An Assistive Reading System for Visually Impaired using OCR and TTS

Akshay Sharma
PEC University of Technology
Electronics and Communication Engineering Department, PEC University of Technology, Sector-12, Chandigarh, India

Abhishek Srivastava
PEC University of Technology
Electronics and Communication Engineering Department, PEC University of Technology, Sector-12, Chandigarh, India

Adhar Vashishth
PEC University of Technology
Electronics and Communication Engineering Department, PEC University of Technology, Sector-12, Chandigarh, India

ABSTRACT
Reading machines are mechatronic devices which use optical character recognition and text-to-speech technology in order to output synthetic voice from printed text. In this paper an assistive system has been proposed for visually impaired or blind persons. It reads textual information on papers and produces corresponding voice using OCR (Optical Character Recognition)and TTS (Text-to-speech) system. To localize text regions in images connected component labeling approach using histogram analysis is done on binarized image. TTS system using Concatenative synthesis based on SDK (Software Development Kit) platform is used. This system is operated via a voice-based user interface and also has a user friendly GUI (graphical user interface) to scan the text and to control various speech parameters. Speech signal produced can be saved and reproduced for later use.

Key Words
Text Information Extraction(TIE), Optical Character Recognition (OCR), Connected Component Labeling, Text-to-speech (TTS),Concatenative synthesis, Graphical User Interface(GUI)

1. INTRODUCTION
Despite the advancement of technology that allows for storing information electronically, textual information presented on papers still remains the most common mode of information exchange. However, such information is not available for visually impaired citizens. To improve their ability to access textual information we propose an assistive system that reads texts from scanned documents and represents the textual information in the form of speech. The development of such systems requires use of two technologies that are central to these systems, namely OCR (Optical Character Recognition) for Text Information Extraction (TIE) and TTS (Text-to-speech) to convert this text to speech.

Text Information Extraction (TIE) is the first and important function of any assistive reading system and is an integral part of OCR because this process determines the intelligibility of the output speech. In recent years, the automatic detection of texts from images and videos has gained increasing attention. However, the large variations in text fonts, colors, styles, and sizes, as well as the low contrast between the text and the complicated background, often make TIE extremely challenging. To find a completely robust and generalized method for TIE still remains an area of research. A lot of efforts have been put on addressing these problems. Text extraction techniques can be divided into four categories. The first category is based on edges [1] which assume high contrast differences between the text and background. It is fast and can have high recall rate. However, it often produces many false alarms since the background may also have strong edges similar to the text. The second category uses connected component analysis (CCA) [2], in which pixels with similar colors are grouped into connected components, and then into text regions. CCA is fast but, it fails when the texts are not homogeneous and text parts are not dominant in the image. The third category is based on textures [3] and assumes that texts have specific texture patterns. It is more time-consuming and can fail when the background is cluttered with text. The fourth category is based on frequencies [4]. In this kind of approach, the text is extracted from the background in the frequency (e.g. wavelet)domain. This is also time consuming, and the frequency representation may not be better than the spatial representation. Recently, there is a lot of interest in using pattern classification techniques (such as AdaBoost [5], support vector machines [6, 7, 8], belief propagation [9] and neural networks [10]) for text localization. With the help of elaborately designed features that incorporate various properties of the text (such as geometry, color, texture and frequency), these techniques are often successful indiscriminating text from its background.

TTS or speech synthesis is a technique for generating intelligible, natural-sounding artificial speech for a given text [11]. The methodology used in TTS is to exploit acoustic representations of speech for synthesis, together with linguistic analysis of text to extract correct pronunciations (‘‘content’’; what is being said) and prosody in context (‘‘melody’’ of a sentence; how it is being said). Speech synthesis system can be divided into two parts i) Front End also called Natural Language Processing module (NLP) [12] used to analyze text, and ii) Back End also called Signal-processing module that generates the speech waveform based on information from the front end. Front end contains: text processor (normalization and letter-to-sound), prosody control, unit selection [13]. So it is basically concerned with the conversion of grapheme- to-phoneme. This process is also called ‘‘letter-to-sound’’ conversion. Back End is concerned with technique used for synthesis. There are two techniques [14]: format synthesis [15, 16,17] and concatenative synthesis [18,19,20].Format synthesis depends on acoustical models in order to produce parametric driven speech, while concatenative synthesis, concatenates segments of recorded speech. Format synthesis can be highly intelligible, but due to the difficult and complex task of obtaining good enough speech models, the synthesized speech has so far a degraded speech quality to some extent. Whereas Concatenative synthesis can be very natural in the sense of having a speech quality close to human speech, but it may suffer from audible discontinuities at concatenation points.

The paper is organized as follows: Section II provides system overview of the proposed system. Detailed description of OCR

Module is given in Section III and in Section IV, TTS Module is discussed in detail. Experimental results are discussed in section V. Conclusion and future scope is discussed in section VI

2. SYSTEM DESIGN

The proposed system can be broadly divided into two modules: OCR module and TTS module as shown in fig. 1. Text is scanned using a scanner to get the image which is given as input to OCR module. The point of the leftmost corner in the sheet is assigned as origin of the coordinates, and Euclidean geometry is considered.

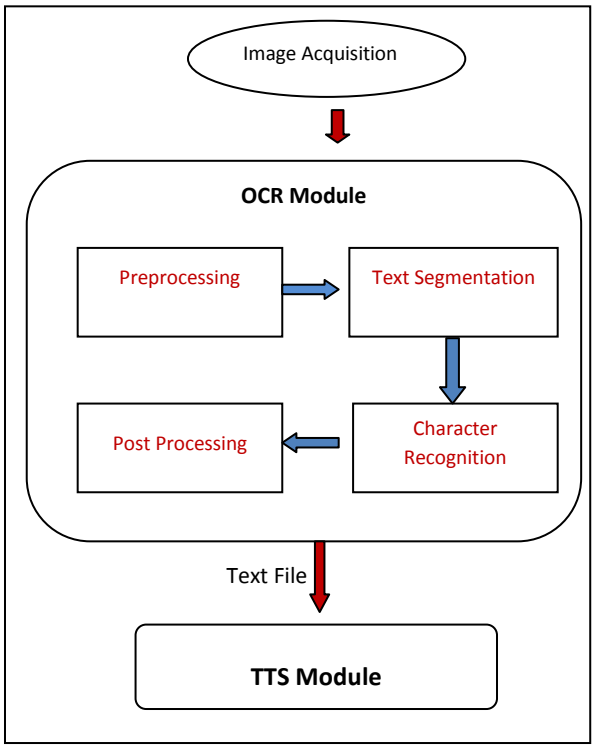


Fig. 1 System architecture

First, text is converted into image using a scanner. Pre-processing is done on the image to reduce the noise and Skew present in image. Then image is binarized and segmented into text and non text regions. Individual characters are isolated and normalized (with predefined size) in order to facilitate feature extraction process and also improve their classification accuracy. Post processing is done to group the various characters together so as to form meaningful word and numbers. Then text file generated above is converted to speech signal.

The GUI of system is shown in fig.2.It can select scanned text files available, using the browsing feature. Text extracted from the image is displayed in the GUI. Speech signal generated from text can be saved and reproduced whenever required. Users can also adjust speech rate, volume and tone according to their requirements.

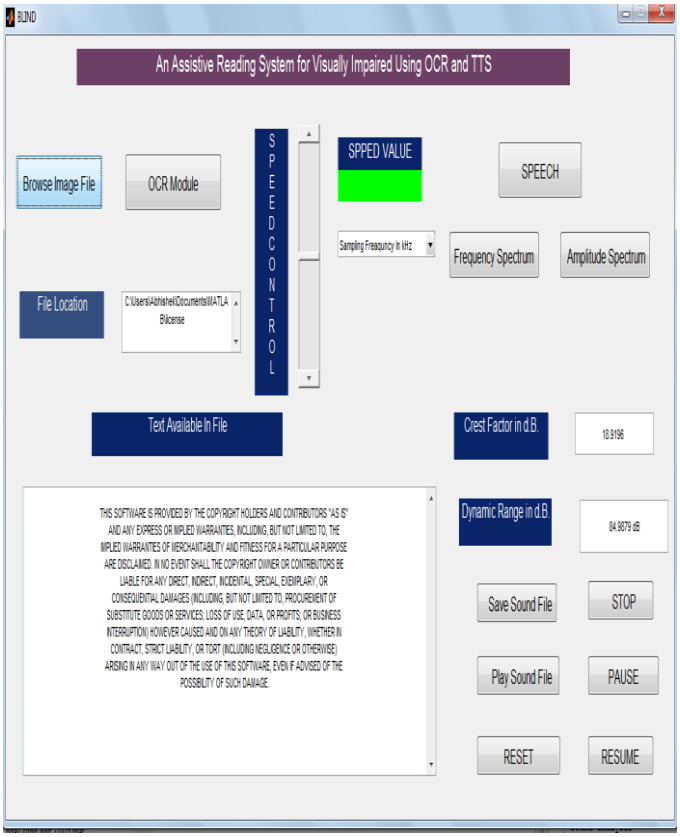


Fig. 2. Graphical User Interface

3. OCR MODULE

The goal of OCR Module is to classify optical patterns (often contained in a digital image) corresponding to alphanumeric or other characters. The process of OCR involves several stages as shown in fig.3.

Each stage is discussed as follows:

3.1. Pre Processing Stage

1. Noise Removal: The primary methods of noise removal are linear low pass filtering using various filter templates. But it decreases the noise at the cost of increase in blurring. And many a times it may cause a particular character misrecognised by OCR. This drawback is efficiently overcome by anisotropic filtering mechanism by approximately weighing the filtering coefficients. Such a weight is based on a suitable monotonically decreasing function $g(||\nabla I||)$ of the local gradient.

$$I(x,y,t) = div(c(x,y,t)\ \nabla I(x,y,t)) \quad (1)$$

where,

$C(x,y,t) = g(||\nabla I(x,y,t)||)$

$I(x,y,0)$ is the initial image and

$I(x,y,t)$ is the filtered scale-space image

Anisotropic diffusion based filtering removes noise at pixel locations where gradient doesn't have large oscillations. So it removes the background noise but the structural elements of characters are preserved.

2. Image Binarization: Binarization is a technique by which the gray scale images are converted to binary images. The most common method is to select a proper threshold for the image. After that all the intensity values above the threshold intensity and all intensity values below the threshold are assigned

either 1 or 0 depending upon the convention used. These intensity value represent either "black" or "white" pixel

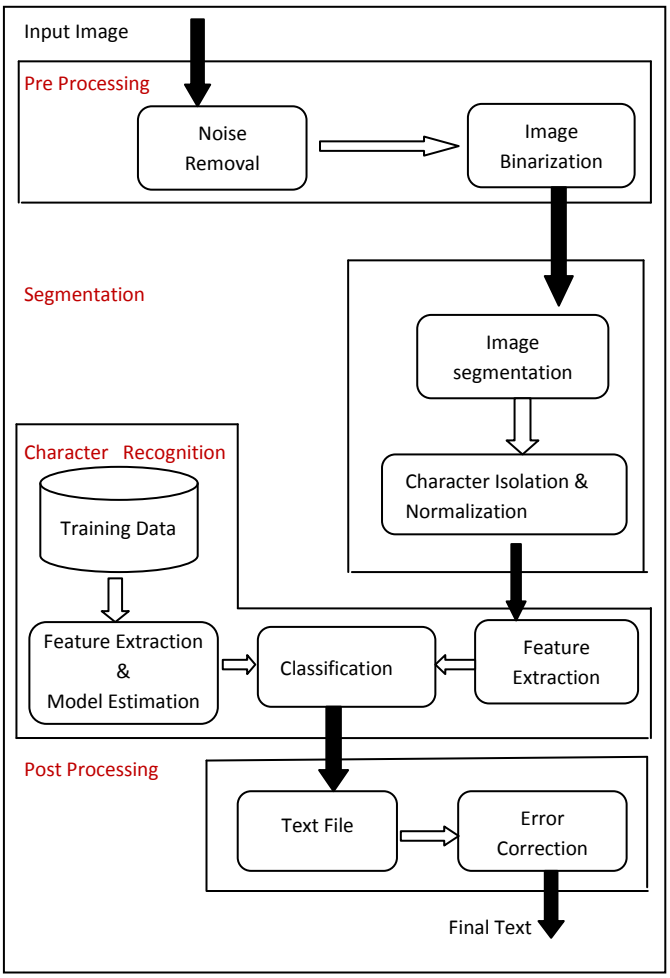


Fig. 3. OCR Module

3.2. Segmentation

1. Image Segmentation: Here segmentation occurs on two levels. On the first level, if the document contains both text and graphics, these are separated for subsequent processing using connected component analysis. On the second level, segmentation is performed on text by locating columns, paragraphs, words, and characters. The image of the document is scanned from the top-left to the bottom-right. We use connected component labelling [2,21] approach to segment the image into background and non-background (text or picture) regions using histogram analysis. A connected component can be either a character or a picture (or parts of a picture). Label is assigned to each connected component in an image according to its size. So a pixel in label image may contain 0 to represent background, or any value other than 0 (value of one of the size labels assigned according to the component's height) to represent non-background. So in label image value of each pixel is not limited to only 0 and 1 as in a binary image and this helps in grouping similar connected components, hence forming homogeneous regions that consist of components of similar height. A connected component can be either a character or a picture (or parts of a picture). Image is scaled down to reduce the time taken for connected component labelling.
First, histogram analysis is applied vertically, counting the number of non-background pixels in every row. When a row is known to have non-background pixels, it is determined as a possible starting row of one or more regions. Then, the ending row is defined as the first row which contains background pixels after the starting row. The vertical scan aims to get a line of text (or text with a picture expanding to several lines of text). Once the starting and ending rows are obtained, similar histogram analysis is performed horizontally. The horizontal scan is done to separate the region into columns. At this point, we have a range in the image that is estimated to include one or more homogeneous regions. To ensure that there is only one individual region in the range, vertical histogram analysis is performed once more in that range, further dividing the range vertically in case the range contains more than one region. After this third histogram analysis, we have a new range that should consist of only one region.
A number of statistical measurements have been proposed for distinguishing between text and non-text regions for binary document images [22, 23]. Two of them are adopted in the present system:(i) mean length of horizontal black runs (MBRL) and (ii) white-black transition count per unit width (MTC). These two features can be calculated by scanning a region from left to right in a row-by-row manner. As the scanning proceeds, the horizontal black run-length is accumulated by a counter, BRL and white-black transition count by another counter, TC. After whole region is scanned, the two features are computed by:

$$MBRL = \frac{\text{BRL}}{\text{TC}} \quad (2)$$

$$MTC = \frac{\text{TC}}{\text{W}} \quad (3)$$

where, W is the width (in pixel) of the region under consideration.
MBRL and MTC are used for classifying each region (e.g. text line, picture) as either text or non-text. If the values of MBRL and MTC indicate that it is a text region, it is not marked directly as text region, but rather is examined further based on its height. If its height is much larger than the expected height of a text line (e.g. double of the average height), histogram analysis from the first row-by-row scheme is performed again on that particular region. Otherwise the region is marked as text. Now image is rescaled and this rescaled label image is then combined with the original binary image, which was produced earlier by pre-processing module, in order to extract only text from the original image. Given the resulting image from layout analysis, the next step is to extract individual characters (e.g. letters, numbers, and symbols/punctuation marks). Since it is assumed that the image contains only text, histogram analysis is sufficient to separate and extract each character.

2. Normalization: In the normalization process, the character image is mapped onto a standard plane (with predefined size) so as to give a representation of fixed dimensionality for classification. The goal for character normalization is to reduce the within-class variation of the shapes of the characters / digits in order to facilitate feature extraction process and also

improve their classification accuracy. The character image is normalized to 12x12 pixels.

3.3. Character Recognition

In character recognition, classification process is used to recognise the extracted character by comparing its certain features with the features of characters stored using the training data. There are two steps in building a classifier: training and testing. In training various images are given as input to the OCR and characters are extracted. Then features of these characters are extracted and stored in a vector. From this finite set of feature vectors we estimate a model (usually statistical) for each class of the training data. During testing we compare feature vectors to the various models and find the closest match.

3.4. Post Processing

Now a string of recognized characters is obtained but to produce an understandable text, the characters must be arranged into words by inserting spaces. The position of a space between two words is approximated as 2.6 (obtained experimentally) times of the average horizontal distance between characters in a word. Thus, for each detected character through histogram analysis, the distance between it and the previous character is calculated. When the distance is larger than 2.6 times the average distance in the word, a space is inserted to generate a text file.

4. TTS MODULE

The TTS module uses Concatenative synthesis for production of speech. Concatenative synthesis uses actual short segments of recorded speech that were cut from recordings and stored in an inventory (voice database), either as waveforms (not encoded), or encoded by a suitable speech coding method. General architecture of a Concatenative synthesis based text-to-speech system is shown in fig. 4. There are three variants of concatenative synthesis based on the types of speech units stored in the inventory: domain specific synthesis, diphone synthesis and unit selection synthesis [24].

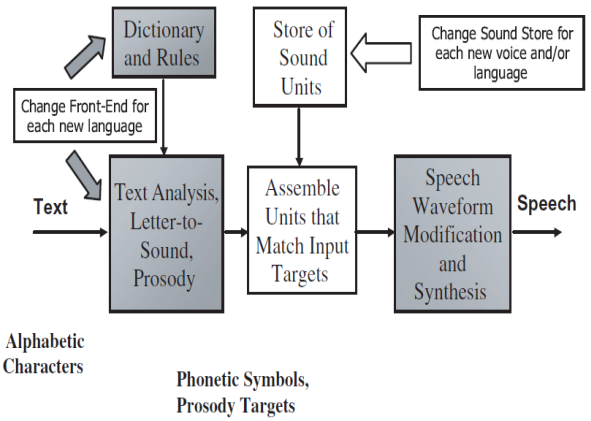


Fig. 4. Architecture of a Concatenative text-to-speech system

Domain specific synthesis normally concatenates words or phrases of speech, and can be used when the output of the synthesis system is limited to a small domain of utterances. Diphone synthesis speech databases consist of only one unit of each diphone [25] occurring in the language. During synthesis, pitch and duration modification are used to obtain a desired prosody. Unit selection synthesis is the most popular variant of concatenative synthesis, and was first proposed by Nakajama and Hamada in 1988[26]. Since then various systems including commercial systems were developed resulting in a higher level of reading-style synthetic speech [27, 28, 29] and it is today considered as the state of the art in text-to speech synthesis.

TTS module is based on MATLAB GUI and uses SDK platform of windows based systems. The Speech API can be viewed as an interface used between applications and speech engines (recognition and synthesis). A MATLAB GUI has been developed to communicate with speech API by sending events using standard callback mechanisms (Window Message, callback proc or Win32 Event) as shown in Fig. 5. These callback mechanisms make speech API accessible from a variety of programming languages by using a standard set of interfaces [30]. In addition, it is possible for anybody to produce their own Speech Recognition and Text-To-Speech engines or adapt existing engines to work with API.

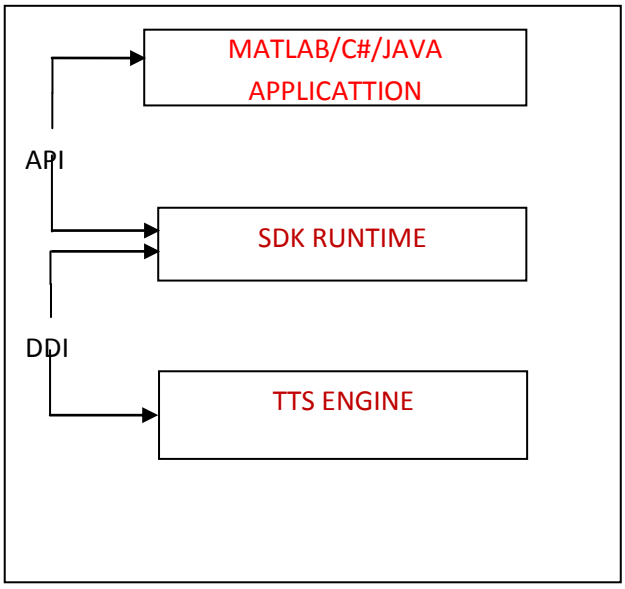


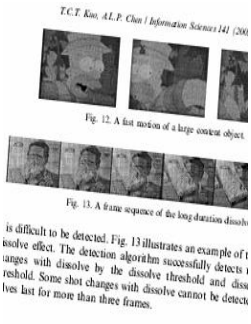
Fig. 5. Architecture of TTS System

5. RESULTS

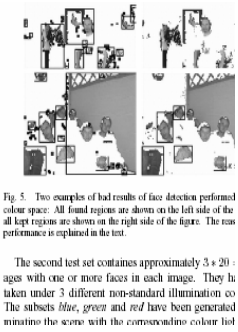
Many scanned documents were converted into text. Then these text file were converted into speech signal. This method is able to segment text from graphics and background efficiently as shown in fig.6.Even in text pages having large variations in text fonts, colors, styles, and sizes, the proposed method is successful in extracting text from these challenging cases also. Images of the magazine pages were scanned using a 300 dpi scanner and the proposed system was evaluated with gray-scale images. The proposed algorithm runs on a PC with Intel Core 2 Quad CPU at 2.66GHz and 4 GB memory under Microsoft 7.

One particular problem is that some inner parts of non-text objects were also labeled -1 since they had height similar to text objects. Those inner parts were actually surrounded by large non-text objects that had different labels and histogram analysis practically saw only - 1 and background labels. So, when histogram analysis was conducted to find the starting and ending points of a region, it failed to see that those -1 labels were

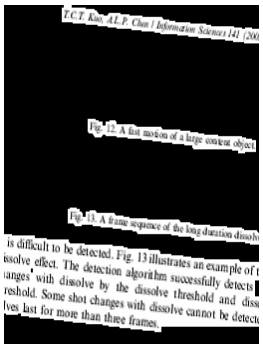
actually parts of a larger non-text object. In addition, MBRL and MTC calculation showed that those spots had similar characteristics to a text region, causing them to be marked as text as shown in Fig. 6(f).



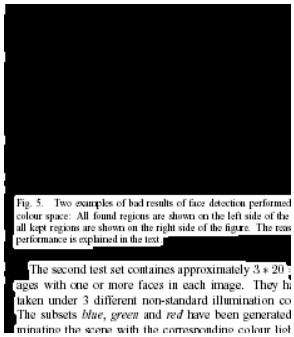
6(a)



6(b)



6(c)



6(d)

6(e)

6(f)

6(a),6(b) 6(e) Original image 6(c),6(d) & 6(f) Final output image after segmentation

Quality of speech can be determined using crest factor and dynamic range. Crest factor is the ratio between peak (crest) level and RMS level of a wave form. This is an important parameter when a voice is to be recorded or reproduced in an electro acoustic system. Dynamic range is the ratio of the loudest sound to that of the quietest sound in a piece of equipment or a complete system, expressed in decibels (dB). Ideally typical dynamic range is 120 dB [31] and crest factor is14 dB to 20 dB. So in order to make the softest speech sounds audible and the loudest still comfortable, it is important to know the dynamic range for speech sounds. Results show that crest factor is between 18-20 db and dynamic range is between 80-90 db which lies in permissible limit for audible speech. For average long term speech spectrum (talking over one minute) maximum energy is in the range of 250Hz to 500Hz band and speech analysis in frequency domain is in conformation with this. These lower-frequency bands correspond to vowel sounds, the higher-frequency bands in the 2k Hz and 4k Hz region correspond to consonant sounds. Vowels carry the power of the voice and consonants provide intelligibility.

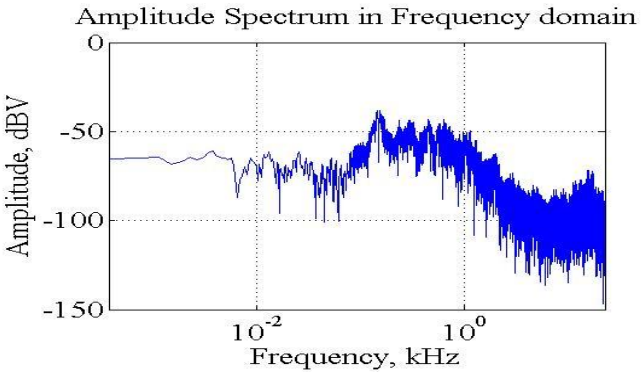


Fig.7.Speech Signal in frequency domain

6. CONCLUSION

In this paper we have presented a scheme for selection and separation of text and its conversion to speech from a scanned document image to build an assistive system aiming to support people with visual impairment. It works well for images having variations in text fonts, colors, and sizes, as well as the low contrast between the text and the often complicated background. Anisotropic diffusion reduces the additive noise efficiently. Generally additive noise is found in all kinds of old documents like newspaper, book set and is removed without increasing blurring.

TTS module works efficiently for various input texts and was perfectly audible as confirmed by the values of Crest Factor and Dynamic range. This system is limited to one voice but can be extended to include more voices by doing minor changes. This system can also be extended to extract text from camera based document images as it works efficiently for them too. Error correction can be added to the post processing part of OCR Module to increase the accuracy of the purposed system. In future this system can be modified by applying neural networks to further increase its accuracy.

7. ACKNOWLEDGMENT

We would like to thank Department of Electronics & Communication Engineering of PEC University of Technology, Chandigarh for providing all the support.

8. REFERENCES

[1] M. Lyu, J. Song, M. Cai, A comprehensive method for multilingual video text detection, localization, and extraction, IEEE Transactions on Circuits and Systems for Video Technology 15 (2) (2005) 243–255.

[2] J. Lim, J. Park, G.G. Medioni, Text segmentation in color images using tensor voting, Image and Vision Computing 25 (5) (2007) 671–685

[3] K.I. Kim, K. Jung, J.H. Kim, Texture-based approach for text detection in images using support vector machines and continuously adaptive mean shift algorithm, IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (12) (2003) 1631–1639

[4] S. Kumar, R. Gupta, N. Khanna, S. Chaudhury, S.D. Joshi, Text extraction and document image segmentation using matched wavelets and MFR model, IEEE Transactions on Image Processing 16 (8) (2007) 2117–2128.

[5] D. Chen, O. Jean-Marc, B. Herve, Text detection and recognition in images and video frames, Pattern Recognition 37 (3) (2004) 595–608.

[6] C. Jung, Q. Liu, J. Kim, Accurate text localization in images based on SVM output scores, Image and Vision Computing 27 (2009) 1295–1301.

[7] Q.X. Ye, Q.M. Huang, W. Gao, D.B. Zhao, Fast and robust text detection in images and video frames, Image and Vision Computing 23 (6) (2005) 565–576.

[8] M. Anthimopoulos, B. Gatos and I. Pratikakis, A two-stage scheme for text detection in video images, Image and Vision Computing, (2010)

[9] H.Y. Shen, J. Coughlan, V. Ivanchenko, Figure-ground segmentation using factor graphs, Image and Vision Computing 27 (7) (2009) 854–863.

[10] C. Strouthopoulos, N. Papamarkos, Text identification for document image analysis using a neural network, Image and Vision Computing 16 (12–13) (1998) 879–896

[11] Tokuda et al," Speech Synthesis Based on Hidden Markov Models",Proceedings of the IEEE | Vol. 101, No. 5, May 2013

[12] A. G. Ramakrishnan, Lakshmish N Kaushik, LaxmiNarayana. M, "Natural Language Processing for Tamil TTS", Proc. 3rd Language and Technology Conference, Poznan, Poland, October 5-7, 2007

[13] Chen, G.L., Yue, D.J., Zu, Y.Q., Yu, Z.L., "An embedded English synthesis approach based on speech concatenation and smoothing", ISCSLP2004, pp.157-160, Hong Kong, Dec. 2004

[14] T. Dutoit, "An Introduction to Text-to-Speech Synthesis". Dordrecht/Boston/London: Kluwer Academic Publishers, 1997.

[15] T. Styger and E. Keller, Fundamentals ofSpeech Synthesis and Speech Recognition: Basic Concepts, State of the Art, and Future Challenges in Formant synthesis, In Keller E. (ed.), 109-128, Chichester: John Wiley, 1994., 4,5

[16] 13. D.H. Klatt, ''Software for a cascade/parallel formant synthesizer,'' J. Acoust. Soc. Am., vol. 67, no. 3,971–995, 1980.

[17] J. Allen, M.S. Hunnicutt, and D. Klatt, From Text to Speech, The MITalk System, Cambridge: CambridgeUniversity Press, 1987

[18] Moulines, E., Charpentier, F. "Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones", Speech Communication, Vol.9, pp.453-468, 1990

[19] Sproat, R., Hirschberg, J., Yarowsky, D., "A corpus-based synthesizer", ICSLP1992, pp.563-566, Alberta, Canada, Oct. 1992

[20] Van Santen J., Sproat, R., Olive, J., Hirshberg, J., editors, Progress in Speech Synthesis, Springer Verlag, New York, 1995

[21] Gonzalez, R. C. andWoods, R. E. 1992."Digital Image Processing". Addison-Wesley.

[22] Wang Y., Phillips I. T., and Haralick, R.M. 2006. Document zone content classificationand its performance evaluation.Pattern Recognition, 39: 57-73.

[23] Shih, F. Y. and Chen, S. S. 1996. Adaptivedocument block segmentation andclassification. IEEE Transfusion. SystemMan and Cybernetics-PART B: Cybernetics,26, 5: 797-802.

[24] IngmundBjørkan,Speech Generation and Modification in Concatenative Speech Synthesis Ph D Thesis,Norwegian University of Science and Technology .Faculty of Information Technology, Mathematics and Electrical Engineering, Department of Electronics and Telecommunications 2010

[25] Sproat, R. and Oliver, J. "An Approach to Text-to-Speech Synthesis". Chapter 17 in book "Speech Coding and Synthesis", Elsevier, 1995

[26] S. Nakajima and H. Hamada, "Automatic generation of Synthesis Units based on context oriented clustering", Proc. ICASSP 1988, pp. 659-662, (New York, USA), 1988].

[27] R. E. Donovan and E. M. Eide, ''The IBM trainable speech synthesis system,'' in Proc. Int. Conf. Spoken Lang. Process., 1998, pp. 1703–1706.

[28] B. Beutnagel, A. Conkie, J.Schroeter, Y. Stylianou, and A. Syrdal, ''The AT&T Next-Gen TTS system,'' in Proc. Joint ASA/EAA/DAEA Meeting, 1999,pp. 15–19.

[29] G. Coorman, J. Fackrell, P. Rutten, and B. Coile, ''Segment selection in the L&H realspeak laboratory TTS system,'' in Proc. Int. Conf. Spoken Lang. Process., 2000,pp. 395–398.]

[30] http://msdn.microsoft.com/en-us/library/ms720151(v=vs.85).aspx.

[31] Zenget a," Speech dynamic range for cochlear implants". J. Acoust. Soc. Am., Vol. 111, No. 1, Pt. 1, Jan. 2002