# Identifying Text in Images Using OCR Testing

**M.JESLIN BENITA PONNARASI**
**PG STUDENT M.TECH (ISCF)**
**DR.MGR EDUCATIONAL & RESEARCH INSTITUTE UNIVERSITY**
**CHENNAI**

## 1. SYNOPSIS

There are many applications in which the automatic detection and recognition of text embedded in images is useful .These applications include multimedia systems, digital libraries and Geographical information systems. When machine generated text is printed against clean backgrounds .it can be converted to a computer readable form(ASCII) using current Optical Character Recognition (OCR) technology. However, text is often printed against shaded or textured backgrounds or is embedded in images.

A system that automatically extracts and detect text in images is proposed .This system consist of four phases .First, by treating the text as distinctive texture, a texture segmentation sheme is used to focus attention on regions where it may occur. Second, strokes are extracted from the segmented text regions. Using reasonable heuristics on text strings, such as height similarity ,spacing and alignment ,the extracted strokes are then processed to form tight rectangular bounding boxes around the corresponding text strings.

To detect text over a wide range of font sizes, the above steps are first apllied to pyramid of images generated from the input image, and then the boxes formed at each resolution of the pyramid are fused at the original resolution. Third, an algorithm which cleans up the background and binarizes the detected text is applied t extract the text from the regions enclosed by the bounding boxes in the iput image. Finally ,text bounding boxes are refined(re-generated)by using the extracted items as strokes. these new boxes usually bound text strings better. The clean-up and binarisation process is then carried out on the regions in the input image bounded by the boxes to extract cleaner text.

The extracted text can then be passed through a commercial OCR engine for recognition if the text is of an OCR-recognizable font. Experimental results show that the algorithms work well on images from a wide variety of sources ,including newspapers ,magazines, printed advertisements ,photographs, digitized video frames and checks .the system is also stable and robust ,the system parameters work for all the experiments.

### 1.2Objective

To identify the text in a given image and extracting the text in a clear background.

**1.3Problem Statement**

OCR technology has been used to convert the text in scanned paper documents into ASCII symbols .However, current commercial OCR systems do not work well if text is printed against shaded or hatched backgrounds, often found in documents such as photographs, maps, monetary documents ,engineering drawings and commercial advertisements .Furthermore ,these documents are usually scanned in greyscale or color to preserve details of the graphics and pictures which often exist along with the text. For current OCR systems, these scanned images need to be binarized before actual character segmentation and recognition can be done. A typical OCR system does the Binarization to separate text from the backgrounds by global thresholding .Unfortunately, global thresholding is usually not possible for complicated images, as noted by many researches. Consequently, current OCR systems work poorly in these cases.
One solution to the global thresholding problem is to use different thresholds for different local regions(adaptive thresholding)

**1.4Chapter wise Summary**

- Texture segmentation method is used to identify the text
- The strokes are detected using stroke fitering ,stroke aggregation methods.
- After the strokes are removed ,text detected and background is cleaned up.
- Using OCR testing ,text is retrieved successfully, and displayed in a notepad.

**2.Existing System**

OCR technology has been used to convert the text in scanned paper documents into ASCII symbols. However, the disadvantage of OCR testings are.

- Current commercial OCR systems do not work well if text is printed against shaded or hatched backgrounds,
- Often found in documents such as photographs, maps, monetary documents, engineering drawings and commercial advertisements.

Furthermore, these documents are usually scanned in greyscale or color to preserve details of the graphics and pictures which often exist along with the text. For current OCR systems,
these scanned images need to be binarized before actual character segmentation and recognition can be done. A typical OCR system does the binarization to separate text from the background by global thresholding Unfortunately, global thresholding is usually not possible for complicated images, as noted by many researchers .Consequently, current OCR systems work poorly in these cases.

**2.1.Proposed System**

A new end to end system is proposed which automatically extracts and recognize text in images First ,a texture segmentation module which directs attention to where the text is likely to occur.
Second ,strokes are extracted from the segmented text regions .using reasonable heuristics on text strings such as height similarity, spacing and alignment .the extracted strokes are then processed to form rectangular bounding boxes around the corresponding hypothesized (detected) text strings. To detect text over a wide range of font sizes. The above steps are first applied to a pyramid of images generated from the input image, and then the boxes

formed at each resolution level of the pyramid are fused at the image in the original resolution level.

Third, an algorithm which cleans up the background and binarizes the detected text is applied to extract the text from the regions enclosed by the bounding boxses in the input image.

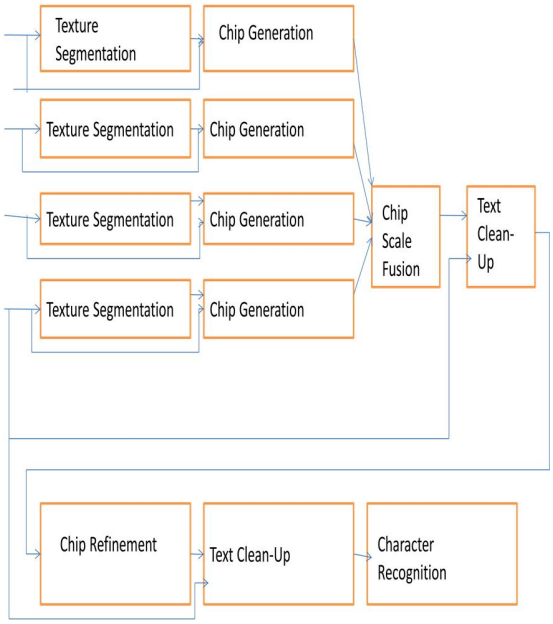
Finally ,the extracted items are treated as strokes and the more restrictive heuristics are used to generate better boxes for the text string while eliminating most of the false positive boxes. The clean-up process is then applies again to extract text which can then be processed by an OCR module for recognition.

3.**IMPLEMENTATION**

**4.IMPLEMENTATION PROCESS**

**4.1Texture Segmentation**

As stated text can be treated as a specific texture . Thus, one natural way to detect text is by using texture segmentation .A standard approach to texture segmen




tation is to first filter the image using a bank of linear filters, such as Gaussian derivatives(or) Global functions, followed by some non-linear transformation such as half-wave rectification, full-wave rectification, or a hyperbolic function).then features are computed to form the feature vector for each pixel from the filtered images. These feature vectors are then classified to segment the textures into different classes.

Actually 9filters are used to segment the texture. The filters are the 3second order derivatives of Gaussian at three different scales. Each filter output is passed through the non-linear function. A local energy estimate is computed using the outputs of the non-linearity. The results consist of 9images represents the local energy due to given filter .At each pixel ,a feature vector can be constructed consisting of the energy estimates from the 9images of the location. The “image” of feature vectors is clustered using a K means algorithm(with K=3).One of the clusters is labelled as text.

**4.2Stroke Filtering**

The purpose of stroke filtering is to eliminate the false positive strokes by using heuristics which take into account the fact that neighbouring characters in the same text string usually have similar heights and are horizontally aligned. it is reasonable to assume that the similarity of character heights causes the heights of the corresponding stokes to be similar. Furthermore, since the focus is on finding text string, a text stroke should have similar strokes nearby which belong to the same text string. These heuristics can be described using connect ability which is defined as:

A stroke is eliminated if one of the following conditions are true:

- It does not sufficiently overlap with the segmented text regions produced by the texture segmentation phase.
- It has no connectable stroke.

**Condition 1** says the strokes are expected to overlap the segmented regions. Since the text segmentation is often not perfect, one cannot expect total overlap. A minimum of 30% overlap rate worked well for all the test images.

**Condition 2** says that if there is no path that leads to some connectable stroke(s),it is probably an isolated stroke or line which does not belong to any text string.

the result of applying this procedure to the strokes in fig4.2.Notice that most of the text is still present while more of the background has been eliminated.

#### 4.2.1Stroke Aggregation

An important function for the Chip Generation phase is to generate chips that correspond to text strings. This is done by aggregation strokes belonging to the same text string.

Since characters which belongs to the same text string are expected to be of similar height and horizontally aligned, the concept of connect ability can be used to aggregate the strokes. In addition, it is clear that strokes corresponding to the same text string should be close to each other. The width of a character and the spacing between adjacent characters in a text string are related to the heights of the characters Thus, it is reasonable to measure the spacing between adjacent strokes as a function of the heights of the strokes. By empirical observation, the spacing between the characters and words of a text string is usually less than 3 times the height of the tallest character, and so is the width of a character in most fonts. Therefore, for all of the experiments, the following criterion is used to generate chips.

#### 4.2.2Chip Filtering

Some non-text strokes may also pass the strokes Filtering process. and therefore form false positive chips requiring further filtering. This might happens, for example, when there are periodically occurring lines or patterns in the image.

A chip is eliminated if the width of its box is less than $c\omega_r$;or the height of its box is less than $ch_r$ or the aspect ratio of its box is larger than $ratio_r$.

It is usually difficult even for a human to read the text when its height is less than 7 pixels, thus 7 has been used for $ch_r$ for the experiments. A horizontal text sting is usually longer horizontally, hence setting $c\omega_r$ to at least twice the minimum height seems reasonable.Thus,in all of our experiments, $c\omega_r$=15 and $ch_r$=7 were used.Normally,the width of a text string should be larger than its height. But in some fonts, the height of a character is larger than its width. Therefore, $ch_r$=1.1 is used here, attempting to cover that case to some extent.

#### 4.2.3Chip Extension

It is expected that some strokes only cover fragments of the corresponding characters. Therefore, these strokes might violent the constraints used for stroke filtering and hence be eliminated.Consequently,some of the chips generated so far may only cover part of the corresponding text strings.Fortunately,this fragmentation problem can usually be corrected. Notice that the chips corresponding to the same text stroke are still horizontally aligned and of similar height. Thus, by treating the chips as strokes, the stroke aggregation procedure can be applied again to

aggregate the chips into larger chips. This is exactly what the chip Extension step does.

**4.3Text Detection and Clean-up**

This experiment demonstrates the performance of the system up to the Text Filtering process. Characters and words (as perceived by one of the authors) were counted in each image | this is the ground truth. Then, detected characters and words which are completely enclosed by the text boxes produced after the Chip Scale Fusion step were counted for each image. The total numbers of detected characters and words summed up over the whole test set are show clearly readable by a person after the Chip Refinement and Text Clean-up steps are counted for each image. Note that only the text which is horizontally aligned is counted (skew angle of the text string is less than roughly 30 degrees).

**4.4OCR Testing:**

After the text detection and cleanup process the text will be printed in a clear background . thus using the OCR tool text retrieved successfully and are displayed in a notepad.

**5 .CONCLUSION AND FUTURE WORK**

Current OCR and other document segmentation and recognition technologies do not work well for documents with text printed against shaded or textured backgrounds or those with non-structured layouts. In contrast, we have proposed a text extraction system which works well for normal documents as well as documents described in the above situations.

The system proposed is composed of the following steps.

First, a texture segmentation module which directs attention to where the text is likely to occur.

Second, strokes are extracted from the segmented text regions. using reasonable heuristics on text strings such as height similarity, spacing and alignment. the extracted strokes are then processed to form rectangular bounding boxes around the corresponding hypothesized (detected) text strings. To detect text over a wide range of font sizes. The above steps are first applied to a pyramid of images generated from the input image, and then the boxes formed at each resolution level of the pyramid are fused at the image in the original resolution level.

Third, an algorithm which cleans up the background and binaries the detected text is applied to extract the text from the regions enclosed by the bounding boxes in the input image.

Finally, the extracted items are treated as strokes and the more restrictive heuristics are used to generate better boxes for the text string while eliminating most of the false positive boxes. The clean-up process is then applies again to extract text which can then be processed by an OCR module for recognition.

- There are 21820 characters and 4406 words in the test images(perceivable to one of the authors).95% of the characters and 93% of the words have been successfully extracted by the system. out of the same 14703 characters and 2981 words of extracted text which are of OCR-readable fonts,84%of the character and 77% of the words are successfully recognized by a commercial OCR system.

The system is stable and robust all the system parameters remain the same throughout all the experiments.

**5.REFERENCE**

1] H. S. Baird and K. Thompson. Reading Chess. IEEE Trans. Pattern Anal. Mach. Intell., 12(6):552{559, 1990.
2] Mindy Bokser. Omni document Technologies. Proceedings of The IEEE, 80(7):1066{1078, July 1992.
3] Lloyd Alan Fletcher and Rangachar Kasturi. A Robust Algorithm for Text String Separation from Mixed
Text/Graphics Images. IEEE Transactions on Pattern Analysis And Machine Intelligence, 10(6):910{918, Nov.
1988.
4] C. A. Glaser. An Analysis of Histogram-Based Thresholding Algorithms. CVGIP: Graphical Models and Image
Processing, 55(6):532{537, Nov. 1993.
5] Anil K. Jain and Sushil Bhattacharjee. Text Segmentation Using Gabor Filters for Automatic Document Process-

.