# Self-directing Superimposed Text Detection and Removal from images using Inpainting for Region Filling with Smoothing

Priyanka Deelip Wagh
Department of Computer Engineering
SES's R. C. Patel Institute of Technology
Shirpur (MH), India.
*sonar.priyanka9@gmail.com*

D. R. Patil
Department of Computer Engineering
SES's R. C. Patel Institute of Technology
Shirpur(MH), India.
*dharmaraj.rcpit@gmail.com*

***Abstract***—In this paper we have implemented an improved superimposed artificial text detection and inpainting method. Two stage frame works to remove unwanted text from images: Detection of text from image and removal of text using method of inpainting. Smoothing is used to bring more efficiency and reduction of noise and unwanted connected components at the first. With the use of feature extraction, stroke filtering and centroid processing text detection is performed. Holes generated in text detection stage are filled using exemplar based inpainting for regions with appropriate information present in same image. We tested the implementation using different images, and compared the results of smoothing with results of implementation without smoothing. Experimental results show, improved text detection rate due to use of smoothing before detection & improved Peak Signal to Noise Ratio and reduced Mean Squared Error for exemplar inpainting over nearest best matching in painting.

***Keywords***—*Text Detection, Text Localization, Inpainting, smoothing.*

*****

## I. INTRODUCTION

Images and video holds various kinds of data. Text present in images and videos gives brief knowledge of many things regarding images or videos containing that. The various forms of texts existence in images or videos are: subtitle, caption, logos etc. But sometimes, to get better visibility of images or videos, or to make reuse of those, or for excluding some text information for security purpose, it is essential to remove texts from images or videos.

The most sophisticated tool used for such a changes, Photoshop is one of the best tool. But it needs a special training about how to use it in perfect way, which is quite difficult of normal nontechnical computer users. For this purpose, it is crucial to have a tool which can process images in such a cases without having user interference and can produce effective results for non-technical users.

To bring automaticity in the process of superimposed text detection and removal from images basic three stages need to be accomplished [1]:

1. Text Localization: To detect the best possible locations of text over image.
2. Text Extraction: To detect proper boundary and regions of text in localized text regions.
3. Text Removal: To remove the texts using inpainting.

The outline of article contains introduction regarding text detection and removal from image in section I, brief information about previous work in the area of text detection and removal in section II, Detail about methodology of used for practical text detection and removal in section III, experimental results and observations under section IV, while section V gives conclusions for this concept.

## II. RELATED WORK

Many researchers have worked in the field of text detection and recognition till a day. Basic parameters to consider while detecting text are: color, features, intensity, gradient, strokes, style, orientation, alignment, inter-character distance, edges etc. Basically text detections are based on color or intensity based approaches, gradient based approaches, and text classifier based approaches; while inpainting methods are based on image interpolation, texture synthesis or combination of these both approaches.

Julinda Gllavata et. al. has proposed a method for automatic text detection, localization and extraction of horizontally aligned text in images using color reduction technique along with projection profile analysis and using geometrical properties. This method generates text boxes with simplified background to be fed into an OCR engine for text recognition. The performance of this approach is tested and demonstrated using different frames from different video sequences [2].

Ohya et. al has proposed a method of adaptive thresholding and relaxation operations using the text features like Color, scene text style for text localization and recognition. Character recognition takes place using high similarity between character pattern candidates and the standard patterns from dictionary. They carried experiments on 100 images involving characters of different size and formats under uncontrolled light to get highly promising results [3].

Li et al. has utilized wavelet based Feature Extraction approach along with neural network for texture analysis in order to detect scene text, localization, enhancement and tracking along the video. Their system implements two modules: first is a sum of squared difference (SSD) to find initial position and second is contour to refine the position. The experimental results on video sources show that this is a robus method for detection and tracking of text [4][5].

Lim et al. has worked on text detection and localization using 2 levels DCT coefficient and macroblock type information for caption text existing in MPEG compressed video for content based indexing. This method works in three stages: text frame detection, text region extraction, character extraction. The benefit of this method is to avoid the stage of decompressing video into individual frames [6].

W. kim et al. has invented a novel frame for overlay text detection and extraction from video using transition map and projection of pixels. The key point about this notion is, to use transition of colors between superimposed text and it's background, with use of transition map. Next stage has a task of candidate region extraction and overlay text region determination in each candidate region. Using projection of overlay pixels in transition map and the text extraction text regions are localized accurately. Basic features of this method are it's robustness to work for different size, position, contrast and color and language independency [7].

Y. L. Chen has proposed a method of decomposing document images into distinct object planes such as textual regions, non-text regions, and background texture. Using knowledge based text extraction and identification accurate detection and extraction takes place from different object planes. An adaptive inpainting neighborhood adjustment scheme is applied for text removal. The experimental results proves that this method gives accurate text extraction in images with diverse illumination level, sizes, fonts in complex document images [8].

B. Epshtein et al. has Proposed a new method called as Stroke Width Transform for text detection in natural scenes. This method provides an operator to give stroke width of each pixel of image. Basic advantage of this method is that there is no need to use multi-scale computation or scanning windows and yet provides various font and language detection [9].

A. Criminisi et al. has proposed an inpainting algorithm, based on exemplar based textures synthesis which combines the approach of texture synthesis as well as inpainting. The replication of texture and structure is essential process here. But the order of filling matters a lot. The color values are computed using exemplar based synthesis which is combination of inpainting and texture synthesis. Here Computational efficiency is achieved by a block based sampling process. Best part of this method is this method gives visually plausible and efficient results for scratches as well as big occluding large objects [10].

J. Malobabic et al. has proposed a method for text detection and localization for superimposed video text using horizontal difference magnitude measure and morphological processing. Smoothing and multiple binarisation is used for enhancing result of modified version of wolf-jolion algorithm in form of character segmentation. They have given the experimental results for text detection, localization and recognition on 20min long MPEG-1 encoded television program [11].

## III. METHODOLOGY

Text in images owns different features such as colors, fonts, gradient, directions etc. Even various languages has different style to write which brings variety in case of text detection algorithms. At the same time, in case of inpainting, to bring the sight of artist for computerized tool various types of textures and their patterns plays a vital role about algorithms and logics. But the flow of processes for self-directing text detection and removal remains constant. The basic three Stages to accomplish the objective of Text detection and removal are as shows in fig. 1:

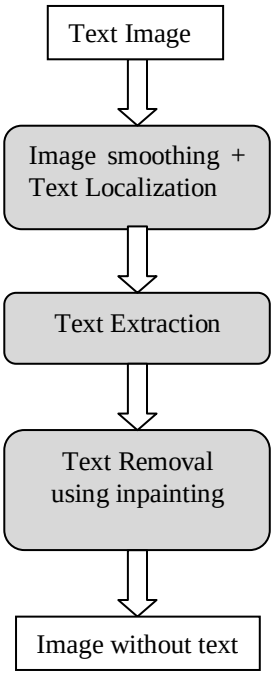


Fig. 1: Basic stages for self-directing Text Detection and Removal process.

### *A. Text Localization*

Text Localization is the first step in the direction to bring automaticity. The image contains so many features. For example objects, texts, colors, edges, gradient etc. These features play a vital role in classifying those as per need to determine text regions from other regions. We have implemented text localization based on the stroke filter [1] But sometimes, images consists of such a texture which makes the localization of correct text region difficult. Due that many times, false positive rate increases. To erode the effect of this problem, it is essential to reduce diversity of texture. This can be achieved using process of smoothing. Smoothing is nothing but low pass filtering, which is works effectively to remove high spatial frequency noise from digital images. Even due to this the unwanted connected components gets reduced which helps in getting better results for text detection. It improves text detection rate while reduces false positivity.

Here we have used image smoothing to remove small magnitude gradient and to suppress low amplitude details. At the same time, it is essential to retain and sharpen edges present in image. For that reason, image smoothing is carried out using L0 Gradient Minimization [12]. Here the amplitude changes are discretely counted. Basic feature of this method is, it is not depend on local features, yet locates important edges. This is powerful smoothing method based on discretely counting spatial changes which works efficiently on narrow object boundaries.

Highest contrast edges are enhanced by confining the discrete number of intensity changes among neighborhood pixels. Here by thinning salient edges, detection is made easier while visibility turns more distinct. Even blurriness is gets removed using this approach.

In this way after smoothing the original image, later the stroke filter application takes place as subtraction of minimum intensity neighborhood pixel from maximum intensity neighborhood pixel while separating color channels for image. Threshold stroke filter is used to highlight edge area of image, which generates double edges in image. A density filter is used to fill the holes between text edges followed by thresholding density image [1].

### *B. Text Extraction*

The regions localized in step A contains many such a blocks which are not text regions but many times important parts of images. So it is essential to filter out exact text regions and boundaries from the located regions. Text extraction is used to perform this task. The main purpose of this step of text extraction is to detect exact text region and boundaries, so that exact text removal takes place rather than losing the important information of remaining image. In text extraction stage, histogram of detected text boundaries is generated using major intensity change, and then gets smoothen. Smoothen histogram gets compared with background histogram. The similar peak values get removed from text boundaries histogram. The maximum peak from each text blocks are then used as text color after applying threshold. Half of the peak value is used as experimental threshold [1][2].

Finally after this step, all the three color channel text extraction outputs are combined using OR operation. The output of text extraction is text detection result which works as a stencil for next section of text removal to guide inpainting method about exact text locations. Fig. 2(c) shows sample result of text extraction by implemented method without smoothing and Fig. 2(d) shows result of text extraction by implemented method with smoothing.

### *C. Text removal using Inpainting*

After text localization and extraction, Inpainting helps to fill region using existing information from same image to remove text regions. There are many approaches works as per texture present in images. So limitation of this approach is that as images consists of various types of textures, it goes tough to fill regions perfectly by using one image feature as a base for all sort of images. Inpainting methods give results as per type of texture. We have used Exemplar based region filling method which is one of the effective inpainting method for wide range of random texture patterns [10].

The main notion behind this method is to combine the advantages of two methods of filling regions in images: first is texture synthesis which is preferred to fill large holes basically while second is inpainting which is used to fill small holes, scratches and gaps in images.

Texture synthesis copies pieces of texture from other parts of image and fix those by synthesizing them at holes or target regions. Inpainting fills the gaps and small holes using structural components of image. The fusion of texture synthesis and inpainting generates remarkable results for text detection and removal. The process of filling whole starts from border of holes. Using convolution, the intersecting edges of unknown area are searched. The method of thresholding is used to filter unnecessary edges followed by thinning of edges. The angel between incoming edge and border unknown region is calculated.

Using product of confidence and structural factor for each patch in consideration priorities of patches are generated and highest priority patch is used to fill particular region. The patches from region except target region work as sources. Best matching patch is calculated and placed at hole. Confidence values are updated for next step. The patches with more overlapping confident areas are best suitable. Iteratively this works till having positive confidence values for all pixels[10].

In implementation it has been observed that Exemplar based text filling works to produce better visually plausible results over nearest best matching algorithm in many cases having natural textures.

## IV. EXPERIMENTAL RESULTS

The implementation has been tested on MATLAB-R2013a, with system having Intel Core i7 2630QM Processor 2GHz, 8GB DDR3 RAM, HD Graphics 3000, Windows 8.1.


Fig. 2. Text Detection Results: (a) Actual Text existing in image, (b) Text Detected using Stroke Width Transformation, (c) Text Detection using Implemented method without Smoothing, (d) Text Detection using Implemented method with smoothing.

To measure the performance of text detection and region filling, we added some text elements to various images. Then after applying text localization and extraction to find accurate measurements of results we worked on pixel based detection rather than text blocks based and compared the results of implemented text detection algorithm with stroke width transformation method [9] and text detection without smoothing. By calculating the difference between binary images of difference between texts inserted images and original images, actual true regions of text are determined as shown in fig.2(a).

Table I, shows the results of text detection using smoothing compared with text detection without smoothing and stroke width transformation based text detection. The results are basically carried out on the English text images generated by superimposing the text on images. It shows the parameters as ATP, TP, FP, FN, DR, FAR, FRR means Actual Text Percentage, True Positive, False Positive, False Negative, Detection Rate, False Acceptance Rate and False Rejection Rate respectively using pixel counting in percentages [1].

DR=TP/(TP+FN)

FAR=FP/(TP+FN)

FRR=FN/(TP+FN)

Table I: Text Detection Output Performance (all are in %)

| Language | ATP | TP | FP | FN | DR | FAR | FRR |
|---|---|---|---|---|---|---|---|
| **Stroke Width Transformation** | 5.46 | 2.72 | 1.15 | 2.74 | 0.49 | 0.21 | 0.50 |
| **Implemented Text Detection without smoothing** | 5.46 | 2.14 | 1.23 | 3.32 | 0.39 | 0.22 | 0.61 |
| **Implemented Text Detection with smoothing** | 5.46 | 3.60 | 1.72 | 1.85 | 0.66 | 0.32 | 0.34 |

From table I, it is observed that, percentage of True Positive and Detection Rate improves in implemented text detection using smoothing as compare to percentage of True Positive and Detection Rate of stroke width transformation based text detection and Implemented text detection without smoothing.


Fig. 3: Text Removal Outputs: (a) Original Image with Texts, (b) Result of Nearest Best matching Inpainting without smoothing (c) Result of Nearest Best Matching Inpainting with smoothing (d) Result of Exemplar Based Inpainting with smoothing.

Table II, shows the comparative performance of Text Inpainting using nearest best matching(NBM) inpainting with smoothing and without smoothing and Exemplar based inpainting with smoothing using Peak Signal to Noise Ratio(PSNR) and MSE(Mean Squared Error) as a reference to original image without text.

Table II: Text Inpainting output comparison for with and without smoothing operation for Nearest Best Matching (NBM) inpainting and Exemplar based inpainting on smoothing based text detection result

| | **NBM Inpainting Without Smoothing** | **NBM inpainting with smoothing** | **Exemplar based inpainting with smoothing** |
|---|---|---|---|
| **MSE** | 0.0054 | 2.7077e-04 | 1.9235e-04 |
| **PSNR** | 70.84 | 83.80 | 85.2899 |

From table II, it is observed that, the Mean Squared Error reduced using smoothing in nearest best matching inpaiting as well as using Exemplar based inpainting while increasing PSNR values with better visually pleasant inpaiting results.

## V. CONCLUSIONS

In this paper, the implementation details for self-directing text detection and removal from images using smoothing along with exemplar based inpainting has been described. Here it has been experimented that, the text detection and text inpainting results can be improved by using smoothing of image and inpainted regions in nearest best matching inpainting as well as exemplar based inpainting. The PSNR value of images without smoothing is less than PSNR values for images with smoothing. Even in case of Mean Squared error (MSE) decreases with use of smoothing. So, this whole implementation and paper gives a conclusion that use of smoothing method in text detection and inpainting gives visually more plausible output as compared to text detection and inpainting without smoothing even for exemplar based inpainting.

## REFERENCES

[1] M. Khodadadi, A. Behrad, "Text Localization, Extraction and Inpainting in color Images", Iranian Conference on Electrical Engineering (ICEE2012),May 2012, pp. 1035-1040.

[2] Julinda Gllavata, Ralph Ewerth, Bernd Freisleben, "A Robust Algorithm for Text Detection in Images",Image and Signal Processing and Analysis(ISPA- 2003), Vol.2, pp. 611-616, 2003

[3] J.Ohya, A. Shio, and S. Akamatsu, "Recognizing Characters in Scene Images", IEEE Transactions on Pattern Analysis and Machine Intelligence, 16(2)(1994)214-224.

[4] H. Li, D. Derman, and O. Kia, "Automatic Text Detection and Tracking in Digital Video", IEEE Transactions on Image Processing, 9(1) January(2000)147-156

[5] H. Li and D. Dermann, "A Video Text Detection System based on Automated Training", Proc. Of IEEE Internationl Conference on Pattern Recognition, 2000, pp.223-226.

[6] Y.K. Lim, S. H. Choi and S. W. Lee, "Text Extraction in MPEG Compressed Video for Content-bsed Indexing", Proc. Of International Conference on Pattern Recognition, 2000, pp, 409-412.

[7] W. Kim and C. Kim, "A new approach for overlay text dtection and extraction from complex video scene", IEEE Trans. Image Process., vol.18, no.2, pp.401-411, feb. 2009.

[8] Yen-Lin Chen, "Automatic Text Extraction, Removal and Inpainting of Complex Document Images", in International Journal of Innovative Computing, Information and Control, Vol. 8, No. 1(A), pp. 303-327, January 2012.

[9] B. Epshtein, E. Ofek, and Y. Wexler, "Detecting text in natural scene with stroke width transform", IEEE conference on Computer Vision and Pattern Recognition(CVPR),2010, pp.2963-2970, June 2010.

[10] A. Criminisi, P. Perez, and K. Toyama, "Region Filling and Object Removal by Exemplar-Based Image Inpainting", IEEE Trans. Image Processing, Vol. 12, no. 8, Aug 2003.

[11] J. Malobabic, N. O'Connor, N. Murphy, and S. Marlow, "Automatic Detection and Extraction of artificial text in video", in WIAMIS 2004-5th International Workshop on Image Analysis for Multimedia Interactive Services, April 2004.

[12] Li Xu, Cewu Lu, Yi Xu and Jiaya Jia, "Image Smoothing via L0 Gradiet Minimization",ACM Transactions on Graphics (SIGGRAPH Asia), 2011, vol. 30, no. 6, pp. 174:1-174:12.