Image to Text Converter and Translator using Deep Learning and Image Processing

Swapnil Sinha, Harsh Kumar Kataruka, Vijayakumar Kuppusamy

Abstract— In this paper, we will present a general study on Image and text converter using deep learning and image processing. The main contents will include how the application will work and all the processing, segmentation, feature extraction, recognition and other steps involved. We will also present how this can be used in our daily lives.

Keywords— Deep learning, Image processing, Optical character recognition, PyTessaract, Tensor flow, Python.

I. INTRODUCTION:

In the field of Image processing data extraction is one of the most complicated. It is a difficult process to determine and perform recognition of text from image. But computer software like OCR can help us in extraction of text from image with the help of complex algorithm. Different types of translators are out there such as voice translator and text based translator but these translator are not easy to use. The main aim of this project is to show how a tight dynamic connection can be made between a text and interactive visualization image. The objective of our software is to identify words from an image with the use of neural networks and translate it to the language we wish to, using a Google translator API. We have to build an appropriate neural network and properly train it. Extraction of words should be done by the program one by one and for training purposes, mapping of the target result. For Training of Classification engine for recognition purpose, automatic processing of the image has to be done. Now a days, everyone has smartphones in this world and they use it all the time so that they can capture an image. The main aim of our system is to extract the words from the input image and convert it into the text. Extraction of text from the documents which are scanned are complex because of different backgrounds and also lot of difference of patterns of words such scale, colour, size of the font, intensity and orientation. Hence, to extract the texture from smartphone captured images, detection of text & extraction is an essential part which computes the regions of the images containing text characters. Once an image is captured from smartphone camera, the image then goes through a lot of processes where detection of text from the image is done and text extraction in done and then translation of the extracted text is done.

Swapnil Sinha*, School of Computer Science and Engineering, Vellore Institute of Technology, Vellore

Harsh Kumar Kataruka, School of Computer Science and Engineering, Vellore Institute of Technology, Vellore

Vijayakumar Kuppusamy, School of Computer Science and Engineering, Vellore Institute of Technology, Vellore

II. BACKGROUND:

OCR's growth started during the late 19th Century, and it is still uplifting everyday. The following capture a few memories in the life of OCR so far. This report presents as a guide and update for all the avid readers working in the Optical Character Recognition (OCR) department. [1] It gives out a complete Optical Character Recognition (OCR) system for documented text for captured image. First of all, extraction and skewness of text areas are corrected. After this, binarization and segmentation of text regions are done into lines and characters. Texts are then passed on to module for recognition. This narrates the different ways for converting all the text related content from a paper into system readable format. A revolutionizing technique called Optical Character Recognition is used by the computer to recognize the characters in the document [3] The study of changes in the models of Artificial Neural Network and recognition of the characters in a given scanned documents are done. This reveals how different Neural Network Models tend to behave in Optical Character Recognition. Neural network for the most part uses the OCR. [4] OCR provides us the solution to the problem of recognition of handwriting character. For more accuracy and fast implementation, this technique is used. All the Characters are classified into a 26 pattern class constructed on suitable properties. [5] This report gives us insights into an effective algorithm which deals with bleed through effect which exists in the images of documents. Double-sided images scanned simultaneously are used as inputs, and the bleed-through effect is detected then removed after the registration of the side images. [6] This Establishment of recognition and extraction of the text from the images clicked by camera based smartphones and when the text is recognized then the details about the text can be obtained. [7] This report shows the latest way to ease the Handwritten Characters Recognition based on the behavior of schools of fish and flocks of birds. [8] This clearly explains that attributes of the classification methods which have been fortunately applied to OCR and the other problems can be possibly solved by learning methods. [9] This report derives the IRIS plant classification by using the Neural Network. Classification method is a machine learning technique used to predict group membership for data instances. It proposes a well-organized system which includes the feature extraction module and the classifier module which are the most important modules [12] [13]. In the feature extraction module, 7 sets of selective features are drawn out and used in the recognition system.

Image to Text Converter and Translator using Deep Learning and Image Processing

This helps in explaining the best idea from the word extraction with the help of description of character and stroke configuration, www context search and data mining with the use of semantic web and synaptic web at low entropy [18]. It gives us an algorithm for implementing the Optical Character Recognition (OCR) for translation of images of typed or characters written with hands into electronically editable format by saving the font properties. Optical Character Recognition can precisely do this by applying pattern matching. The text characters which are recognized are kept in editable format [25]. It shows a simple, efficient and minimum cost approach to make an Optical Character Recognition System for scanning any file that has a fixed font size and style or handwriting style. Using this, the machines have the ability to give excellent output. They are mostly used with the existing Optical Character Recognition methods, especially for English text. This shows that the problems the developers face while using the Optical Character Recognition as a technology on a large scale and give the solution to that problem [30]. OCR system provides different features which require no need to type, edit raw data, fast translation, and memory management. The objective of this paper is to make user-friendly machine which will extract texts and words from pdf and convert the extracted text into user friendly language then it will convert it into voice which describes the text more efficiently. It is a
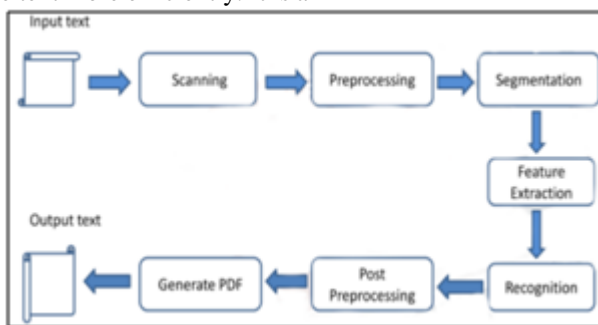

Fig 2.1 Flow chart of OCR technology

shared memory architecture that is used in parallel computers. In this model, single memory controller is used. In the architecture of the UMA model, access time to a memory location is independent of which processor makes the request [5] [6].

III. ARCHITECTURE OF IMAGE TO TEXT TRANSLATOR:

The Image to text converter and translator is a software that enables you to take an image as an input, identifies the text by performing contour extraction and we are using an inbuilt library called PyTessaract. To classify the text, we have used machine learning to create a model. Wikipedia is used to scrape the words as a key. So if we type a word and the word is present in the Wikipedia so all the words of that page will be extracted from the Wikipedia page and a dictionary will be created. When we visit a different country we face the language barrier A common problem faced by travelers is that of not understanding the local language. Failing to understand unknown languages when travelling can lead to some problems. The system which we proposed consists of two subsystems which perform text extraction

and text translation. There are lots of software packages or web services present for the extraction and translation part. The most challenging part is the extraction of the text and converting it to the given language.

Fig 3.1 Test Image used for Contour extraction

Fig 3.2 Contours around the text

The user clicks an image from his phone which is given as input. The clicked image is then pre-processed to remove any kind of noise present. The input image is then converted into a grayscale image which can then be converted into binary image. PyTessaract is an Optical Character Recognition engine for various operating systems. It is free software, released under the Apache License, Version 2.0, and development has been sponsored by Google since 2006. Tesseract is known as an open source OCR system currently available. The total count of support languages to over 60. To extract the text from an image, Tessaract is used. After extracting the words from image by using the OCR software, the input words are translated into the language. To perform this step, the Google Translator API service is used. This is a free service. It provides many libraries for translation. We need to understand; the translation of words is the transfer of meaning from one language to another. The efforts have been dedicated to extracting the words from a smartphone clicked image, and the output is to get as close to the image as possible. The last step is translating the extracted text using the translator. The extracted language is translated with the help of Google translator. After that the input and output words and texts are generated in PDF format.
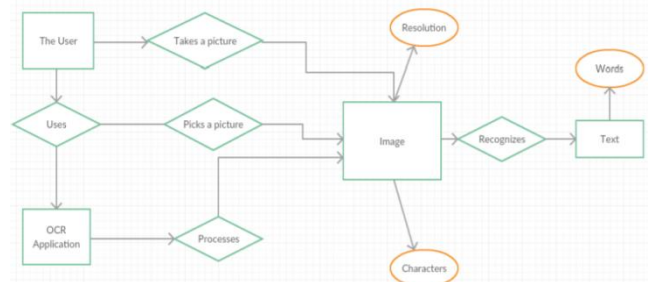

Fig 3.3 Flowchart of our proposed system

As shown in the picture above, these are the steps used by the model to extract the picture and processing it and displaying it.

IV. RESULT:

We have created an Image to Text converter and translator using Deep Learning and Image Processing. It solves a big problem that people face, not being able to understand a language when they see a text in some printed material or online.

If it is just text, it is easy to translate it online, but most of the times, this isn't the case. We face a lot of issues when we go abroad to a place or country where we are unfamiliar with the local language and seek help from people who may choose not to help us or even con us.

We have solved this issue by creating the web application. All it required is a picture of the text that we need to translate. It could be a screenshot, a snip from online articles or a photo from a pamphlet, poster, screens, paper etc. After uploading the picture, our system creates contours around the text in the picture and identifies the structure using PyTessaract. It is much more reliable than a normal OCR that sometimes fail to identify a text because of the issues like color, contrast, blurriness etc. We have fixed this issue by image processing using dilation and color correctness. This allows our system to correctly identify the text, be it small or big, italic or bold.

Along with the text identification, we have also added a classification of language. We have used Machine Learning to create a model that determines the percentage of words owing to one particular language in the text. For instance, a text can have words in both French and English. Our system identifies the correct language the words belong to and creates a probability distribution model which provides us with knowledge of the percentage of words in each language. We have scrapped data from Wikipedia, with certain words as the keys, and all the text that is available under the given word and created dictionaries with them. With every dictionary key, a unique value was assigned. This model was created from scratch and worked very well in determining the language. The output is displayed in a bar graph with x-axis as the language and Y-axis as the percentage.

We used 5 languages for the same: English, Czech, German, Swedish and French. After classification, we also implemented Translation of the identified text back to English, which is one very useful addition. The translation is done through Google Translate API available using googletrans library in Python. We displayed the translated text back to the user, under the original text. This is not only useful in translating the text, but it also helps the user to understand the language better.

Our system can be implemented into phones using a mobile application or on web using a website / web extension. We have created a website to showcase the system. The information that is displayed on the website consists of a graph, showing the probability distribution, the originally identified text and the translated text.

There are alternatives to our product that can do either of the two jobs: either identify the text or translate it. Our system combines both the concepts together and does the job well, making it very easy and effective for the end user to click a picture and translate the text into English. There are a lot of practical applications for the same and it is going to be useful to a large population of tourists as well.
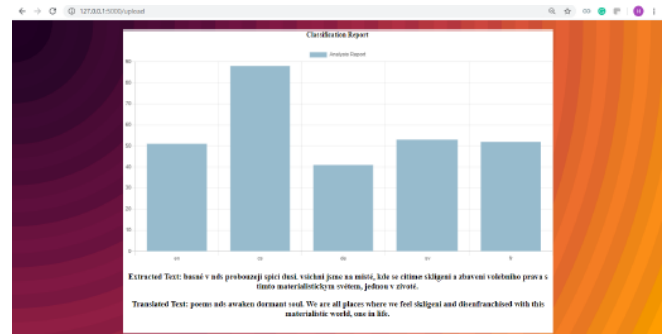

Fig 4.1 Result of a test image with Czech language

V. COMPARISON STUDY:

In this paper, the web app which we built makes the use of Wikipedia to make dictionary and extract the words. There are other apps out there which do the similar work like Google translate, Mobile OCR or simple OCR. But the one thing that they lack on how accurately the text has been detected. The implementation which we chose keeps in check all the details mentioned above. We scrapped data from Wikipedia, with certain words as the keys, and all the text that is available under the given word and created dictionaries with them. With every dictionary key, a unique value was assigned. This value was matched against the text extracted from the image in all 5 languages. With all the matches, a probability distribution graph was formed. Highest the value in the probability distribution graph, that was the language of the text. Some of the other work in the field uses just image capturing, text identification, language conversion and pdf generation and for image capturing they first have to feed the image into their system and then to the app. But in our project you can just click the image, you can take a snapshot, a snip from online article or a photo from a pamphlet, poster, paper, screen, etc. After uploading the picture, our system creates contours around the text in the picture and identifies the structure using PyTessaract. It is much more reliable than a normal OCR that sometimes fail to identify a text because of the issues like color, contrast, blurriness etc. We have fixed this issue by image processing using dilation and color correctness. This allows our system to correctly identify the text, be it small or big, italic or bold. Along with the text identification, we have also added a classification of language. We have used Machine Learning to create a model that determines the percentage of words owing to one particular language in the text. For instance, a text can have words in both French and English. Our system identifies the correct language the words belong to and creates a probability distribution model which provides us with knowledge of the percentage of words in each language. There are alternatives to our product that can do either of the two jobs: either identify the text or translate it. Our system combines both the concepts together and does the job well, making it very easy and effective for the end user to click a picture and translate the text into English. There are a lot of practical applications for the same and it is going to be useful to a large population of tourists as well.

VI.   CONCLUSION:

In this paper, we got to know about Image to Text converter and translator using deep Learning and Image Processing. It solves a big problem that people face, not being able to understand a language when they see a text in some printed material or online. If it is just text, it is easy to translate it online, but most of the times, this isn't the case. We face a lot of issues when we go abroad to a place or country where we are unfamiliar with the local language and seek help from people who may choose not to help us or even con us.

REFERENCES:

1. Kai Ding, Zhibin Liu, Lianwen Jin, Xinghua Zhu, "A Comparative study of GABOR feature and gradient feature for handwritten chinese character recognition", International Conference on Wavelet Analysis and Pattern Recognition, pp. 1182-1186, Beijing, China, 2-4 Nov. 2007.
2. Pranob K Charles, V.Harish, M.Swathi, CH. Deepthi, "A Review on the Various Techniques used for Optical Character Recognition", International Journal of Engineering Research and Applications, Vol. 2, Issue 1, pp. 659-662, Jan-Feb 2012.
3. Bhatia Neetu, "Optical Character Recognition Techniques", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 4, Issue 5, May 2014.
4. Liana M. Lorigo and Venu Govindaraju, "Offline Arabic Handwriting Recognition: A Survey", IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume 28 Issue 5, May 2006.
5. K. Gaurav and Bhatia P. K., "Analytical Review of Preprocessing Techniques for Offline Handwritten Character Recognition", 2nd International Conference on Emerging Trends in Engineering & Management, ICETEM, 2013.
6. Salvador España-Boquera, Maria J. C. B., Jorge G. M. and Francisco Z. M., "Improving Offline Handwritten Text Recognition with Hybrid HMM/ANN Models", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 33, No. 4, April 2011.
7. U. Pal, T. Wakabayashi and F. Kimura, "Handwritten numeral recognition of six popular scripts," Ninth International conference on Document Analysis and Recognition ICDAR 07, Vol.2, pp.749-753, 2007.
8. Anita Pal & Dayashankar Singh, "Handwritten English Character Recognition Using Neural," Network International Journal of Computer Science & Communication. Vol. 1, No. 2, July-December 2010, pp. 141-144.
9. J. Pradeep, E. Srinivasan and S. Himavathi, "Diagonal Based Feature Extraction For Handwritten Alphabets Recognition System Using Neural Network", International Journal of Computer Science & Information Technology (IJCSIT), Vol 3, No 1, Feb 2011.
10. Evaluation of model-based retrieval effectiveness with OCR by Julie Borsack, Kazem Tagva and A Condit.
11. Information security retrieval accuracy with simulated OCR output by W.B. kroft, S.M. harding, University of Nevada.
12. OCR Post-Processing Error Correction Algorithm using Google Online Spelling Suggestion Youssef Bassil, Mohammad Alwani.
13. Probabilistic retrieval of OCR degraded text using N-grams by S. M. HardingW. B. CroftC. Weir.
14. OCR and bar code reader with optimized sensor Medford D. SannerLynn D. McWaters.
15. Optical Character Recognition (OCR) for Printed Devnagari Script Using Artificial Neural Network by Raghuraj Singh, C. S. Yadav, Prabhat Verma.
16. Effects of OCR errors on ranking and feedback using the vector space model by KazemTaghva, JulieBorsack, AllenCondit.
17. Adapting the Tesseract open source OCR engine for multilingual OCR by Ray Wensley Smith profile image, Ray Smith.
18. Automated evaluation of OCR zoning by J. Kanai ; S.V. Rice ; T.A. Nartker ; G. Nagy
19. Ciresan, D.C., Meier, U., Gambardella, L.M., &
20. Schmidhuber, J, 2011, Convolutional neural network committees for handwritten character classification presented in International Conference on Document Analysis and Recognition, Beijing, China, 2011. USA: IEEE.
21. 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
22. A. Canedo-Rodriguez, S. Kim, J. Kim and Y. Blanco-Fernandez, 'English to Spanish translation of signboard images from mobile phone camera', IEEE Southeastcon 2009, 2009.
23. H. Nakajima, Y. Matsuo, M. Nagata and K. Saito, 'Portable Translator Capable of Recognizing Characters on Signboard and Menu Captured by Built-in Camera', in Proceedings of the ACL Interactive Poster and Demonstration Sessions, 2005.
24. S. Mori, C. Suen and K. Yamamoto, 'Historical review of OCR research and development',Proceedings of the IEEE, vol. 80, no. 7, pp. 1029- 1058, 1992.
25. M. Laine and O. Nevalainen, 'A Standalone OCR System for Mobile Cameraphones', 2006 IEEE 17th International Symposium on Personal, Indoor and Mobile Radio Communications, 2006.
26. A. Gohr, 'Linux OCR Software Comparison', Splitbrain.org, 2015. [Online]. Available: http://www.splitbrain.org/blog/2010-06/15-linux_ocr_software_comparison
27. Hymes, K. & Lewin, J. 'OCR for Mobile Phones', 2008
28. S. Fong, A. Elfaki, M. bin Md Johar and K. Aik, 'Mobile language translator', 2011 Malaysian Conference in Software Engineering, 2011.
29. V. Fragoso, S. Gauglitz, S. Zamora, J. Kleban and M. Turk, 'TranslatAR: A mobile augmented reality translator', 2011 IEEE Workshop on Applications of Computer Vision (WACV), 2011.
30. Koga M, R. Mine, T. Kameyama, T. Takahashi, M. Yamazaki and T. Yamaguchi, 'Camera-based Kanji OCR for mobile-phones: practical issues', Eighth International Conference on Document Analysis and Recognition (ICDAR'05), 2005.

AUTHORS PROFILE

Swapnil Sinha Currently pursuing B.Tech in Computer Science and Engineering from Vellore Institute of Technology, at School of Computer Science and Engineering (SCOPE), Vellore – 632014.

Harsh Kumar Kataruka Currently pursuing B.Tech in Computer Science and Engineering from Vellore Institute of Technology, at School of Computer Science and Engineering (SCOPE), Vellore – 632014.

Vijayakumar Kuppuswamy received his M.Tech. And Ph.D. Degree in 2006 and 2018 from VIT, Vellore, India. Research interests are Data Mining and Missing data. Working as Associate Professor, in Department of Information Security, at School of Computer Science and

Engineering (SCOPE), in Vellore Institute of Technology, Vellore-632014