International Journal of Scientific Research in Computer Science, Engineering and Information Technology
ISSN : 2456-3307 (www.ijsrcseit.com)
doi : https://doi.org/10.32628/CSEIT206412

Language Translation on Intelligent Navigation System using Image Processing

Anubhuti Rane[1], Sampada Gaonkar[1], Gauri Gulwane[1], Tamanna Kasliwal[1], Dr. Chaya Jadhav[2]
[1]Department of Computer Engineering, Dr. D. Y. Patil Institute of Technology, Pimpri, Pune, Maharashtra, India
[2]Professor, Department of Computer Engineering, Dr. D. Y. Patil Institute of Technology, Pimpri, Pune, Maharashtra, India

ABSTRACT

Article Info
Volume 6, Issue 4
Page Number : 38-47
Publication Issue :
July-August-2020

Visitors traveling to different countries around the world often find it hard to understand and communicate in local languages, because they don't understand it. They can't read the words written on the navigational boards or banners at these new locations. Text detection, extraction, and translation system must, therefore, be built to identify and recognize the text found on the navigation boards. This system proposes and implements a three-stage process that involves detection, extraction, and translation using the concepts of Convolutional Neural Network (CNN) and Long Short Term Memory networks or simply "LSTMs". The framework has been designed to take into account the need to create a desktop application that extracts the text from images based on traffic navigation boards and then translates it further into a user-understandable language. In this way, the user can grasp the unfamiliar language and roam freely in the unfamiliar terrains.

Article History
Accepted : 10 July 2020
Published : 15 July 2020

Keywords: Detect, Extract, Translate, Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), Long Short Term Memory networks (LSTMs).

I. INTRODUCTION

Traveling to new places always involves difficulty in navigating to the specified destination when the language used for communication isn't understandable to the person. Thus, this poses a retardant and limits the traveling experience. Also, the varied applications available wherein the user needs to type the text so as for it to be translated thereto user's language may be a tiresome job. Hence, this paper is proposing an implementation of a system that will easily detect and extract the text from images and further translate it to English easily as the user only needs to upload the image in the desktop application. This system helps to translate the text from the navigational board images from Spanish to English and from French to English. In artificial intelligence (AI) and Natural Language Processing (NLP), Machine Translation (MT) is a crucial research topic.

Along with the growing tourism, people find it difficult to grasp the language of the place they are traveling to. Since they are unable to interpret the words written on the navigational boards or banners, it causes hindrance in their enjoyment. Therefore, this system is developed which will extract the text from images uploaded by the user, furthermore, this extracted text would be translated to English, and help the users to know what exactly is written on the navigational boards. Additionally, text-based traffic signs in countries like Spain and France usually contain Spanish language and French language respectively. However, thus far there's no unified method to transform the text-based traffic signs in various languages to a user understandable language. Therefore, text-based traffic sign detection and translation remains a challenging task.

In recent years, with the continual success of deep neural networks in many fields, they have become the mainstream for several vision tasks. The proposed system aims to research a way to use deep learning tools to unravel the matter of text-based traffic sign detection, extraction, and translation. The goal is to accurately detect the texts in traffic signs with high efficiency, fully avoiding the influence of background texts and symbol-based traffic signs.

1.1 Objective:

The system proposes to solve the concerns faced by the tourists that have difficulty in understanding the local language of the place they are visiting. In this case, the original language is Spanish or French and the target language is English.

Following are the various objectives of the system:

1) To detect the text area in the image of text-based navigation boards by removing the non-textual areas.
2) To extract the words from the recognized text area.
3) To translate this extracted text into a user understandable language, that is English.

II. BASIC CONCEPTS

2.1 Deep Learning:

Deep Learning is a machine learning field of study concerned with algorithms inspired by brain functionality, called artificial neural networks. It imitates the human brain's workings in data processing and in creating patterns for use in decision making. Deep learning AI is capable of learning from the disorganized and unlabelled data. Taking inspiration from this concept, various sub-field concepts like Convolutional Neural Network, Deep Neural Network, Recurrent Neural Network, Long Short Term Memory can be used.

2.2 Machine Translation:

Machine Translation (MT) is a machine learning and deep learning sub-field which relies on the translation of text from one language to another. Neural Machine Translation (NMT) has emerged as the most powerful algorithm for carrying out this mission, with the help of deep learning.

2.3 Convolutional Neural Network

Convolutional Neural Network (CNN) facilitating the concept of deep learning is the key focus of Artificial Intelligence, which can acquire an input image, meaning (learnable weights and biases) for different aspects / objects in the image to recognize from one to the other. Region based convolutional neural networks or regions with CNN (R-CNN) characteristics are a highly innovative approach that applies deep models for object detection. An R-CNN model chooses several proposed regions and uses a CNN to perform forward calculation and extract the characteristics from each proposed field. It then uses those features to predict the proposed regions' divisions and boundary boxes.

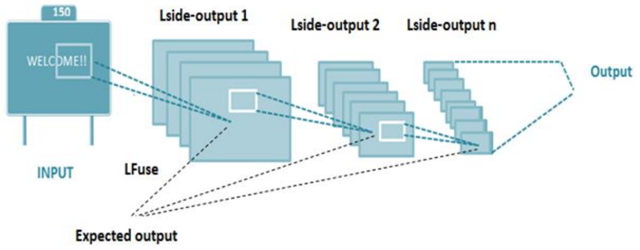


Figure 1. Convolution Neural Network in Edge Detection

The system uses CNN concept in edge detection for image processing. Canny edge detection is a multi - stage algorithm which can detect edges simultaneously with noise suppressed. Implanting the CNN concept into Canny Edge detection treats edge detection as a problem of signal processing. The key idea is that if you evaluate the intensity change on every pixel in an image, it's very high on the edges. Canny operator makes use of intermediate-layer side outputs. An edge point's location may be the numerical row and column indices of the pixel where the edge was observed, or the edge directional co-ordinates at sub-pixel resolution. An edge fragment may be defined as a small line component about the size of a pixel, or a point with an aspect of orientation. The result of earlier layers is called side output, and to generate the final predictions the output of all 5 convolution layers is fused. Since the characteristic maps generated at each layer are of different sizes, the image is effectively viewed at varied scales. Thus, the input system goes through the entire process in order generate the desired output using the concept of convolutional neural network for edge detection.

2.4 Recurrent Neural Network

Recurrent Neural Network (RNN) falls within a special category of neural network with loops that allow information to persist over various stages in a network. It can also be thought of as using the same network constantly, with each new addition the model has somewhat more knowledge than the previous one. While RNNs learn similarly during training, while generating output from prior data, they also remember the things learned. It's a part of the grid. RNNs can take one or more input vectors and generate one or more output vectors, and outputs are determined not only by weights applied to inputs such as a normal neural network, but also by a 'hidden' state vector representing the background on input and output. It can work on any Hierarchical tree structure. Unlike feedforward neural networks, RNNs can process input sequences using its internal state (memory). An RNN recalls every single detail over time. It's only useful to remember previous inputs in time series prediction, even because of the function. And this is known as short memory Long Term.

2.5 Long Short Term Memory

Recurrent neural networks are neural networks with loops in them, which allows the information to persist. It is like multiple copies of the same network, each copy passing a message to its successor. Long Short Term Memory networks or simply “LSTMs” are special kind of recurrent neural network which works, for many tasks, better than the standard version. They are the most powerful and well known subset of RNN. It can be said that LSTM networks are modified version of RNN which makes it easier to remember past data in memory. They are capable of learning long-term dependencies. They are a type of artificial neural network which are designed to recognize patterns in the sequences of data. They were introduced by Hochreiter & Schmidhuber in the year 1997. LSTMs also have a chain like structure, but each repeating module has a different structure. Instead of having one neural network layer, there are four, interacting in a special way. It trains the model by using back-propagation .
LSTM module has 3 gates named as Forget gate, Input gate, and Output gate.

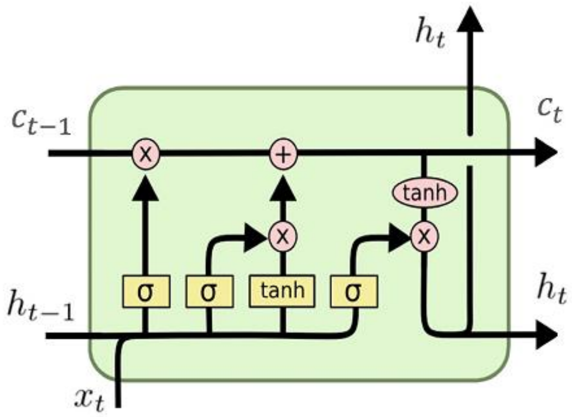


Figure 2. Long Short Term Memory [10]

Here,

- ct-1 : input from a memory cell in time point t;
- xt : an input in time point t;
- ht : an output in time point t that goes to both the output layer and the hidden layer in the next time point.

Every block has three inputs namely, xt, ht-1, and ct-1 and two outputs which are : ht and ct. All these inputs and outputs aren't single values, but are vectors with lots of values behind each of them.

Forget Gate:
Decides how much of the past it should remember. This gate decides which information is to be omitted from the cell within that particular time stamp. It is determined by the sigmoid function.

Update Gate/input gate:
It determines how much of this unit is added to the current state. Sigmoid function determines which values to let through: 0,1. The tanh function allots weightage to the values which are passed deciding their level of importance which ranges from-1 to 1.

Output Gate:
Decides which part of the present cell makes it to the output cell. Sigmoid function decides which values to let pass through 0,1. The tanh function gives weightage to values which are passed to determine their level of importance ranging from-1 to 1. .

LSTMs help preserve the error which will be back propagated through time and layers. By keeping a more constant error, they permit recurrent nets to still learn over many time steps,thus causing the opening of a channel to link causes and effects remotely. Since the gates can prevent the remaining networks from changing the contents of the memory cells for multiple time steps, LSTM networks save the signals and propagate errors for much longer than an ordinary RNN. By independently reading, writing and deleting the content from the memory cells, these gates also learn to attend to specific parts of the input signals and ignore the other parts.

III. LITERATURE SURVEY

Fares Aqlan et al. [1] proposes a methodology for translating complex languages such as Arabic into Chinese, and vice versa, which, due to a large number of rare words, is a difficult task. The byte-pair encoding (BPE) and Neural Machine Translation (NMT) techniques are used where the rare words can be effectively encoded as sub-word sequences that include Romanization. The method also provides a qualitative analysis of the results of a translation. It also compares the effect of different segmentation strategies on the Arabic-Chinese and Chinese-Arabic NMT framework, while proposing common requirements for Chinese-Arabic parallel corpus data filtration. When adopting other languages, the results of the translation can be more accurate and efficient to improve the accuracy of segmentation between Romanized Arabic and those languages.

Yongchao Xu et al. [2] presents the TextField detector used to detect unusual scene texts. A direction field is learned and this direction field text is fully detected using the Cascaded Convolution Network (CNN). Post-processing based on morphology is applied to the learning direction for final detection. This system provides grouping of the regions of the text and improves performance and efficiency. Traditional segmentation is challenging than this method as they hardly separate adjacent instances of text. For some difficult images, including object occlusion, spacing of large characters, it still fails. TextField also has erroneous detections on some text-like regions.

Muhammad A.Panhwar et al.[3] presents that learning machines and recognition of patterns play a vital role in extracting information from natural scenes. They introduced a framework to extract the text from natural scenes and landscapes. Image capture is the very first step towards identifying signboards. The next step is detection of the real-time signboard followed by detection and acknowledgement of the text. The image text (pixel-based text) is converted, in recognition, to a readable and editable form. They randomly conducted experiments on 500 images from natural scenes. For the recognition of these images Neural Network was chosen. The program performs English- and Urdu-language recognition. Up to 85 per cent accuracy was achieved in signboard detection.

Tiejun Zhao et al. [4] Introduces source-dependence context representation for prediction of translation. This approach to the neural network is for encoding the bilingual meaning. It is not only capable of encoding long-distance source dependencies but also capturing functional similarities to help predict translations. This method has the potential to significantly improve SMT performance over strong baseline methods and has verified that structural clues in the context are beneficial for translation prediction.

Yingying Zhu et al. [5]. Traffic based signs consist primarily of traffic signs or text based navigation boards. To locate the signs from the images, the detection process is a two-stage detection method that reduces the text detection search area and removes text from outside the traffic signals. The text-based traffic sign language is educated in English and Chinese based on a public dataset and a self-collected dataset. This system utilizes a fully convolutionary neural network to train the images. The proposed application improves the speed of detection and solves the multi-scale problem for text detection. Detection of traffic signs and detection of text are two different object detection problems, detection of two different objects in a unified framework is not reasonable. Also, it is difficult for TextBoxes to detect the texts because they are too small in the entire images.

ZhaorongZong et al. [6] Presented that the source language is distorted as it travels through a noisy channel and emerges at the other end of the channel as Target. The role of the program of statistical machine translation is to find the highest likelihood of the sentence as a result of the translation. Neural machine translation replaced computer translation by statistical translation. The idea of end-to - end neural machine translation is to implement automatic translation directly through neural networks between the natural languages. Neural machine translation usually uses an encoder-decoder framework for this, and achieves conversion from sequence to sequence. After that, the target language uses another recursive neural network to reverse decode the sentence vector in the source language to generate the target language. All of this decoding process is created word by word. Hence, the decoder can be considered as a language model containing the source language knowledge target language.

Therefore, the attention mechanism-based encoder-decoder model changes the way information is transmitted and can dynamically calculate the most relevant context in order to better solve long distance information transmission issues and improve the performance of neural machine translation.

Kehai Chen et al. [7] Presenting the sentence-level context as latent topic representations through the use of a neural convolution network (CNN) and modeling topic focus to incorporate source-sentence-level topic context knowledge into both attention-based and transformer-based NMT. This method can improve NMT 's performance by jointly shaping source topics and translations. It is a variant of CNN which captures information about the source topic based on the context at the sentence level. The proposed CNN maps the source subject knowledge into the subject vectors implicitly, and named the latent topic representations (LTRs).

Youbao Tang et al. [8] Presents text detection and segmentation via Cascaded Network Convolution (CNN). Extraction model for the candidate text region (CTR) is generated using the edges and the entire region. The CTR is then transformed to refined CTRs, then to text. The refined CTR is classified using the CTR classification model (Cnet), based on CNN. This system generates more accurate text region which has little background noise than traditional techniques. To train the models this system needs the decisive information of the edges of text and regions. But the publicly accessible datasets containing this critical information are too limited to train these models effectively, therefore it is difficult to train these datasets.

Jack Greenhalgh et al. [9] Under Open Source Computer Vision (OpenCV), presented a method for detecting and recognizing traffic signs running at a frame rate of 14 frames/s on a 3.33-GHz Intel Core i5 CPU. By this approach, considerable speed increase was obtained by running the algorithm as a pipeline in parallel. Their system they propose consists of two main stages: detection and recognition. Detection consists of three phases: search area determination, detection of large numbers of text-based traffic sign candidates using simple color information and shape, and reduction of candidates using contextual constraints. This over detection is important to ensure that there are no missing true positives (TPs). Potential traffic sign candidate regions are then only found within the scene search regions, using a combination of MSERs and color thresholding for hue , saturation, and meaning (HSV). Then this large number of identified candidate regions are reduced to exclude unlikely candidates by making use of the scene structure and temporal information. Recognition is the next step. Candidate text character components are then located within the region and sorted into potential text lines before being interpreted using an optical character recognition (OCR) package. The collection of identified lines of text (in grayscale) are transmitted for identification to the open-source OCR engine "Tesseract." To improve OCR's accuracy, the tests are averaged across several frames to improve identification accuracy.

IV. SYSTEM ARCHITECTURE

4.1. Text Detection and Extraction Module

The motive is to perform text detection and extraction on a text-based navigational board based image so as to translate the meaning of the generated text into the desired language. In the first step, the detection of edges based on shapes like square, rectangle, pentagonal, triangle and circular is done that potentially contain text. In the second step, the system performs text extraction, where, for each of the detected regions, a Convolutional Neural Network (CNN) and the Long Short-Term Memory (LSTM) (detailed explanation in section 3. BASIC

CONCEPTS) is used to recognize and decipher the word in the region.

The module can further divided into:

a) Detection of text areas from non-text areas in images.
b) Extraction of the text area.
c) Display of the detected text in standard format.

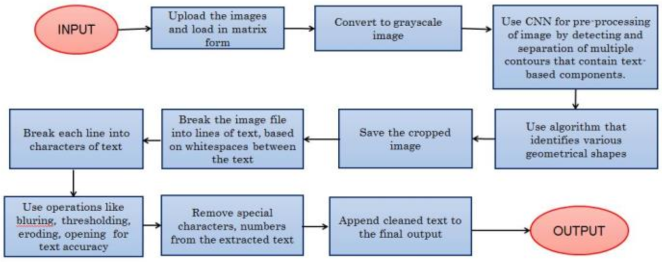

INPUT
Upload the images and load in matrix form
Convert to grayscale image
Use CNN for pre-processing of image by detecting and separation of multiple contours that contain text-based components.
Use algorithm that identifies various geometrical shapes
Save the cropped image
Break the image file into lines of text, based on whitespaces between the text
Break each line into characters of text
Use operations like bluring, thresholding, eroding, opening for text accuracy
Remove special characters, numbers from the extracted text
Append cleaned text to the final output
OUTPUT

Figure 3. Text Detection and Extraction Architecture

a) Detection of Text Areas from Non-Text Areas in Images:
The images obtained in this system comprise of traffic navigational boards. Thus main aim is to obtain the text from these images. The entire detection system is jointly done in a controlled, end-to-end manner. The users can upload the images that they want to be translated through the user interface provided by the application i.e- through the Django environment. Use of a scheduling of learning rates is done, starting with a very low learning rate to ensure that the model does not diverge, and progressively increasing the learning rate during the first few epochs to ensure that the model reaches a nice, stable point.

Grayscale images mainly consist of only grey tones of colour, which are of 256 steps. There are, in other words, only 256 grey colours. The main feature of grayscale images is the intensity of red, green , and blue rates. The color code should be as RGB(R,R,R), RGB(G,G,G), or RGB(B,B,B) where 'R,G,B' is a single digit of 0 to 255.

As a result, the image is composed of a number of pixels, and these pixel values can differ from one range to another.

1) The pixel value for binary image should be either 1 or 0 which means only two shades 1=white and 0=black.
2) In the case of gray image scale, for example 8-bit gray image scale (2 ^ 8=256) pixel value can vary from 0 to 256. Here the image is 256 shades (0=black, 1=white, and for others the combination of both).

Therefore, the gray scale image can have shades of gray that differ between black and white, while the binary image can either be white or black at two extremes for a pixel value. This makes it easy to remove noise in images while preserving edges for the grayscale images.

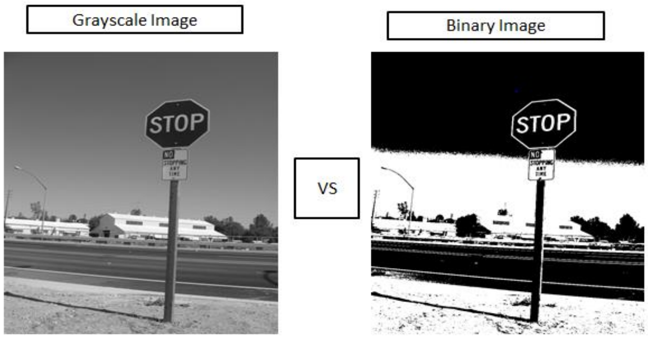


Figure 4. : Grayscaling vs Binary Image

This detection process is carried out using the Convolutional Neural Network, where each algorithm that can take an input image attributes importance to various aspects / objects in the image and can distinguish between them for edge detection. For CNN, the pre-processing level is much lower compared with other forms of classification. This algorithm is best used in the identification of artifacts to distinguish patterns, edges, contours and other variables such as colour, strength, shapes and textures.

b) Extraction of the Text Area:

Textboxes are defined as areas of importance where detections from all the irrelevant sections of the picture are the text field. The textboxes can also treated as regions that are essential for this type of model, since determining the location of multiple objects and shapes such as square, rectangle, pentagon, triangle and circles. After detections of regions in the image, the following steps are done

1) Dividing each line into characters, built based on whitespace between the characters.
2) Divide the document into lines of text, built based on whitespace between the lines.

c) Display of the Detected Text in Standard Format

1) Determine the most closely matching character from the model for each character extracted from the Text field.
2) Append the identified character in the output text.
3) Display the generated text in the standard format.

4.2 Text Translation Module

The Stage I process in translation depicts the pre-processing steps in which the Spanish, French and English datasets are imported through library setup. The pre-processing stage consists of cleaning text to get accuracy by eliminating noise. The steps like removing unprintable characters and symbol-based characters such as punctuation marks. To normalize the Unicode characters to ASCII (American Standard Code for Information Interchange) and finally remove non-alphabetic tokens. The final output is a text free of noise used for the translation process

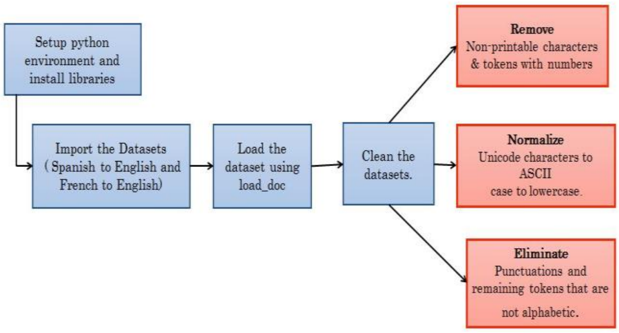

Setup python environment and install libraries
Import the Datasets (Spanish to English and French to English)
Load the dataset using load_doc
Clean the datasets.
Remove Non-printable characters & tokens with numbers
Normalize Unicode characters to ASCII case to lowercase.
Eliminate Punctuations and remaining tokens that are not alphabetic.

Figure 5. Stage I of Text Translation Process

The Stage II translation process is based upon model training and testing. The datasets are split into train and test sets according to the necessary criteria. The model uses the tokenization principle, whereby the specified terms are attached with an unique numerical value that remains constant throughout. The model depends primarily on the Recurrent Neural Network (RNN) and the Long Short-Term Memory (LSTM) mechanism (detailed description in section 3. BASIC CONCEPTS). The trained model acts on the system test set or given input where it finally translates the text from English to Spanish or from French to English.

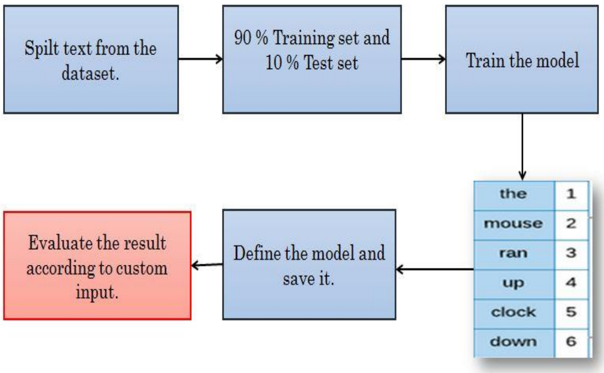

Spilt text from the dataset.
90 % Training set and 10 % Test set
Train the model
the 1
mouse 2
ran 3
up 4
clock 5
down 6
Define the model and save it.
Evaluate the result according to custom input.

Figure 6. Stage II of Text Translation Process

4.3 Django Architecture

Django follows a Model-View-Controller(MVC) architecture, which is divided into three different parts:

- The Model is the logical and functional part which is represented by a database
- The View is the user interface, that is, what you see in the browser.
- The Controller is the middleman that connects the view and model together, that is, passing the data from model to view.

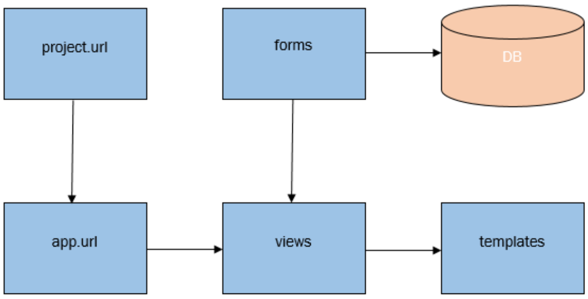
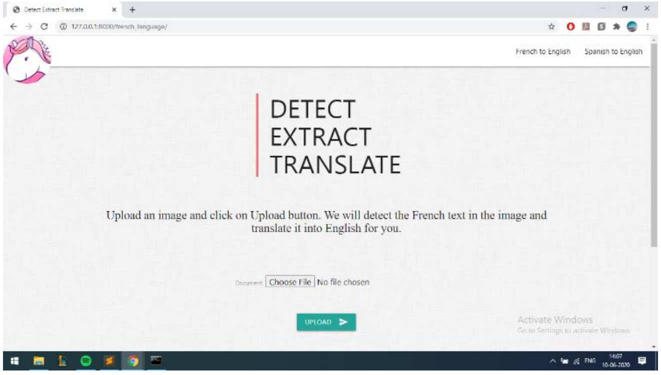


Figure 7. Interaction between different parts of Django framework

When the URL is entered in the browser, it passes its web application URL. The URL calls the function in the view file. The function renders html file. The form manages all the control such as text, buttons, etc. When the form is created, database is created too.

V. RESULTS AND DISCUSSION

Django Framework has been used for creating a web application based on the system. The user uploads an image, the system process the image and gives result, that is, it detects Spanish/French text in the image, extract the text and further translate it into English.



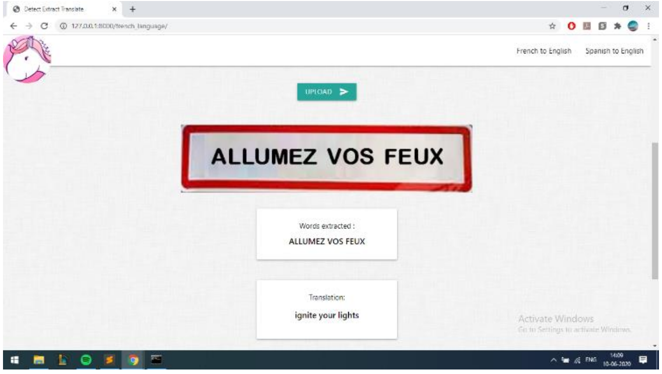
DETECT
EXTRACT
TRANSLATE

Upload an image and click on Upload button. We will detect the French text in the image and translate it into English for you.

Document Choose File No file chosen

Figure 8. User Interface

The User Interface consists of the upload file button and submit button. The user uploads an image from the local computer and submit it.



Words extracted :
ALLUMEZ VOS FEUX

Translation:
ignite your lights

Figure 9. Result

The result gets displayed on the web page; it consists of the image that has Spanish/French text, extracted text and the translation of the extracted text.

VI. CONCLUSION

The system implemented a technique with the Convolution Neural Network and Long Short Term Memory to produce the best translated text after extracting it from the images the device has identified. Therefore, the above features can be overcome to escape the difficulties faced by the travellers for not knowing the language. The system created on the basis of text detection and translation concepts helps the users navigate seamlessly. The CNN-based navigational board detection stage greatly reduces the text detection search area and greatly improves the detection effectiveness. In addition, the direct detection of text on the approximate navigation board can greatly alleviate the problem of huge variation in text scale thanks to the relatively fixed scale ratio between texts and navigation board. In the meantime, this can also reduce the number of fake optimistic texts not in the navigation board. The experimental results show that the implemented method is powerful and has been reliable and applied easily to the Spanish and French language text-based navigation boards.

VII. REFERENCES

[1] Fares Aqlan , Xiaoping Fan , Abdullah Alqwbani, and Akram Al-Mansoub. “Arabic-Chinese Neural Machine Translation: Romanized Arabic As Subword Unit For Arabic-Sourced Translation” IEEE Access, 2019

[2] Yongchao Xu, Yukang Wang, Wei Zhou, Yongpan Wang, Zhibo Yang, Xiang Bai. “TextField: Learning A Deep Direction Field for Irregular Scene Text Detection” IEEE Transaction on Image Processing, 2019.

[3] Muhammad A.Panhwar Kamran A. Memon,SaleemullahMemonSijjad A. Khuhro. “Signboard Detection and Text Recognition Using Artificial Neural Networks Signboard Detection and Text Recognition Using Artificial Neural Networks” IEEE Transcation on Image Processing, 2019.

[4] Kehai Chen, Rui Wang, Masao Utiyama, EiichiroSumita, and Tiejun Zhao. “Neural Machine Translation with Sentence-level Topic Context” IEEE/ACM Transactions On Audio, Speech, And Language Processing, 2019.

[5] YingyingZhu ,Minghui Liao , Mingkun Yang , and Wenyu Liu. “Cascaded- Segmentation-Detection Networks for Text-Based Traffic Sign Detection” IEEE Transactions On Intelligent Transportation Systems, 2018.

[6] ZhaorongZong, Changchun Hong. “On Application of Natural Procesing in Machine Translation” 3rd International Conference on Mechanical Controland Computer Engineering, 2018.

[7] Kehai Chen , Tiejun Zhao, Muyun Yang, Lemao Liu , Akihiro Tamura , Rui Wang , Masao Utiyama,and EiichiroSumita. “A Neural Approach to Source Dependence Based Context Model for Statistical Machine Translation” IEEE/ACM Transactions On Audio, Speech, And Language Processing,2018.

[8] Youbao Tang and Xiangqian Wu, “Scene Text Detection and Segmentation Based on Cascaded Convolution Neural Networks” IEEE Transaction on Image Processing, 2017.

[9] Jack Greenhalgh, Majid Mirmehdi. “Recognizing Text-Based Traffic Signs” IEEE Transaction on Intelligent Transportation Systems, 2014.

[10] Fig 2 : Image source : MingxianLin (https://commons.wikimedia.org/wiki/File:LSTM.png), https://creativecommons.org/licenses/by-sa/4.0/legalcode

Cite this article as :

Anubhuti Rane, Sampada Gaonkar, Gauri Gulwane, Tamanna Kasliwal, Dr. Chaya Jadhav, "Language Translation on Intelligent Navigation System using Image Processing", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 6 Issue 4, pp. 38-47, July-August 2020. Available at doi : https://doi.org/10.32628/CSEIT206412
Journal URL : http://ijsrcseit.com/CSEIT206412