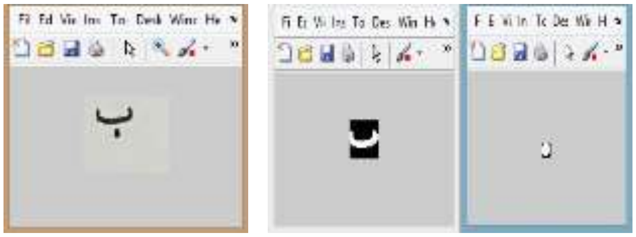


Fig. 4. The 'nuqta' of character 'baay' is extracted here

5) *Padding*

To reduce computational complexity, the processed images are resized and padded so that the dimensions of each image are the same before processing to the classifier as shown in Fig 5.

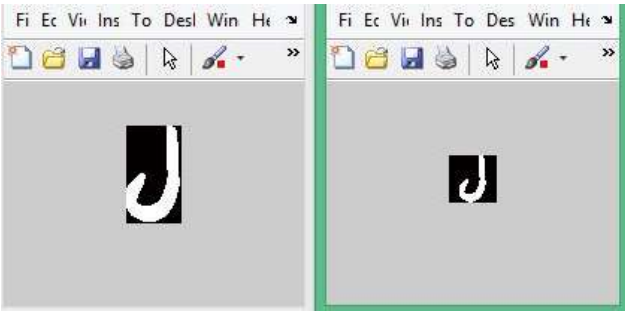


Fig. 5. Cropped image (left), Padded image (right)

C. *Classification*

In this work, an efficient technique of Artificial Intelligence is utilized for the purpose of classification that is Artificial Neural Network (ANN). ANN is a computational tool that replicates the structure and function of biological nervous system. This model can be changed and adapted according to the information that is passed through the network for processing. It consists of three layers: input layer, hidden layer and output layer. These layers comprise of a number of interconnected neurons and weights. All these parameters are set according to the requirement through trial and error method. ANN are very useful for finding patterns in data [12].

1) *Designing the Neural Network*

In this study, the neural network that is used has specific parameters as tabulated in Table 1. Here validation shows network generalization.

TABLE I. PARAMETERS OF NEURAL NETWORK

Network Specification	
Type	Feed forward neural network
Learning method	Back propagation
Number of layers	3
Number of hidden layers	1
Number of nodes in hidden layer	16
Activation function used	Gradient descent
Training set volume	70%
Validation set volume	15%
Testing set volume	15%
Input, Output nodes	1024,24

2) *Sample Preparation and Training*

For the purpose of training the neural network, all the characters are first divided into distinct classes. Characters that have similar primary structure like 'baay', 'paay', 'taay' are placed in a single class and 24 such classes are created on the basis of primary structural discrimination as shown in Table 2. Similarly, based on the number of connected elements, characters are further classified within a class. For example, if the character 'taay' is taken as an input, the primary structure of 'taay' is first classified into the second class as shown in Table 2 and then based on the number of connected elements, which are three in this case, it is recognized as 'taay'. In this way all the characters are recognized into their respective classes. Further classification of individual characters is based on the number of connected elements in each character. Network was trained on 60 samples where each sample represents a matrix of 1024 × 24.

A target matrix of 24 × 24 is also set which represents '1' diagonally at the index of each class with rest of the indices being 0. Network is then trained and ready for implementation as shown in Fig 7.

TABLE II. 24 CLASSES OF URDU ALPHABETS

Class No	Characters/Groups
1	ا
2	ب پ ت ٹ ث
3	ج
4	چ
5	ح خ
6	د ڈ ذ
7	ر ڑ ز ژ
8	س ش
9	ص ض
10	ط
11	ظ
12	ع غ
13	ف
14	ق
15	ک
16	گ
17	ل
18	م
19	ن
20	و
21	ہ
22	ء
23	ی
24	ے

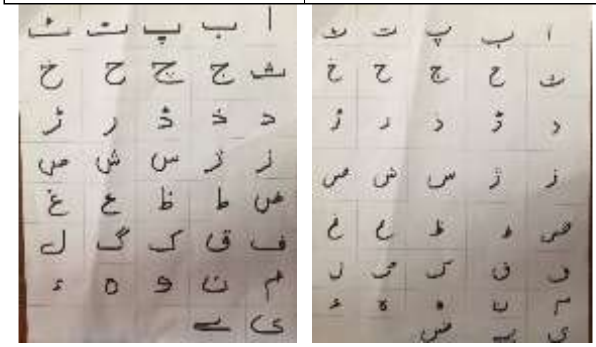
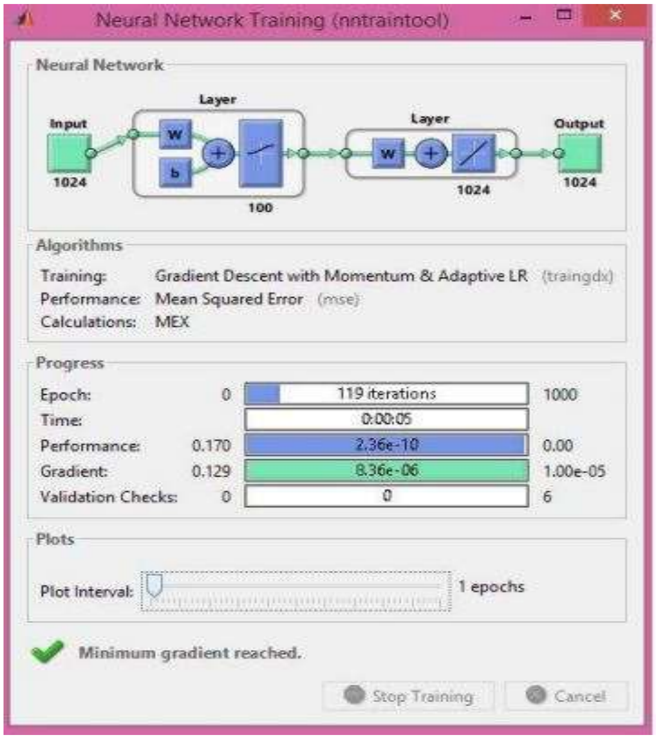


Fig. 6. Sample handwritten Urdu Text



Neural Network Training (nntraintool)

Neural Network

Algorithms

Training: Gradient Descent with Momentum & Adaptive LR (traingdx)
Performance: Mean Squared Error (mse)
Calculations: MEX

Progress

Epoch:	0	119 iterations	1000
Time:		0:00:05	
Performance:	0.170	2.36e-10	0.00
Gradient:	0.129	8.36e-06	1.00e-05
Validation Checks:	0	0	6

Plots

Plot Interval: 1 epochs

✓ Minimum gradient reached.

Fig. 7. Neural Network training

3) *Results of OCR*

To test the system, a novel input is given to the network. This novel input is an image of a character. If the character is correctly recognized, an output matrix of 24x1 is displayed which shows ‘1’ at the place of that character’s index and 0 at the rest of the indices as shown in Fig 8. After 1000 epochs (iterations), an overall accuracy was computed as 91.4%. The 37 characters are classified into 24 distinct classes and there is no major misclassification of characters as shown in Fig 9. An enlarged view of the confusion matrix showing classification of two character classes i.e. 23 and 24 is shown in Fig 10.

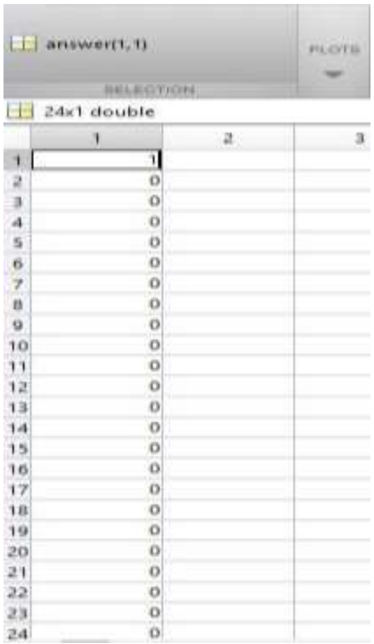
D. Speech Synthesis

Speech synthesis is the final step of the proposed methodology which is achieved using MATLAB to acquire human like speech. For this purpose, Digital Signal Processing (DSP) toolbox is used to manually record prosodies of each character which are then stored in a format of .wav file. A database is created which contains the recorded sounds of all characters. Once the character is recognized, stored sound is retrieved by the system through program for audible output. The speech synthesis system works effectively.

E. Graphical User Interface (GUI) Development

For the purpose of testing, a Graphical User Interface (GUI) is also developed which displays the following features:

- Input image: Shows the uploaded character image.
- Output character image: Shows the image of character recognized by the system.
- Output character text: Shows the character text, recognized by the system.
- Output speech: Shows the speech generation of recognized character.



24x1 double

	1	2	3
1	1		
2	0		
3	0		
4	0		
5	0		
6	0		
7	0		
8	0		
9	0		
10	0		
11	0		
12	0		
13	0		
14	0		
15	0		
16	0		
17	0		
18	0		
19	0		
20	0		
21	0		
22	0		
23	0		
24	0		

Fig. 8. Output matrix of character ‘alif’ displaying 1at first index and 0 at other indices

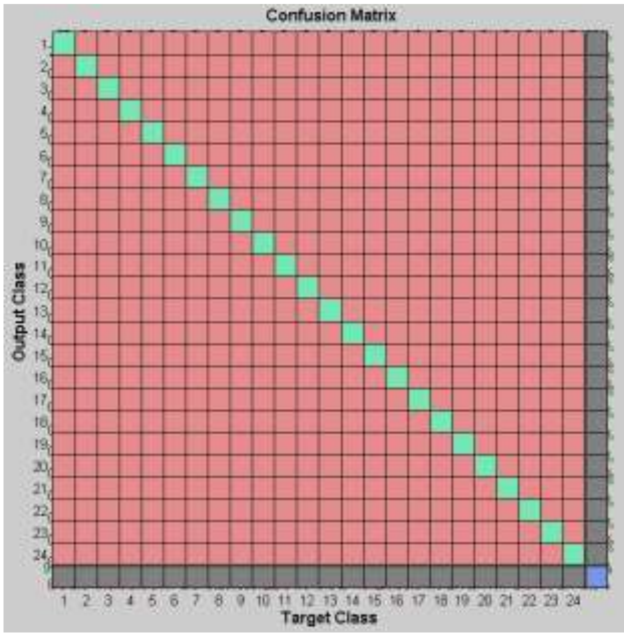


Fig. 9. Confusion Plot

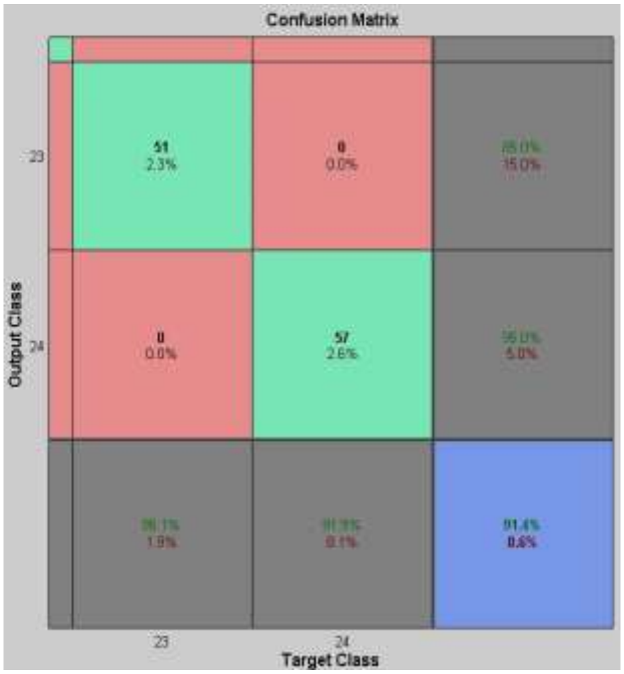


Fig. 10. Enlarged view of confusion plot

IV. Conclusion and Discussion

The system proposed here has two parts; handwritten Urdu character recognition and speech conversion. The Urdu alphabets were initially divided into twenty four classes grouping similar characters. After that, sixty samples of different handwritings were collected and presented to the classifier for training the network. Once the training is completed, novel samples were used for testing and validation. For every input character, a matrix of 24 × 1 is displayed as an output. Results show 91.4% accuracy for 60 sample data set thus ending the recognition process. The recognized character is then passed through MATLAB's digital signal processing toolbox which converts it into its corresponding human speech recorded earlier.

TABLE III. Comparison of Different OCR Techniques

Paper	Languag e	Online/Offlin e	OCR (%- accuracy)
[7]	English	Offline	85.62
[13]	Latin & Bengali	Offline	98.3
[14]	Urdu	Offline	94.97

The proposed approach accomplishes the aim of recognizing Urdu characters in real time and its conversion into human speech. This system has an advantage over other systems as it serves as a basic architecture for handwritten Urdu characters captured in real time. In Table 3, TTS systems developed for different languages are compared and their character recognition accuracies are highlighted. In comparison, the overall OCR accuracy for the proposed methodology is 91%. In addition, the TTS developed here uses simple algorithms, has online recognition feature and a dedicated GUI for easy implementation. Most of the literature available doesn't incorporate online character recognition. Moreover, although a dedicated GUI for TTS is included in many studies [1, 10], the GUI proposed here is very easy to use. On the basis on these advantages, this system can be extended to use for people with limited vision or are illiterate and unable to read important text messages and instructions.

Like other TTS systems, the reliability and efficiency of this system can be affected due to certain environment constraints (such as light intensity and direction) while taking real time data as input. It is observed that poor visibility acts as a barrier for correct OCR process. The accuracy of OCR and classification can further be enhanced by including other features like estimating the curve angles of characters [6]. We anticipate that although this approach produces better results, improved algorithms will be needed to improve the computational time.

The database used for training classifier only consists of handwriting styles of the people aged between 18 and 24 years. The above system can be improved by presenting the handwritten samples of children and senior citizens to the classifier. The network trained on the basis of the handwriting of these age groups covers a wide range of handwritten text and improves the OCR process. In addition, this modification is beneficial so that the TTS system can be used for the senior citizens having poor muscle control due to disease or illness.

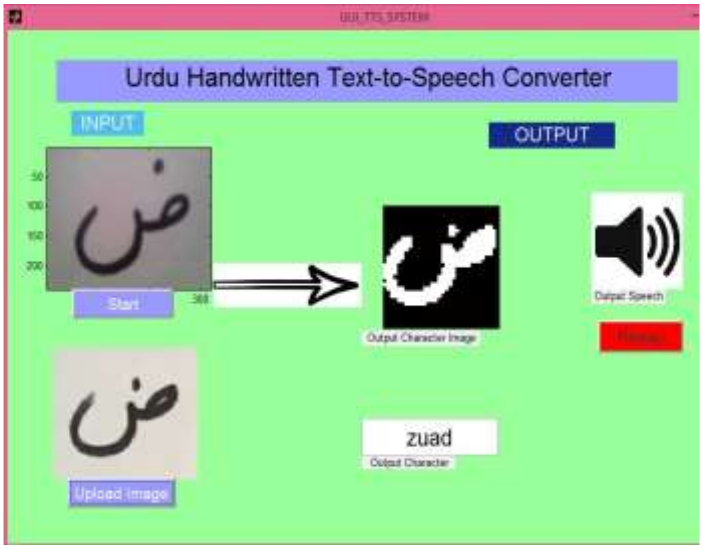


Fig. 11. Graphical User Interface showing character to speech Conversion of Urdu Character 'Zuad'

In addition to the technique proposed above, we have also attempted template matching rather than ANN but it did not produce good results. This was mainly because of the obvious differences in typed character's topology with the handwritten character's shape.

This research project can be further extended for complete Urdu words and complete sentences for a fully functional TTS system that may serve the visually impaired people. The system may also be extended as platform independent to any particular software such as MATLAB, etc.

References

[1] S. A. S. S.-u. Haque and M. K. Pathan, "A finite state model for urdu nastalique optical character recognition," IJCSNS, vol. 9, p. 116, 2009.

[2] S. Sardar and A. Wahab, "Optical character recognition system for Urdu," in Information and Emerging Technologies (ICIET), 2010 International Conference on, 2010, pp. 1-5.

[3] S. G. Dedgaonkar, et al., "Survey of methods for character recognition," International Journal of Engineering and Innovative Technology (IJEIT) Volume, vol. 1, pp. 180-189, 2012.

[4] N. Umeda, et al., "Synthesis of fairy tales using an analog vocal tract," in Proceedings of 6th International Congress on Acoustics, 1968, pp. B159-162.

[5] D. H. Klatt, "Review of text-to-speech conversion for English," The Journal of the Acoustical Society of America, vol. 82, pp. 737-793, 1987.

[6] M. Farhad, et al., "An efficient Optical Character Recognition algorithm using artificial neural network by curvature properties of characters," in Informatics, Electronics & Vision (ICIEV), 2014 International Conference on, 2014, pp. 1-5.

[7] Choudhary, et al., "Off-line handwritten character recognition using features extracted from binarization technique," AASRI Procedia, vol. 4, pp. 306-312, 2013.

[8] M. B. Ganai and E. J. Arora, "Text-to-Speech Conversion," 2016.

[9] S. Hussain, "Phonological Processing for Urdu Text to Speech System," Yadava, Y, Bhattarai, G, Lohani, RR, Prasain, B and Parajuli, K (eds.) Contemporary issues in Nepalese linguistics, 2005.

[10] D. S. S. Kashif Shabeeb, "Urdu Text to Speech Convertor: Database Creation and GUI Developement," International Journal of Computer Science And Technology, vol. Vol. 6, pp. 191 - 193, 2015.

[11] N. Otsu, "A threshold selection method from gray-level histograms," Automatica, vol. 11, pp. 23-27, 1975.

[12] S. Barve, "Artificial Neural Network Based on Optical Character Recognition," in International Journal of Engineering Research and Technology, 2012.

[13] S. Pal, et al., "Line-wise text identification in comic books: A support vector machine-based approach," in 2016 International Joint Conference on Neural Networks (IJCNN), 2016, pp. 3995-4000.

[14] S. Naz, et al., "Urdu Nasta'liq text recognition system based on multi-dimensional recurrent neural network and statistical features," Neural Computing and Applications, pp. 1-13, 2015.