Detection and localization of texts from natural scene images using scale space and morphological operations

CONFERENCE PAPER · JANUARY 2013

DOI: 10.1109/ICCPCT.2013.6528865

5 AUTHORS, INCLUDING:

Ajith V Pillai

3 PUBLICATIONS **0** CITATIONS

SEE PROFILE

Arun A. Balakrishnan

Rajagiri School of Engineering and Technology

13 PUBLICATIONS **0** CITATIONS

SEE PROFILE

Rina Simon

Rajagiri School of Engineering and Technology

1 PUBLICATION **0** CITATIONS

SEE PROFILE

Renoh Johnson Chalakkal

University of Auckland

6 PUBLICATIONS **0** CITATIONS

SEE PROFILE

Detection and Localization of Texts from Natural Scene Images using Scale Space and Morphological Operations

Ajith V.Pillai, Arun A. Balakrishnan†, Rina Anna Simon ‡, Renoh C Johnson §and Padmagireesan S ¶*
Dept. of Applied Electronics & Instrumentation, Rajagiri School of Engineering & Technology, Kochi. 682039
Email: ajithpillai87@yahoo.com*, arun31.01.1988@gmail.com†,
rinasimon@yahoo.com‡, renohcj@gmail.com§, pappan_gireesh@yahoo.co.in¶.

***Abstract*—Detection and localization of texts from natural scene images is important and can provide a much truer form of content-based image analysis if it can be extracted and harnessed efficiently. This problem becomes challenging because of complex background, variations of text font, size and line orientation, non-uniform illumination. A new unsupervised text detection algorithm is proposed in this paper. In this approach scale space and morphological operations for the edge detection are utilized. The non-text components are efficiently filter out by using scale decomposition and 2D Gaussian low pass filter. They are extracted based on observation that the edge of a character can be extracted from the complex scenes by taking into consideration the high similarities in length and aspect ratio. The proposed method yielded high precision when experiments where evaluated in the ICDAR 2003 dataset.**

***Keywords*—Scale space decomposition, image pyramid, morphological operations, Gaussian filter, thresholding.**

I. Introduction

The motivation for extracting text from image documents came up because, nowadays images are very popular in multimedia document, and are being produced on a daily basis by a wide variety of sources including the long distance educational programs, medical diagnostic systems, business and surveillance applications, broadcast and entertainment industries, etc. Recently, with the increasing availability of low cost portable cameras, the number of images being captured are growing at an explosive rate. According to statistics provided by Flickr [1], the quantity of photos uploaded to Flickr has been increasing 20% year-over-year over the last 5 years and reached 6 billion in August 2011.

Text information in image contents has inspired great interests, since it can be easily understood by both human and computer, finding wide applications. Applications include text appearing on vehicles, text on signs and billboards, and text on T-shirts and other clothing. Technically, the extraction of scene text is a tough task due to varying size, position, lighting, orientation, and deformation. Contrast to a large amount of text extraction techniques for caption, only limited work is found in the literature that focuses on robust scene text extraction from images.

An integrated image text information extraction system is shown in Figure 1 with four stages: text detection, text localization, text extraction and enhancement, and recognition. Bounded by dashed line is the text detection and localization, which is critical to the overall system performance.

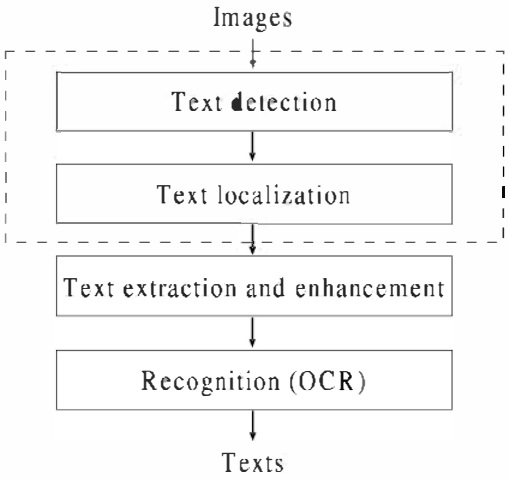

Fig. 1: Architecture of a TIE system

Existing methods of text detection and localization can be categorized into two groups: region-based and connected component (CC)-based. Region-based methods attempt to detect and localize text regions using texture analysis. Generally, a feature vector is extracted from each of the local region and is fed into a classifier for estimating the likelihood of text. Text blocks are then generated by merging neighboring text regions. The advantage of this method is that even if the images are noisy text regions can be detected since they have distinct textural properties from the non-text ones. While CC-based methods use color clustering or edge detection to segment candidate text components directly

Speed of region-based method is slow and the performance is sensitive to orientation of the text. While, CC-based methods need to have prior knowledge of scale and text position accurately. Designing a reliable and fast CC analyzer is difficult as there are many non-text components which can easily be confused with texts when analyzed individually.

Considering the difficulties mentioned above, a new approach is proposed to detect and localize texts efficiently

978-1-4673-4922-2/13/$31.00 ©2013 IEEE

from natural scene images by taking advantages from both region and CC-based methods. Experimental results on ICDAR 2003 dataset [2] shows high precision with the proposed method when compared with state-of-the-art methods.

A. Paper Organization

Section II discusses the related works in the field of text detection and localization. In Section III, the proposed method for text localization and detection are discussed. Section IV deals with the simulation results of the proposed method using ICDAR 2003 dataset. Finally Section V concludes this paper.

II. Related Works

Hundreds of approaches have been proposed for text object extraction in image and video documents since 1990s. Comprehensive surveys of text extraction approaches is proposed in [3] and [4].

A. Text Detection and Localization Approaches

In [4], the text detection and localization approaches are divided into two categories:

1) Region-Based Approaches: Text object typically has high edge densities, similar edge heights, big edge gradient magnitudes and large edge gradient direction variations. This occurs due to the fact that text is composed of several aligned characters with similar sizes and sharp contrast to background. Usually, morphological operation and block-based dividing and merging are used to generate candidate text regions based on local edge information and spatial and geometrical constraints are used to remove false alarms. In edge detection stage, background edges are suppressed by local thresholding. The boundaries of text objects are located by iteratively computing the horizontal and vertical projections of the edge map in the text localization stage.

Color-based approaches are based on the observation that the color of text object is homogeneous and different from the background colors. The approaches usually first extract the regions with homogenous colors based on color similarity using clustering method, intensity histogram, or binarization method, then the candidate text regions are localized by spatial information and geometrical constraints. Text objects are rich of corners, which are typically uniform distributed over text regions. Based on this property, many corner-based approaches are proposed to separate text from other objects. Block-based corner density and morphological dilation based corner merging are often employed to generate candidate text regions using extracted corners. By dilation, merging and corner refinement candidate text regions are formed. Densities of corner, edge, ratio of the vertical and horizontal edge densities and overall edge maps are computed for finding the edges

2) Texture-Based Approaches: Besides the region properties, the distinct textural properties between text and background can also provide important information for text extraction. For the approaches in this category, statistical features, frequency transforms, Gabor filters, and machine learning based methods are often used to describe and distinguish the texture of text and background. It is known that a region with rich texture in spatial domain has high frequencies in frequency domain. Taking advantage of this fact, many texture-based approaches separate text by localize high frequency regions in frequency domain. Fourier Transform (FT), Discrete Cosine Transform (DCT) and Discrete Wavelet Transform (DWT) are most used for this purpose.

The approaches proposed in [5], uses HL, LH, and HH subbands of DWT and K-means method for text detection and localization, but different features are used for clustering. Wavelet coefficients are selected as the clustering feature in [5] to localize text regions with high valued coefficients. Instead of using SVM to classify text objects, [6] SVM can extract features within its own architecture efficiently based on kernel functions and present an approach for analysis of textural properties. The raw pixels intensities in the original image are input to the SVM without external texture feature extraction module. The approach is limited because it can detect only text objects in horizontal rectangle shapes.

Text localization in scene images using Conditional Random Field (CRF) is discussed in [7] & [8]. Text regions are detected by HOG and WaldBoost algorithm. The confidence map showing the probability of a region containing text are computed using the WaldBoost output based on a boosted classifier calibration method. CCs are obtained by Niblack′s binarization algorithm. Then, CRF is adopted for connected component analysis. Based on the neighborhood graph of CCs, unary and binary features that represent the properties of CCs and component neighboring relationship are used to construct a CRF model.

III. Proposed Text Localization and Detection

An unsupervised method to detect and localize text objects in images is introduced. This method is based on a new approach in which four scale decomposition and morphological methods for finding the edges of the image is incorporated. The proposed method is simple, computationally very fast and robust to the size, orientation, font, color of text and can efficiently discriminate text objects from other objects.

A. Scale Space

Real-world objects are multiscale in nature so objects has to be perceived in different ways depending on the scale of observation. For developing automatic algorithms for interpreting images of unknown scenes, which all scales are relevant cannot be determined before. So the solution is to consider representations at all scales simultaneously.

Image operators can be used as basis to solve a large variety of visual tasks including feature classification, feature detection, stereo matching, motion descriptors, image-based recognition, and shape cues. Scale-space representation can be complemented with a module for automatic scale selection based on maximization of normalized derivatives over scales, early visual modules can be made scale invariant. In this way, visual modules can adapt automatically to the unknown scale variations that may occur because of objects and objects with varying distances to the camera as well as substructures of varying physical size.

Scale-space theory [9], developed by the computer vision is a framework for multi-scale signal representation, image

processing and signal processing communities with complementary motivations from biological vision and physics. A formal theory for handling image structures at different scales, by representing an image as one-parameter family of smoothed images, parametrized by the size of the smoothing kernel used for suppressing fine-scale structures, the scale-space representation. The parameter t in this family is referred to as the scale parameter, with the interpretation that image structures of spatial size smaller than $\sqrt{t}$ about have largely been smoothed away in the scale-space level at scale t. The linear (Gaussian) scale space is the main type of scale space, which has wide application. Visual operations are made scale invariant in this framework, necessary for dealing with size variations that occur in image data, because real-world objects may differ in sizes and in the distance between the camera and the object might be unknown and varies depending on the circumstances.

Scale space applies to signals of arbitrary numbers of variables as well. The most common case in the literature applies to two-dimensional images. For a given image $f(x, y)$, its linear (Gaussian) scale-space representation is a family of derived signals $L(x, y; t)$ defined by the convolution of $f(x, y)$ with the Gaussian kernel

$$g(x, y; t) = \frac{1}{2\Pi t} e^{-(x^2+y^2)/2t} \tag{1}$$

such that

$$L(., .; t) = g(., .; t) * f(., .), \tag{2}$$

where the semicolon in the argument of L implies that convolution is performed on the variables x and y, The scale parameter t after the semicolon just indicates which scale level is being defined. This definition of L works for a continuum of scales $t \geq 0$, but typically only a finite discrete set of levels in the scale-space representation would be actually considered [10].

Gaussian filter is used because when faced with the task of generating a multi-scale representation there is no low-pass type filter g with a parameter t which determines its width to be used to generate a scale space. Smoothing filter should not introduce any new spurious structures at coarse scales that do not correspond to simplifications of corresponding structures at finer scales.

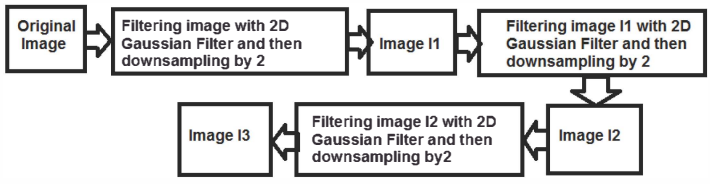

Fig. 2: Architecture of Scale Decomposition

A 2-D Gaussian low pass filter is created to filter input image. The filtered image is downsampled by 2 and the output is passed through the Gaussian LPF and the same procedure is repeated for two more stages as shown in Figure 2.

Scale space decomposition is used for efficiently finding out the edges during the subsequent edge detection operations. The kernels in the four directions are initialized as Kernel 0, Kernel 45, Kernel 90,and Kernel 135 as shown in equations (3) - (6).

$$\text{Kernel } 0 = \begin{bmatrix} -1 & -1 & -1 \\ 2 & 2 & 2 \\ -1 & -1 & -1 \end{bmatrix} \tag{3}$$

$$\text{Kernel } 45 = \begin{bmatrix} -1 & -1 & 2 \\ -1 & 2 & -1 \\ 2 & -1 & -1 \end{bmatrix} \tag{4}$$

$$\text{Kernel } 90 = \begin{bmatrix} -1 & 2 & -1 \\ -1 & 2 & -1 \\ -1 & 2 & -1 \end{bmatrix} \tag{5}$$

$$\text{Kernel } 135 = \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix} \tag{6}$$

This results in a 4-scale decomposition and scaled Images can be represented as an Image Pyramid. Figure 3 shows a 120×160 image sub-sampled upto 15×20.

Fig. 3: A 120×160 image sub-sampled upto 15×20

The scaled images are filtered with the four direction kernels for finding the scaled edges as shown in Figure 4. All the scale edges are added up together to get the total scale edges.

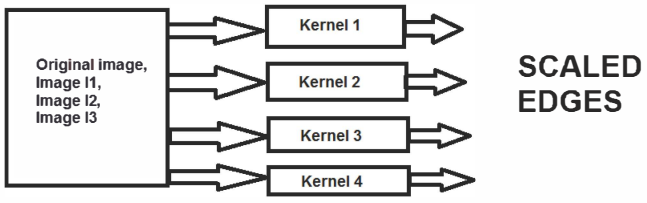

Fig. 4: Block Diagram for finding the Scaled Edges

Figure 5 shows the output of Kernel 1 through Kernel 4 of the original image, by looking at the images it is concluded that the vertical edges found by Kernel 3 is the most useful image of them all, this is because texts have vertical edges. So further analysis of the vertical edges is done for detecting and localizing the texts in images.

For finding the total edges in the vertical direction, the vertical direction edges of all the scaled images are added up to

Fig. 5: Output of Original image of size 120×180 after passing through the 4 Kernels

get the total edge in the vertical direction. The same operation is done for the other three directions as well thus obtaining the total scale edges.

B. Morphological Operations

Morphology means form and structure, and in this contexts, its referred to extraction of image components that are useful in the representation and description of region shape, such as skeletons, boundaries etc [11]. Morphology provides a unified and powerful approach to numerous image processing problems. Gray scale digital images can be represented as sets whose components are in Z^3 plane. First two elements are the coordinates (x,y) of a pixel, and the third corresponds to its discrete gray-level value. The morphological operations discussed are : 1) Dilation 2) Erosion 3) Opening and Closing 4) Thinning.

1) Dilation : With A and B as sets in Z^2, the dilation of A by B, denoted as $A \oplus B$, is defined as

$$A \oplus B = \{z|(\hat{B})_z \cap A \neq \oslash\}, \tag{7}$$

This equation is obtained by the reflection of B about its origin and shifting this reflection by z. The dilation of A by B is the set of all displacements, z, such that B and A overlap by at least one element. Based on this interpretation equation is rewritten as

$$A \oplus B = \{z|[(\hat{B})_z \cap A] \subseteq A\}, \tag{8}$$

Set B is commonly referred to as the structuring element in dilation, as well as in other morphological operations.

2) Erosion : For sets A and B in Z^2 the erosion of A by B is denoted by $A \ominus B$, and is defined as

$$A \ominus B = \{z|(B)_z \subseteq A\}, \tag{9}$$

This indicates that the erosion of A by B is the set of all points z such that B, translated by z, is contained in A.

3) Opening and Closing : Dilation expands an image while erosion shrinks it. Opening generally smoothes the objects contour, breaks narrow isthmuses, and eliminates thin protrusions. Closing also tends to smooth sections of contours but, as opposed to opening, it generally fuses narrow breaks and long thin gulfs, eliminates small holes and fills gaps in the contour. The opening of set A by structuring element B, denoted as $A \circ B$ is defined as

$$A \circ B = (A \ominus B) \oplus B, \tag{10}$$

Thus, the opening A by B is the erosion of A by B, followed by a dilation of the result by B. Similarly, the closing of set A by structuring element B, denoted by $A \bullet B$ is defined as

$$A \bullet B = (A \oplus B) \ominus B, \tag{11}$$

which, in other words, closing of A by B is simply the dilation of A by B, followed by the erosion by B.

4) Thinning : Thinning is a morphological operation that is used to remove selected foreground pixels from binary images, somewhat like erosion or opening. The thinning operation is calculated by translating the origin of the structuring element to each possible pixel position in the image, and at each such position comparing it with the underlying image pixels. If the background and foreground pixels in the structuring element exactly match background and foreground pixels in the image, then the image pixel underneath the origin of the structuring element is set to background (zero). Otherwise it is left unchanged. Thinning is the dual of thickening, i.e. thickening the foreground is equivalent to thinning the background.

The Thinning of a set A by a structuring element B, denoted by $A \otimes B$, can be defined in terms of the hit-or-miss transform:

$$A \otimes B = A - (A \odot B) = A \cap (A \odot B)^c \tag{12}$$

C. Strong and Weak Edges

For selecting the strong and weak edges global thresholding is done, thereby converting the intensity image to a binary image. All the intensities greater than the threshold is shown as 1 and the remaining as 0.

1) Otsu's Thresholding Method: In computer vision and image processing, Otsu's method is used to automatically perform histogram shape-based image thresholding. for the reduction of a gray-level image to a binary image. The algorithm assumes that the image to be thresholded contains two classes of pixels or bi-modal histogram (e.g. foreground and background) then calculates the optimum threshold separating those two classes so that their combined spread (intra-class variance) is minimal.

In Otsu's method exhaustively search is done for the threshold that minimizes the intra-class variance (the variance within the class), defined as a weighted sum of variances of the two classes:

$$\sigma_\omega^2(t) = \omega_1(t)\sigma_1^2(t) + \omega_2(t)\sigma_2^2(t) \tag{13}$$

Weighted ω_i are the probabilities of the two classes separated by a threshold t and σ_i^2 variances of these classes.

Otsu shows that minimizing the intra-class variance is the same as maximizing inter-class variance.

$$\sigma_b^2(t) = \sigma^2 - \sigma_w^2(t) = \omega_1(t)\omega_2(t)[\mu_1(t) - \mu_2(t)]^2 \tag{14}$$

where ω_i is the class probabilities and class means μ_i. The class probability $\omega_1(t)$ is computed from the histogram as t:

$$\omega_1(t) = \sum_0^t p(i) \tag{15}$$

$$\mu_1(t) = \sum_{0}^{t} p(i)x(i) \tag{16}$$

where $x(i)$ is the value at the center of the ith histogram bin. After computing two maximums $\sigma^2_{b1}(t)$ and $\sigma^2_{b2}(t)$. The desired threshold level is obtained by taking the average of the two. After thresholding the strong edge as shown in Figure 7 is obtained

Fig. 6: Please do not disturb sign board

To make the edges thick a morphological operation known as dilation is done. Figure 7 shows the strong and weak edges of the Figure 6.

Fig. 7: Strong and Weak Edges of 'Please do not disturb sign board'

Gray scale dilation of f by b, denoted as $f \oplus b$, is defined as,

$$\begin{aligned}(f \oplus b)(s,t) &= max\{f(s-x,t-y)+b(x,y)| \\ &\quad (s-x),(t-y) \in D_f; (x,y) \in D_b\}\end{aligned} \tag{17}$$

where D_f and D_b are the domains of f and b functions. Similarly erosion, opening and closing for gray images are denoted as,

$$\begin{aligned}(f \ominus b)(s,t) &= min\{f(s+x,t+y)-b(x,y)| \\ &\quad (s+x),(t+y) \in D_f; (x,y) \in D_b\}\end{aligned} \tag{18}$$

where D_f and D_b are the domains of f and b functions.

$$f \circ b = (f \ominus b) \oplus b \tag{19}$$

$$f \bullet b = (f \oplus b) \ominus b \tag{20}$$

Edges of the pixels which are less than one-fourth of the total vertical length of the image is removed, by performing closing morphological operation. To find the weak edges the absolute difference between the closed and dilated images are taken. The weak edges are found out after performing the thresholding operation of the difference that was found earlier. Adding up the strong and weak edges in the vertical direction

Fig. 8: Thinning, Short and Refined Edges

to get the sum of edges in the vertical direction. Thinning of the edges result in thinned image.

Short edges are then found from the thinned image by finding the pixels were the major axis length is less than at least 20% of the length of the image.

Short edges are thickened by the dilation operation to get a candidate Image. Multiplying this image with the total edges in all direction results in our refined edges. The thinned image along with the short and the refined edge image is shown in Figure 8.

Feature map image is found out which is the AND operation of the 0 and 90 degree and then ORing that with the AND of 45 and 135 degrees, and this image is dilated to remove unwanted noises. For removing the unwanted noises in the image which have less areas, those pixels whose area is less than 20% of the maximum area of the image is removed. For getting the image without any unwanted noise, those pixels which has an aspect ratio greater than 6 is also removed.

The final text detected image is obtained by multiplying the above image with the binarized original image. The final image is shown in Figure 9.

Fig. 9: The Final Output Image

IV. Experimental Results

The experimental results of the proposed detection and localization method is shown here. Images from the ICDAR 2003 dataset and the corresponding experimental results are shown. The proposed algorithm is implemented in matlab simulation language (MATLAB 7.6) with image processing tool box on a PC with Intel Core i5 CPU at 2.67GHz and 4 GB RAM under Microsoft 7 Home Basic. The execution time for an image is within 10 second for a 240×320 image and can vary slightly depending on the size of the image. Figure 10 to Figure 14 shows the detected text images and the corresponding detection results of some images in this dataset. It can be observed from the results that the proposed method

Fig. 10: Peacocks Board Sign

can detect text objects successfully. The results show that the proposed detection method performs with high precision considering the simplicity and the amount of computational time required.

Fig. 11: Emblem of Ford Motors

Fig. 12: Sign Of Essex County

Fig. 13: Sign direction of Station Car Parking

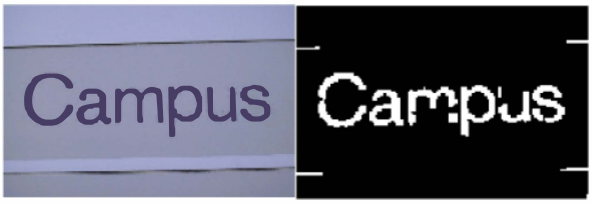

Fig. 14: Campus Hoarding Board

V. Conclusion

Text objects occurring in images can provide much useful information for content based information retrieval and indexing applications, because they contain much semantic information related to the documents contents. However, extracting text from images is a very difficult task due to the varying font, size, color, orientation, and deformation of text objects. Although a large number of text extraction approaches have been reported in the literature, no specific designed text model and character features are presented to capture the unique properties and structure of characters and text objects. In this work, a new unsupervised method to detect and localize the text objects occurring in image documents based on scale space and morphological operations for the edge detection is proposed. The assumptions made that of aspect ratio, short edges, and the area has made the proposed algorithm efficient. It is observed that the proposed text detection method can detect the text objects with various fonts, sizes, colors, and orientations efficiently.

References

[1] www.blog.flickr.net/en/2011/08/04/6000000000/, 2012.

[2] www.algoval.essex.ac.uk/icdar/Datasets.html, 2012.

[3] D. Chen, J. Luettin, and K. Shearer, "A survey of text detection and recognition in images and videos," *Institut Dalle Molled Intelligence Artificielle Perceptive (IDIAP) Research Report, IDIAP-RR 00-38*, 2000.

[4] K. Jung, K. In Kim, and A. K Jain, "Text information extraction in images and video: a survey," *Pattern recognition*, vol. 37, no. 5, pp. 977–997, 2004.

[5] J. Gllavata, R. Ewerth, and B. Freisleben, "Text detection in images based on unsupervised classification of high-frequency wavelet coefficients," in *Proceedings of the 17th International Conference on Pattern Recognition, ICPR 2004.*, vol. 1, 2004, pp. 425–428.

[6] K. Kim, K. Jung, and J. Kim, "Texture-based approach for text detection in images using support vector machines and continuously adaptive mean shift algorithm," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 12, pp. 1631–1639, 2003.

[7] Y. Pan, X. Hou, and C. Liu, "Text localization in natural scene images based on conditional random field," in *10th International Conference on Document Analysis and Recognition, ICDAR'09.*, 2009, pp. 6–10.

[8] X. Hou, Y. Pan, and C. Liu, "A hybrid approach to detect and localize texts in natural scene images," *IEEE Transactions on Image Processing*, vol. 20, no. 3, pp. 800–813, 2011.

[9] A. Ralston, E. Reilly, and D. Hemmendinger, *Encyclopedia of computer science*. Van Nostrand Reinhold, 1993, vol. 536.

[10] www.en.wikipedia.org/w/index.php?oldid=509364125, 2012.

[11] R. Gonzalez and R. Woods, *Digital image processing*. Prentice Hall, 2002.

[12] M. 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, no. 2, pp. 243–255, 2005.