Hindi Comic Book Character Extraction

Student
1. Isankumar Fulia
2. Samyukta Bharadwaj
3. Yogesh Bhati

Abstract
Comics are one of the most fun and enjoyable means of entertainment. They are widely read all over the world by children as well as adults. Published in many countries in different languages, each comic depicts its own country's culture and tradition alongside a unique storyline. In Indian culture, one of the most famous are the Hindu comic books. Hindu comics ranging from ancient gods to superhero comics to funny children's comics, these are widely read in India. Although, some of them are in English, many are published in Hindi. Their popularity has increased and now they have gone worldwide. These comics are not understood by all. Everyone irrespective of different language and culture must be allowed to enjoy the right to read comics of any other language, learn and know more about the culture. Hence, there should be a way to translate the Hindi text into, the most common and widely read languages, English. That's where the need for language character extraction, recognition and translation from an image arises.

The work done in this project includes extracting hindi text from hindi comic page. This is done by achieving various sub goals of extracting panel or frame from an image, extracting speech balloons from panel and then extracting the text from the balloons with the help of various image processing techniques.

Previous Work
In the past, many techniques on comic character extraction and recognition have been published. Most of these works start off by detecting and extracting parts of the comic page and then working on them separately for further analysis until text is extracted. Most of the previous works published were to detect Japanese or English text.

The Arai[1] method extracted the frame as a single blob object, where a blob is defined as all connected white pixels. Balloons are also extracted as an area of connected white pixels. Arai[1]'s approach makes use of a fixed threshold and assumes that the comic balloon is always made up of white pixels or very light colored pixels. Also balloon detection is done with the presence of two vertical lines with minimum length and fixed width to length ratio. This was the method followed for Japanese Manga comic.

Whereas, Ho[2] proposed a method for frame and balloon detection by using region growing and connected components extraction technique. Balloons are detected by dilating each of the candidate component obtained so as to connect the components within them and then the small connected components are discarded and the bigger ones are considered as speech balloons containing the text.

Introduction

The step prior to translation is text extraction. Only if the words are extracted can they be sent through to the pre-existing translator. Translation can be done automatically using online translation APIs like Google or Bing. To extract the text, the first part is to detect the frames.

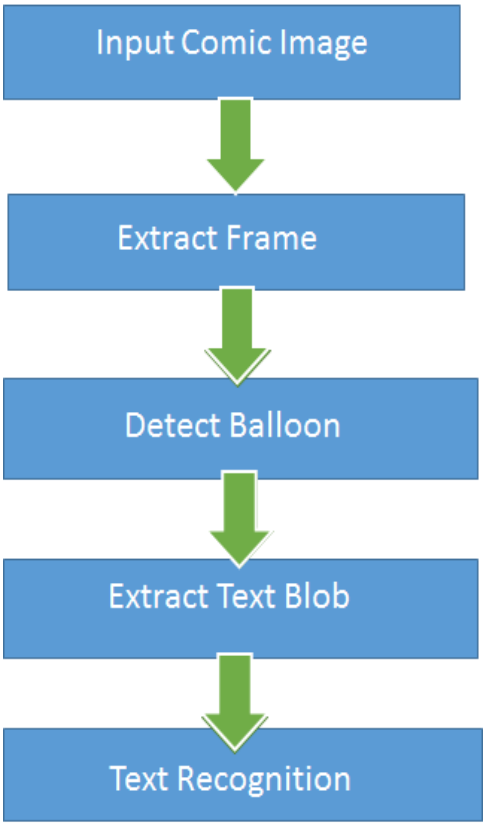
As initial step several methods were tried out. Arai[1] considered taking a fixed threshold. But when the same was tried for our approach, it didn't work on all images. Furthermore, edge detection filter, Canny, was applied that didn't work either. Finally the method proposed by Ho[2], based on region growing worked followed by morphological operations to break the link between the connected frames. Then the connected component and morphology was used so as to extract the speech balloons and text and remove the unwanted components.

The concepts learnt in class for segmenting an image by using Color property created a base for using region growing. Furthermore, connected components and regionprops property of connected components was used so as to get the area, filled image and bounding box of connected component.

In the further sections, we will be discussing in detail about Frame detection and extraction. This is followed by comic balloon extraction and finally the text extraction and noise removal.

Our Approach

Initially, all color images are converted to grayscale images. As stated by Arai[1], it is assumed that text is situated only inside the comic balloon which is located in the comic frame and hence, the frame and balloon have to be detected first.



Flow Diagram

Frame Detection

Frame detection is an essential phase as frames help maintain the order of text. The detected frames can be extracted and worked upon for further content extraction.

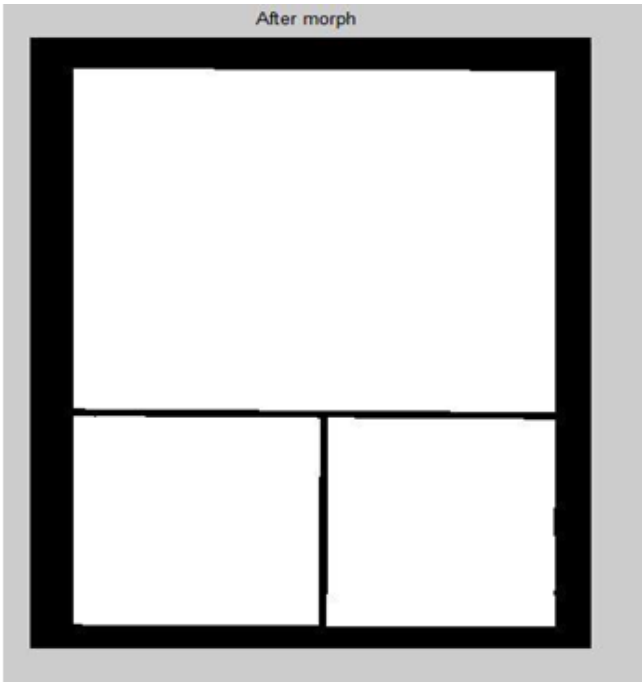
A region based algorithm is used to detect the frames where a seed point is taken as a region and the pixels around it are added to the region, if they are similar. A 5 pixel frame around the image boundary is considered by taking 5 columns and 5 rows from left to right and top to bottom of the image respectively. The average of this frame is calculated. Four points are selected as seed points at top, bottom, left and right corner of the image respectively. Each seed point is then used to compare the pixels around it with the average of the 5 pixel frame. If the pixels are similar to this average, they are added to the region. When the region growing algorithm stops, the output is the binary image with frames in black and background in white.

Also, the minimum width of frame should be at least 1/6 of the image width and height should be at least 1/8 of the image height. All those frames whose area pixels are less than the minimum area of frame are removed using the binary area open function. Furthermore,

any holes or noise remaining are filled.

Morphology is then used so as to remove the link between the connected frames. The image is dilated N times followed by same number of erosions. N is decided based on the image width. The algorithm only works for those images which have gutter in it i.e. the frames are separated by some pixels of background.

Figure : Frames detected



Once frames are detected they have to be extracted and then worked upon for further analysis. This is done by using Connected Component Labelling algorithm. First connected components are found in the page. Then the bounding box property of regionprops is used so as to get the bounding box across the component which is then cropped from the image. This is done exhaustively for all the connected components in the image and the frames are extracted.

The RGB frames extracted are then converted to HSV color space. The processing further continues with the value component of the image.

Figure :HSV image

Hue Saturation Value

Balloon Detection

Once the frames are extracted, the comic balloon needs to be extracted. The balloon area called the candidate object is extracted by making use of Connected Components algorithm yet again.

Ho[2] proposed that the candidate area can be selected by retaining only the regions where ratio between pixels of the candidate area and the bounding box are higher than 60% . Our approach is slightly different. We have considered a heuristic value of 0.01, and if ratio of the area of the connected components to that of the entire area of the frame is greater than this value, then that connected component is considered further and the rest are considered as noise and discarded. The problem with bounding box is that it may also include other parts of image which are not part of bounding box as it is rectangle and connected component may not always be rectangle. The image obtained from the FilledImage property of regionprops contains the actual connected component. It is multiplied with the above thresholded image so as to get rid of part which is not the part of connected component.

There may be a lot of connected components identified which may not be the balloon. These components need to be removed. The idea is to dilate each component so as to join the text together and the ones with larger number of white pixels compared to black are chosen as candidate balloons i.e. the large connected components is considered as balloon and the small connected components are considered as noise and discarded. To

dilate the image N times, N is decided based on the component width and height

.

Figure: Candidate objects obtained as balloon along with noise.

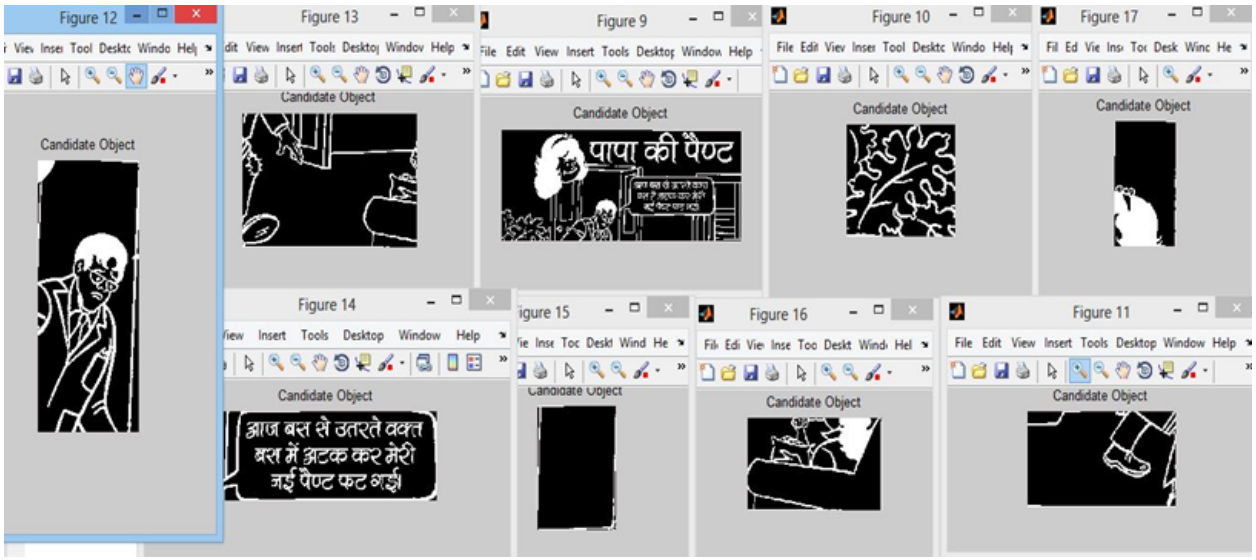


Figure: Dilation of candidate objects

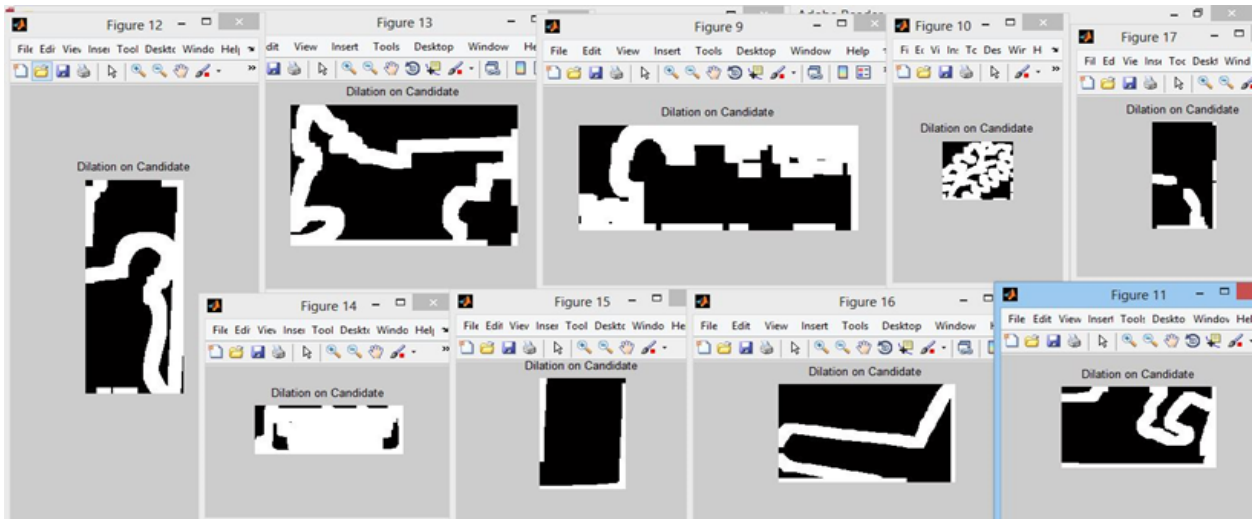
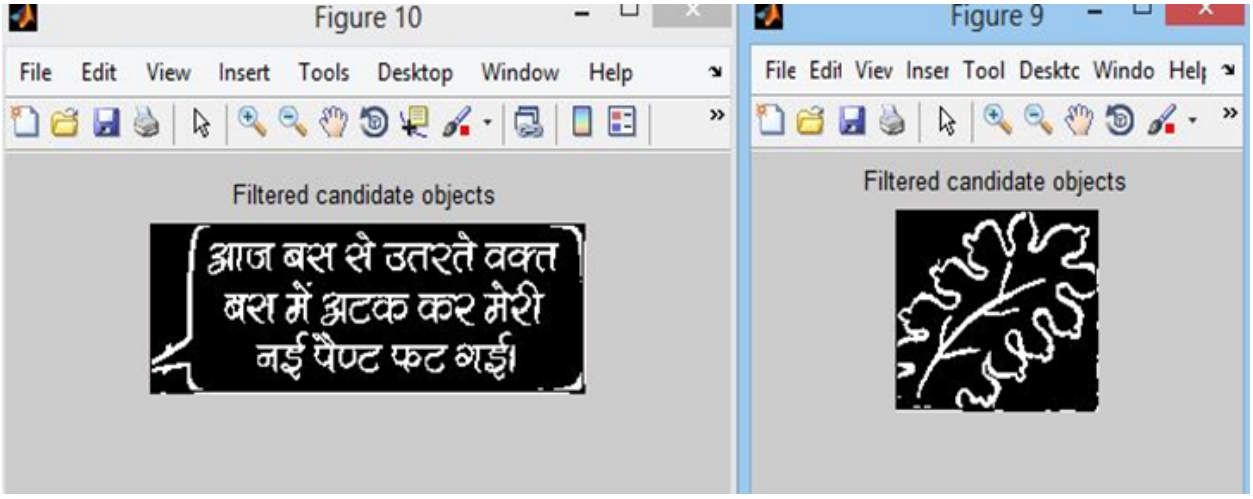


Figure: Filtered candidate objects after noise removal



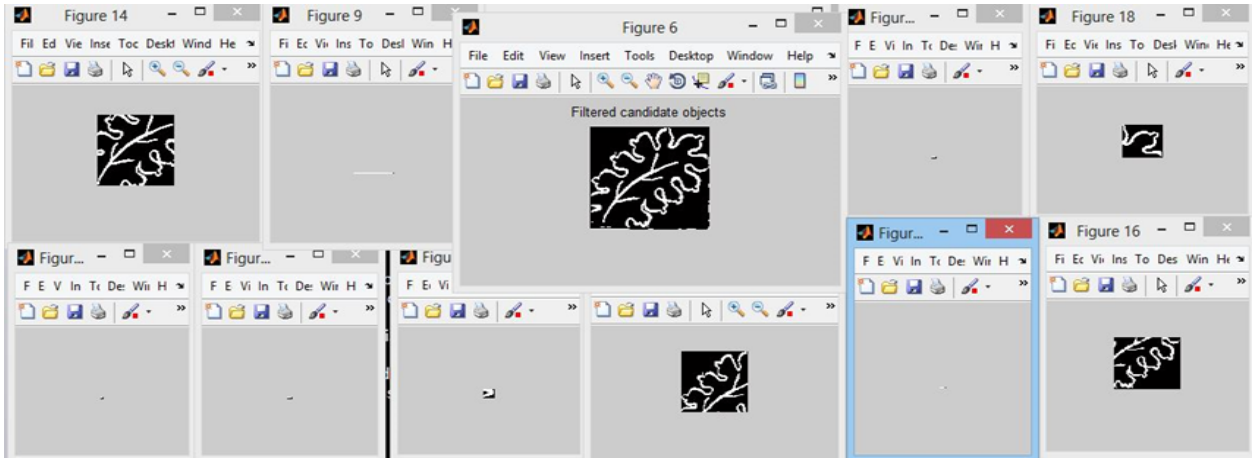
Text Detection

Once the candidate area is extracted, our next task is to determine that the detected balloon contains text or not. We consider that the structure of text is regular connected component whereas noise is considered to have irregular structure. To extract the text connected component is applied on filtered candidate component to get the connected text as separate word. Now, using the imcrop and boundingbox property of regionprops each of the word from the text is cropped.

Removal of noise components

Here again apart from text, a lot of noise is obtained as a part of connected component which needs to be removed. Based on the size of the component and the ratio of number of black pixels to white pixels, we decide whether the component is a text or noise.

Figure : Text extracted with noise for the two candidate balloon components



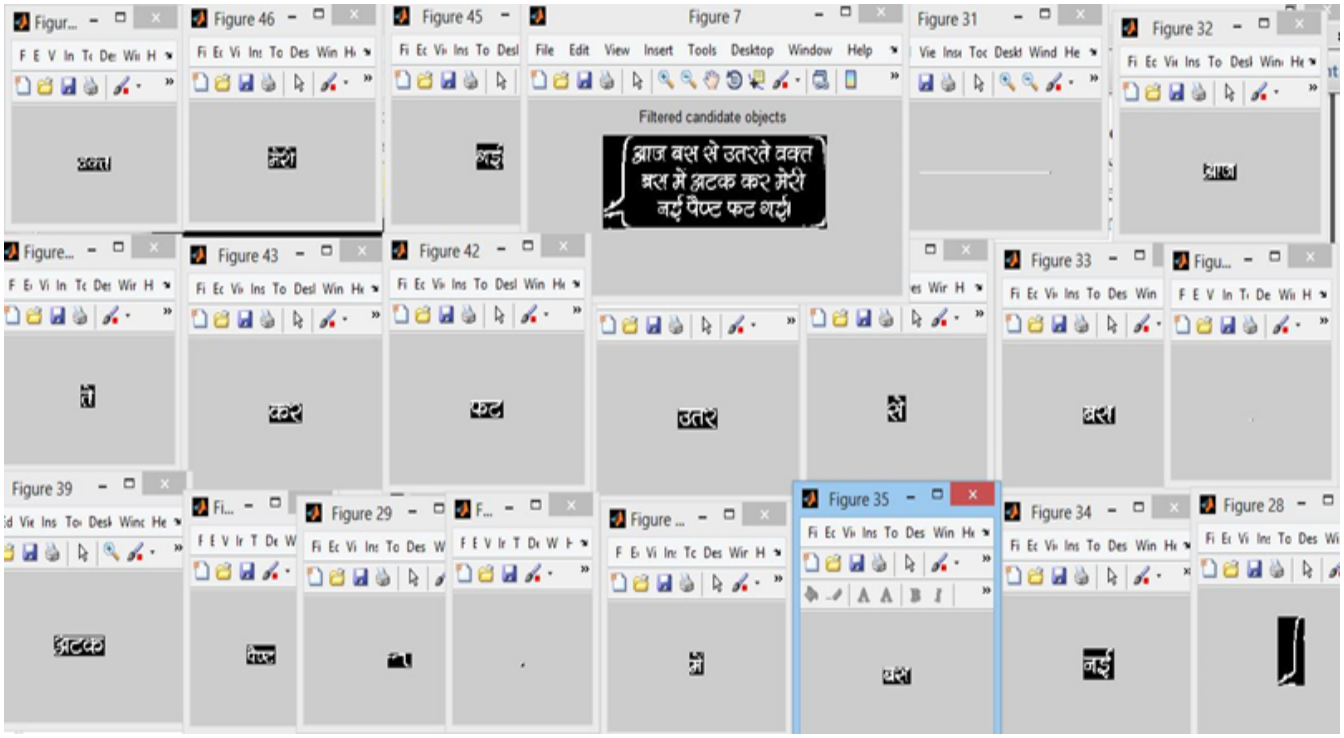
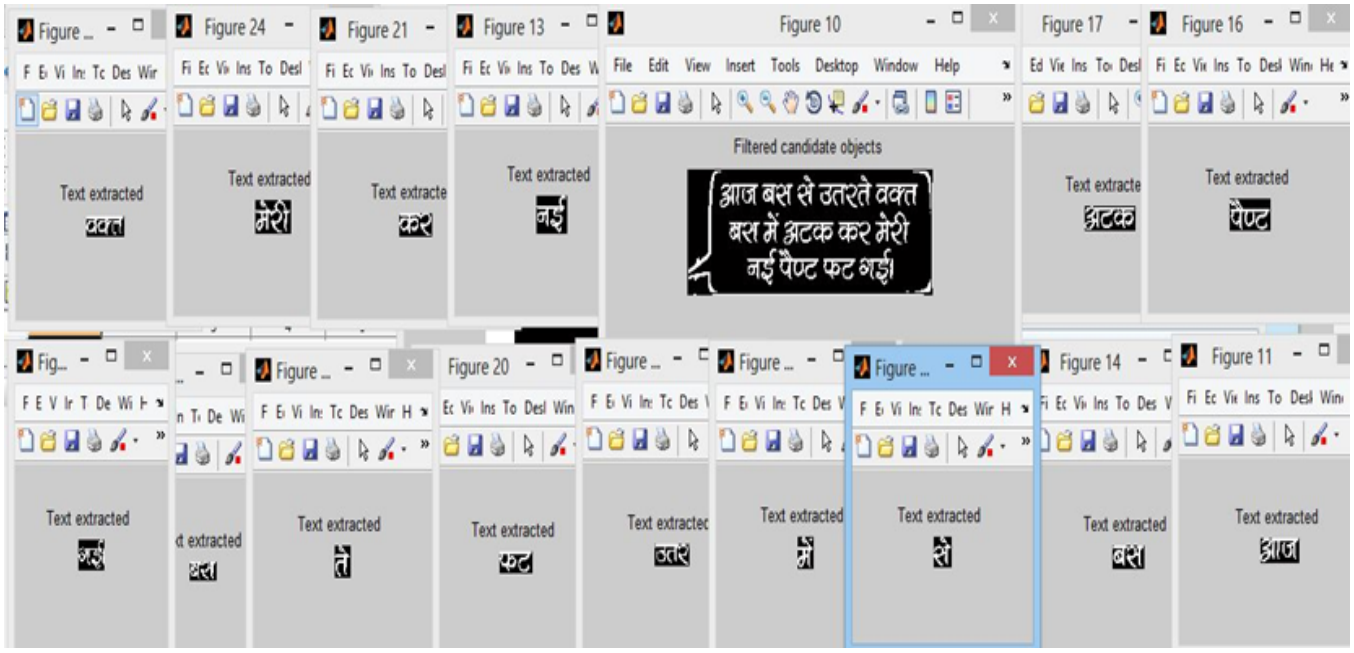


Figure: Text after filtering the noise



As we can see, most of the noise component has been removed from the text and we are able to extract each character word in the balloon. Also the other candidate balloon which did not consist of any text was completely filtered out.

Result

We have applied the algorithm on around 20 comic pages and were able to successfully extract text from almost all the images. Those images which do not contain gutter failed in detecting the frames. But even for them, the text is extracted from the balloon by considering the entire image as a single frame.

Future Work

Our approach can be further carried forward and made into more robust detection algorithm that is not dependent on the gutter between the frames. A great deal of work needs to be done in optical character recognition as Hindi language OCR's are not much developed. If this is achieved, our application, upon further improvisation can help build a mobile or an online application for comic character detection and translation.

References

1. Kohei Arai and Herman Tolle ,"Method for Real Time Text Extraction of Digital Manga Comic", International Journal of Image Processing (IJIP), vol.4, Issue (6), pp 669676, 2010.

2. A.K.N. Ho, J.C. Burie and J.M. Ogier, "Panel and Speech Balloon Extraction from Comic Books," 2012 10th IAPR International Workshop on Document Analysis Systems, pp. 424428, Mar. 2012.

3. Christophe Rigaud, Norbert Tsopze, Jean-Christophe Burie and Jean-Marc Ogier ," Robust Frame and Text Extraction from Comic Books", 9th International Workshop, GREC 2011, Seoul, Korea, September 1516, 2011, Revised Selected Papers

4. Digital Image Processing USING MATLAB Gonzalez, Woods and Eddins
Second Edition - Gatesmark Publishing (c)2009