Local Features based Text Detection Techniques in Document Images

M Sharmila Kumari
Department of Computer Science and Engineering
P A College of Engineering
Nadupadavu, Mangalore

Akshatha
Department of Computer Science and Engineering
P A College of Engineering
Nadupadavu, Mangalore

ABSTRACT
Video text information plays an important role in semantic-based video analysis, indexing and retrieval. It is observed that the detection of texts in video remains as a challenging task due to its complex varying conditions. In this paper, we present a study on local features based text detection in document images and more focus is provided for text detection based on Laplacian method. The document image is convolved with Laplacian operator to filter the document image. Then the maximum gradient difference value is computed for each pixel to generate threshold. Based on the computed threshold, a binarized frame is obtained which highlights the text block. The candidate text block regions are further verified and refined that is, the corresponding region in the Sobel edge map of the input image undergoes projection profile analysis to determine the boundary of the text blocks. Finally, empirical rules are employed to eliminate false positives based on geometrical properties. In addition, a comparative study of the Laplacian method with a novel text detection and localization method based on Corner response and Multi scale edge based method for video text detection is made. The techniques are evaluated on documents taken from ICDAR 2003 robust reading and text locating database. Experimental results show that the Laplacian method is able to detect texts of different fonts, contrast and backgrounds. To give an objective comparison of the Laplacian approach, we have used detection rate and false positive rate as decision parameters and metrics.

Index Terms—Laplacian operator, Corner response, Multi-scale edge. Text detection, Text localization,

I. INTRODUCTION
Texts in images usually carry important messages about the content [3].Images and videos on webs and in databases are increasing [12]. Video sequences are usually integrated with audio, image, graph, text and so on. So, the text in images and video frames carries important information for visual content understanding and retrieval. Images from a mobile camera, indoors or outdoors pose considerable challenges to understanding, such as blurred or out of focus frames, uneven lighting, complex backgrounds, and lens distortion [14].
In general, there are two types of texts in video sequences, namely, the scene texts and the artificial texts. The scene texts came from cameras and are naturally embedded in scenes, such as the text in trademarks, signpost and so on. Artificial texts are purposely added to video frames during video editing. Thus, they are closely related to the content of the video. Detection of both scene text and graphic text in video images is gaining popularity in the area of information retrieval for efficient indexing and understanding the video [12]. Compared with the other image features, text is embedded into images or scenes by human, which can directly reveal the image content in a certain point of view without requiring complex computation. Therefore, it has inspired a lot of research on text detection and recognition in images and video.
The role of text detection is to find image regions containing only text that can be directly highlighted for certain applications. Text, which carries high-level semantic information, is a kind of important object that is useful for many tasks. For example,

- Text in web images can reflect the content of the web pages.
- Text on book and journal covers can be helpful to retrieve the digital resources.
- Caption text in news videos usually annotates information on where, when and who of the happening events.
- Sub-title in sport videos often annotates information of score, athlete and highlight. Scene text often suggests the presence of a fact such as advertisement board, traffic warning, etc [13].

Texts are important objects embedded in natural scenes.They often carry useful information such as traffic signals advertisement billboards, dangerous warnings, etc. Automatic text recognition induces a lot of potential applications such as:

- Helping a foreigner to understand the contents on an information board (by translating the recognized text into his native language);
- Helping blind people to walk freely in a street; and
- Drawing attention of a driver to traffic signs [10].

Unfortunately, text characters contained in images and videos have many undesirable properties of such as low-resolution, low contrast, unknown text color, size, position, orientation, color bleeding noise, image quality and text embedded in complex backgrounds [4]. Although, many methods have been proposed over the last decade, great number of works deals with detection of text from natural images and video frames. The accuracy of text detection in document images or video greatly influences on the performance of information retrieval and subsequently recognition task[4]. Therefore, text should be detected for semantic understanding and image indexation.
Various methods were proposed for the isolation of text data from the documented image. The previous work of text detection usually classified into: 1) Connected component-based, 2) Edge-based, 3) Texture-based, 4) Machine learning based method [2]. The first approach does not work well for all video images because it assumes that text pixels in the same region have similar colors or grayscale intensities. The second approach requires text to have a reasonably high contrast to the background in order to detect the edges. So these methods often encounter problems with complex backgrounds and produce many false positives. Finally, the

third approach considers text as a special texture and thus, uses fast Fourier transform, discrete cosine transform, wavelet decomposition and Gabor filters for feature extraction. However, these methods require extensive training and are computationally expensive for large databases. Machine learning based method use features extracted from text region and non-text region to train support vector machine or neutral network and then text detection be-comes a supervised classification problem. In [5], a SVM based algorithm using response from stroke filter is proposed. Huet al. propose an adaptive SVM based paradigm, based on maximum gradient differences and other connected component features, which can obtain relatively low false rate [6]. The shortcoming of machine learning based method is that it needs a large number of training samples of different kind.

In this paper, we consider existing methods [7, 8, 15] for comparative study. Liu et al. [7] extract edge features by using the Sobel operator. This method is able to determine the accurate boundary of each text block. However, it is sensitive to the threshold values for edge detection. Wong et al. [8] compute the maximum gradient difference values to identify candidate text regions. This method has a low false positive rate but uses many threshold values and heuristic rules. Therefore, it may only work well for specific datasets. Finally, Mariano et al. [15] perform clustering in the L*a*b* color space to locate uniform-colored text.

In the following sections, we present the descriptions of the methods and present the experimental results with a conclusion.

2. CORNER RESPONSE BASED METHOD

A novel text detection and localisation method based on corner response consist of 3 stages: (1)computing corner response in multi-scale space and thresholding it to get the candidate region of text; (2) Verifying the candidate region by combining color and size range features;(3) Locating the text line using bounding box. Corner is a special two-dimensional feature point which has high curvature in the region boundary. It can be located by finding the local maximum in corner response (CR). In [7], corner points in video frame are used to generate connected component. But they use just the number of corner points, not CR, to classify text and non-text region. Given an image I (x, y), the basic form of CR is

$CR(x,y)=\sum_{u,v} W(u,v)\,[I(x+u, y+v) - I(x,y)]^2$ (1)

Here W (u, v) is window function. It can be proved that CR can be approximately computed using the formula below.

$CR(x,y) = A(x,y)\,B(x,y) - (C(x,y))^2 - \text{weight} * (A(x,y) + B(x,y))^2$ (2)

Here A(x, y), B (x, y) and C (x, y) are computed as follow:

$$A(x, y) = W(u, v) * (\nabla x\, I(x, y))^2 \quad (3)$$

$$B(x, y) = W(u, v) * (\nabla y\, I(x, y))^2 \quad (4)$$

$$C(x, y)=W(u,v) * \nabla x\, I(x, y) * \nabla y\, I(x, y) \quad (5)$$

In the formula above, ∇x I (x, y) and ∇y I (x, y) are edge amplitudes along x direction and y direction which we can get by Sobel operator. W (u, v) is a Gaussian template for smoothing.

$$W(u, v) = \exp -(u^2 + v^2)/2\sigma \quad (6)$$

We can choose σ value and size of the template. Then the mean intensity value of each block in CR M_{blk} is calculated. A threshold T_{blk} is set for M_{blk}. If the following condition satisfied,

$$M_{blk} > T_{blk} \quad (7)$$

$$T_{blk} = \frac{1}{H \times W} \sum_{x=0,y=0}^{H,W} CR(x, y) \quad (8)$$

The current block is considered as one of the block in text candidate region. In each candidate block, we set a threshold T_{CR} for every pixel in CR and get a collection of points R_t and R_b in each block.

$$CR(x, y) \geq T_{CR}, (x, y) \in R_t$$
$$CR(x, y) < T_{CR}, (x, y) \in R_b$$

Then we calculate Dev and Dis as follows. Here g (x, y) is the gray value of a pixel.

$$DEV = \sqrt{\frac{1}{NT} \sum_{(x,y) in Rt} (g(x,y) - Mt)^2} \quad (9)$$

$$Dis = |Mt - Mb| \quad (10)$$

Mt and Mb are mean gray value in Rt and Rb respectively. Finally, we check if the following condition is satisfied.

Dis >TdisDev<Tdev (11)

If it is satisfied, we consider the current block as text block.

3. MULTISCALE EDGE BASED METHOD

Multi-scale edge based method or video text detection consist of 3 stages: (1) Candidate text region detection, (2) Text region localization, (3) Character extraction. Feature map is generated using three important properties of edges: edge strength, density and variance of orientations. The feature map is a gray- scale image with the same size of the input image, where the pixel intensity represents the possibility of text. Here we use magnitude of the second derivative of intensity as a measurement of edge strength. The edge density is calculated based on the average edge strength within a window. Considering effectiveness and efficiency, four orientations (0°,45°,90°,135°) are used to evaluate the variance of orientations, where 0° denotes horizontal direction, 90° denote vertical direction, and 45° and 135° are the two diagonal directions, respectively. A convolution operation with a compass operator results in four oriented edge intensity images which contain all the properties of edges. Regions with text in them will have significantly higher values of average edge density, strength and variance of orientations than those of non-text regions. We exploit these three characteristics to generate a feature map which suppresses the false regions and enhances true candidate text regions.

$$fmap(i,j)=\bigoplus_{s=0}^{n} \sum_{\theta} N\{\sum_{x=-c}^{c} \sum_{y=-c}^{c} E(s,\theta,i+x,j+y) \times W(i,j)\} \quad (1)$$

fmap is the output feature map, is $\bigoplus$ across-scale addition operation, which employs the scale fusion. N is the highest level of scale, which is determined by the resolution (size) of the input image, N is the normalization operator. W(i ,j) is the weight for pixel(i, j), whose value is determined by the number of edge orientations within a window. The window size is determined by a constant c. The intensity of the feature map represents the possibility of text; a simple global thresholding can be employed to highlight those with high text possibility regions resulting in binary image. A morphological dilation operator can easily connect the very close regions together while leaving those whose position is far away to each other isolated. A morphological dilation operator with a 7x7 square structuring element to the previous obtained binary image to get joint areas referred to as text blobs. Four pairs of coordinates of the boundary boxes are determined by the maximum and minimum coordinates of the

top, bottom, left and right points of the corresponding blobs. In order to avoid missing those character pixels which lie near or outside of the initial boundary, width and height of the boundary box are padded by small amounts.

4. LAPLACIAN OPERATOR BASED METHOD

This section explains the method for finding text regions in images based on Laplacian method. It consists of 5 stages: (1) Text detection; (2) Maximum Gradient Difference; (3) Compute Threshold; (4) Boundary refinement; (5) False positive elimination.

In this approach, firstly we identify candidate text regions by using the Laplacian operator. In order to capture the relationship between positive and negative values, we use the maximum gradient difference. Threshold is computed to classify all the pixels into text and non-text, and then projection profile analysis is done to determine the accurate boundary of each text block. Finally, false positives are removed based on geometrical properties. Experimental results show that the Laplacian method outperforms the above methods in terms of detection and false positive rates [1].

A. Filtering the input image

The input image has to be segmented to obtain a set of regions representing individual characters or small groups of them. For this purpose, a text detector is employed. In the first step, we identify candidate text regions by using the Laplacian operator ie the transitions between text and background. Therefore, the input image is converted to grayscale and filtered by a 3 × 3 Laplacian mask to detect the discontinuities in four directions: horizontal, vertical, up-left and up-right(fig. 1).

1	1	1
1	-8	1
1	1	1

Fig1: The 3 × 3 Laplacian mask.

It is observed that the Laplacian-filtered image restores the overall gray level variation in the image by increasing the contrast at the location of gray-level discontinuities. The net result is an image in which small details were enhanced and the background tonality was perfectly preserved. Such results have made Laplacian based enhancement a fundamental tool used frequently for sharpening digital images(Fig 2).

Fig 2: Laplacian filtered image

B. Computation of Maximum Gradient Difference

Since these filtered image produces two values it corresponds to the transitions between text and background. Text regions typically have a large number of discontinuities, e.g. transitions between text and background. In order to capture the relationship between positive and negative values, we use the maximum gradient difference (MGD), defined as the difference between the maximum and minimum values within a local 1 × N window (horizontally) N ×1 (vertically) . The value of N will have an effect on the net result. The MGD value at pixel (x, y) is computed from the Laplacian-filtered image f as follows.

MGD(x, y) = max (f(x, y-t)) – min (f(x, y-t)) (1)

$$t \in \left[-\frac{N-1}{2}, \frac{N-1}{2} \right]$$

A local 1 × N window slide over the image is used to obtain the MGD map. It is observed through extensive experimentation that the maximum gradient difference value of the pixel is more or less positive and high in the case of the text pixel and is relatively small in the case of a background pixel. Text regions have larger MGD values because of the peaks of large magnitudes.This can be seen in the figure 3 where we have given an image containing both text and non text data.

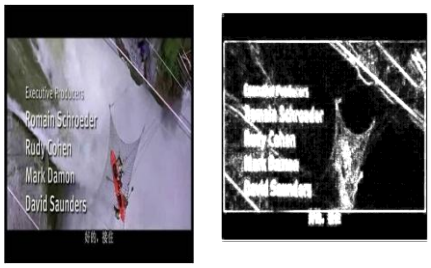


Fig.3. Maximum Gradient Difference

It shall be observed that the textual information is highlighted and non-textual information is more or less suppressed [4]. In MGD Map it can be visualized that text regions typically have larger MGD values than non-text regions because they have many positive and negative peaks. So these values are normalized to the range [0, 1] using Min –max normalization , Z-score normalization and normalization using Decimal Scaling. Suppose that min_a and max_a are minimum and maximum values of an matrix A, then min-max normalization maps a value V of A to V' in the range[new min, new max] by computing

$V'=V-min_A/max_A-min_A(newmax_A-newmin_A)+newmin_A$

It preserves the relationships among the original data values and reduces the complexity for computation.

C. Threshold Computation

Let G_H and G_V be the MGD images horizontally and vertically respectively. Then the pixel is classified as text pixel based on following rule.

$$T(x,y)=\begin{cases} text & \text{if } G_H(x,y)>\tau_H \text{ and } G_V(x,y)>\tau_V \\ nontext & otherwise \end{cases}$$

The threshold T_H is determined based on the average value of horizontal MGD values as:

$$H_{AVG}=\frac{1}{m\times n}\sum_{i=1}^{m}\sum_{j=1}^{n} G_H(x,y)$$

where m and n are the dimension o the horizontal image. Next we count the number of significantly higher horizontal values as:

$NH_{HOR}=count\ (G_H(x,y))=H_{AVG}$

The sum of horizon G_H is computed as

$$SH_{AVG}=\sum_{i=1}^{m}\sum_{j=1}^{n} G_H(x,y)$$

Finally the value of τ_H is computed as follows:

$$\tau_H=\frac{SH_{AVG}}{((m\times n)-NH_{HOR})}$$

Similarly the vertical threshold is computed. Both τ_H and τ_V are used to classify the text and non-text pixel (fig 4).

Fig 4. Text region highlighted

D. Boundary refinement

The Sobel operator performs a 2-D spatial gradient measurement on an image and so emphasizes regions of high spatial gradient that corresponds to edges. Typically it is used to find the approximate absolute gradient magnitude at each point in an image grayscale image. Since it is difficult to determine the boundary of each text block directly from the text cluster because of false positives and connected text lines. Therefore, we compute the binary Sobel edge map SM of the input image (only for text regions) (fig 5 & 6). The horizontal projection profile is defined as follows.

$$HP(i) = \sum SM(i,j) \quad (2)$$

Fig 5. Horizontal refinement

If HP(i) is greater than a certain threshold, row i is part of a text line; otherwise, it is part of the gap between different text lines. From this rule, we can determine the top row i1 and bottom row i2 of each text line. The vertical projection profile is then defined as follows.

$$VP(i) = \sum SM(i,j) \quad (3)$$

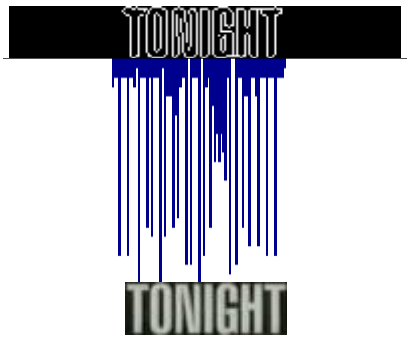


Fig 6. Vertical refinement

Similarly, if VP (j) is greater than a certain threshold, column j is part of a text line; otherwise, it is part of the gap between different words. Finally, different words on the same text line are merged if they are close to each other.

By applying this step recursively, we can determine the accurate boundary of each text block, even when the text blocks are not well-aligned or when one candidate text region contains multiple text lines. At the end of this step, each detected block is a candidate text block (Fig 6)

E. False Positive Elimination

Finally, false positives are removed based on geometrical properties. Let W, H, AR, A and EA be the width, height, aspect ratio, area and edge area of text block B.

$$AR = W/H \quad (4)$$

$$A = W \times H \quad (5)$$

$$EA = \sum SM(i,j) \quad (6)$$

If $AR < T_1$ or $EA / A < T_2$, the candidate text block is considered as a false positive; otherwise, it is accepted as a text block (Fig. 7). The first rule checks whether the aspect ratio is below a certain threshold. The second rule assumes that a text block has a high edge density due to the transitions between text and background.

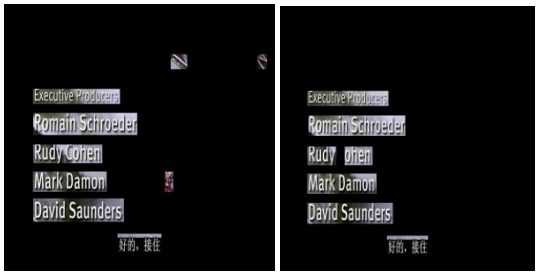


Fig 7. False detection elimination

It is optimal to use any edge detector to prune the results so that no non-textual information is on the output image. To summarize, the overall algorithm is as follows:

5. EXPERIMENTAL RESULTS

For experimental purpose, we have created our own dataset on our own video database of images such as movies, news clips, sports and music videos, and scene images, document images (ICDAR-2003) available over internet and this section presents the results of the experiments conducted to present the comparative results of three local features based approaches. The database contains images of varying size, different formats and also of different colors. All experiments are performed on a P-IV 2.99GHz windows machine with 504 MB of RAM.

Text detection results on some sample images due to the Laplacian model are given in fig 8 below:

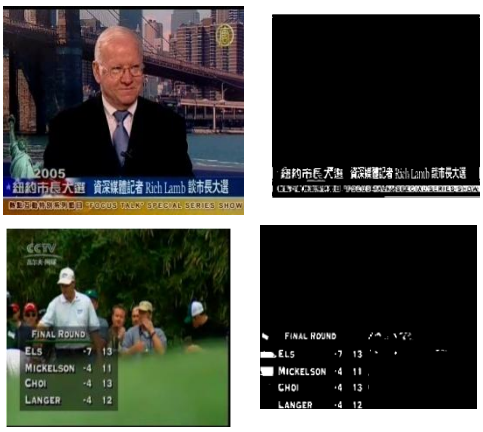


Fig 8. Sample result (a) different font size (b) different language (c) different contrast.

We have also provided an experimental comparison with corner response based method and multi scale edge based method (see Fig. 9). The results of the Laplacian method are able to locate text better when compared to other two methods. To give an objective comparison of the Laplacian method, we have made a comparison with corner response based method and multi scale edge based method. Similar to their evaluation criteria, we have used detection rate and false positive rate as decision parameters and metrics. To judge the correctness of the text blocks detected, we manually count the Actual Text Blocks (ATB) in the video frames. Also manually label each of the detected blocks as either truly detected text

blocks (TDB): a detected block that contains text or a falsely detected text block (FDB): a detected block that does not contain text.

Detection Rate (DR) =Number of TDB/Number of ATB.

False positive rate (FPR) =Number of FDB/Number of (TDB+FDB).

The performance of the Laplacian approach in comparison with the other existing approaches is summarized in table below.

Table -1. Comparative analysis of Text Localization

Method	DR	FPR
Corner response	79.90	17.90
Multi-scale edge	89.50	10.60
Laplacian method	92.24	18.33

Performance is estimated for each method separately as follows:

Recall= TDB/ATB

Precision= TDB/ (ATB+FDB).

In addition, we have also provided results of each method on a sample set of five images taken from ICADR dataset in the following tables.

Table 2. Result for Laplacain method

Images	ATB	TDB	FDB	Recall	Precision
1.jpg	6	6	0	1	1
2.jpg	16	15	1	0.938	0.882
3.png	17	14	8	0.824	0.56
4.jpg	11	11	5	1	0.687
5.jpg	9	7	5	0.777	0.5

Table 3. Result for corner response method

Images	ATB	TDB	FDB	Recall	Precision
1.jpg	6	6	4	1	0.6
2.jpg	16	13	6	0.812	0.590
3.png	17	16	8	0.941	0.64
4.jpg	11	11	9	1	0.55
5.jpg	9	9	3	1	0.75

Table 4. Result for multiscale edge method

Images	ATB	TDB	FDB	Recall	Precision
1.jpg	6	6	11	1	0.35
2.jpg	16	16	7	1	0.695
3.png	17	15	8	0.882	0.6
4.jpg	11	11	6	1	0.647
5.jpg	9	8	7	0.888	0.5

6. CONCLUSION

Text data present in images and video contain useful information for automatic annotation, indexing and structuring of images. Extraction of this information involves detection, localization, extraction, enhancement and recognition of the text from a given images. Experimental results show that:

1) Based on Laplacian method, the gradient information helps to identify the candidate text regions and the edge information serves to determine the accurate boundary of each text block.
2) Multi-scale edge based method distinguishes text regions from texture like regions, such as window frames, wall patterns, etc., by using the variance of edge orientations.
3) Corners give more clues for text detection and localization in image and it already reduces noise in the feature extraction stage.

However, variations of text due to differences in size, style, orientation and alignment as well as low image contrast and complex background make the problem of automatic text detection extremely challenging which we planning to address in our future works.

REFERENCES

[1] TrungQuyPhan, PalaiahnakoteShivakumara and Chew Lim Tan," A Laplacian Method for Video Text Detection", IEEE DOI vol.10, 2009, pp.153.

[2] Li Sun, Guizhong Liu, XuemingQian, DanpingGuo," A Novel Text Detection And Localization Method Based On Corner Response", Pattern Recognition, vol. 37, 2004, pp. 595–608.

[3] Ching-Tung " Embedded-Text Detection and Its Application to Anti-Spam Filtering", IEEE Trans. on Pattern Analysis and MachineIntelligence, vol. 10, 2008, pp. 910-918.

[4] XuemingQian, Guizhong Liu, Huan Wang, Rui Su," Text detection, localization, and tracking in compressed video", Signal Processing: Image Communication, vol. 22, 2007, pp.752 – 768.

[5] M. R. Lyu, J. Song and M. Cai, "A Comprehensive Method for Multilingual Video Text Detection, Localization, and Extraction", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 15, February 2005, pp. 243-255.

[6] C. Liu, C. Wang and R. Dai, "Text Detection in Images Based on Unsupervised Classification of Edge-based Features", ICDAR, 2005, pp. 610-614.

[7] E. K. Wong and M. Chen, "A new robust algorithm for video text extraction", Pattern Recognition , vol.36, 2003, pp. 1397-1406.

[8] P. Shivakumara, W. Huang and C. L. Tan, "An Efficient Edge based Technique for Text Detection in Video Frames", The Eighth IAPR Workshop on Document Analysis Systems (DAS2008), Nara, Japan, September 2008, pp 307-314.

[9] Qixiang Ye*, Jianbin Jiao, Jun Huang, Hua Yu," Text detection and restoration in natural scene images", J. Vis. Commun. Image R, vol.18, 2007, pp. 504–513.

[10] A.K. Jain and B. Yu, "Automatic Text Location in Images and Video Frames", Pattern Recognition, Vol. 31, 1998, pp. 2055-2076.

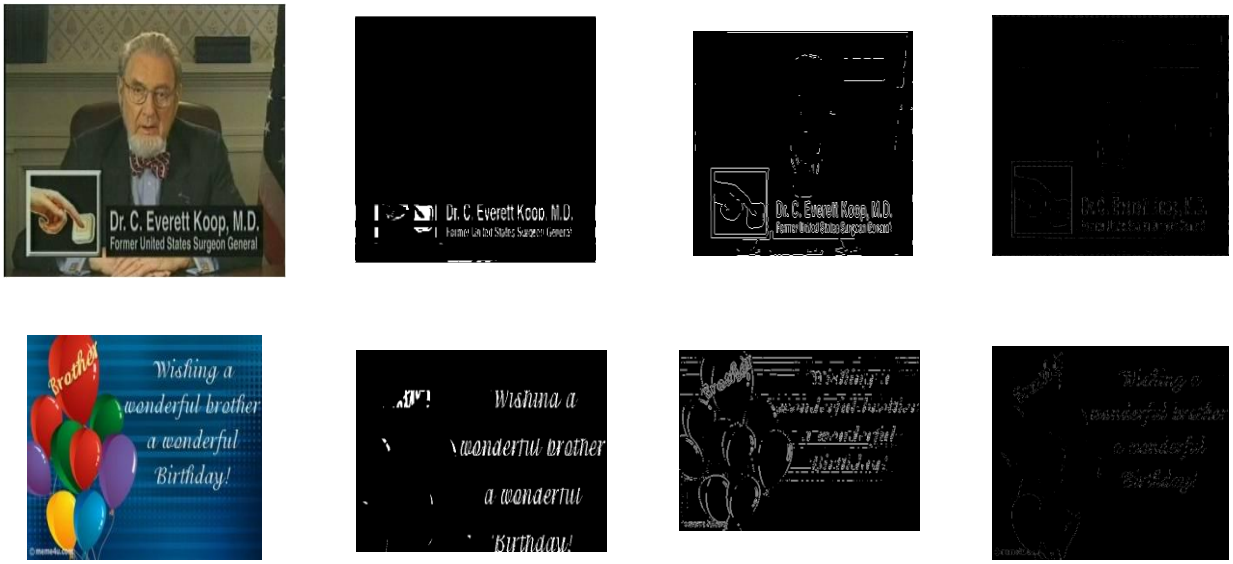
[11] P Shivakumara, Weihua Huang, TrungQuyPhan, Chew Lim Tan," Accurate video text detection through classification of low and high contrast images", Pattern Recognition, vol. 43, 2010, pp. 2165–2185.

[12] Qixiang Yea,*, QingmingHuangb, Wen Gao, Debin Zhao," Fast and robust text detection in images and video frames"', Image and Vision Computing , vol.23 , 2005, pp.565–576.

[13] B H Shekar and M SharmilaKumari"Text Detection in Video Frames: An integrated approach based on weber's local descriptor and differential excitation difference"

[14] V. Y. Mariano and R. Kasturi, "Locating Uniform-Colored Text in Video Frames", 15th ICPR , Vol. 4, 2000, pp 539-542.

[15] Datong Chen∗, Jean-Marc Odobez, Herv/ e Bourlard," Text detection and recognition in images and video frames", Pattern Recognition, vol. 37, 2004, pp. 595–608.



a)original document image **b)Result due to Laplacian Method** **c)Result due to Multiscale edge based method** **d)Result due to corner response method**

Fig. 9. Experimental comparison of Laplacian method, Corner response based method and Multiscale edge based method