Image Text Detection Using a Bandlet-Based Edge Detector and Stroke Width Transform

Ali Mosleh[1]
mos_ali@encs.concordia.ca
Nizar Bouguila[2]
bouguila@ciise.concordia.ca
A. Ben Hamza[2]
hamza@ciise.concordia.ca

[1] Department of Electrical and Computer Engineering
[2] Concordia Institute for Information Systems Engineering
Concordia University
Montréal, QC, Canada

Abstract

In this paper, we propose a text detection method based on a feature vector generated from connected components produced via the stroke width transform. Several properties, such as variant directionality of gradient of text edges, high contrast with background, and geometric properties of text components jointly with the properties found by the stroke width transform are considered in the formation of feature vectors. Then, k-means clustering is performed by employing the feature vectors in a bid to distinguish text and non-text components. Finally, the obtained text components are grouped and the remaining components are discarded. Since the stroke width transform relies on a precise edge detection scheme, we introduce a novel bandlet-based edge detector which is quite effective at obtaining text edges in images while dismissing noisy and foliage edges. Our experimental results indicate a high performance for the proposed method and the effectiveness of our proposed edge detector for text localization purposes.

1 Introduction

Digital images and videos are nowadays increasingly used due to the rapid development of image capturing devices. The need of information retrieval from images led to semantic content analysis techniques. A slew of such techniques are specialized in extracting text embedded in images since it is a vital source of semantic information. A robust text detection step is the basic requirement for a scheme designed to extract text information from images. Text detection is still a challenging issue due to unconstrained color, sizes, alignments of characters, lighting and also various shapes of fonts, even though various methods have been proposed in the past years [6, 9]. Existing text detectors are broadly classified into two main groups: texture (also called region) based and connected component (CC) based methods.

Texture-based methods scan the image at a number of scales and consider the embedded text as a particular texture pattern distinguishable from other parts of the image and its background. Basically, features of various regions of the image are retained. Then, the presence of text is identified by either a supervised or an unsupervised classifier. Finally, the neighboring text region candidates are merged based on some geometric features to generate text

http://dx.doi.org/10.5244/C.26.63

blocks. As examples of such methods, the technique introduced in [1] applies Sobel edge detector in all Y, U, and V channels, then invariant features such as edge strength, edge density, and edge's horizontal distribution are considered. The method utilizes different thresholds in the edge detection to define text-area enhancement operators. The method introduced in [10] produces a statistical-based feature vector using the Sobel edge map and applies k-means algorithm to classify image regions into text and non-text parts. Assuming that the horizontal gradient value of text regions is higher than that of other parts of the image, the proposed method in [27] thresholds the variance of gradient values to identify text-regions. A support vector machine (SVM) classifier is used in [7] to generate text maps from the gray-level features of all local areas. The method extracts the features through each layer of image pyramids. The proposed method in [13] also takes advantage of image pyramids to find local thresholds to detect text areas. Frequency domain is shown to be practical in text-region classifications. A k-means classification is applied in wavelet domain in [5] in order to detect the horizontally aligned texts in an image. The proposed technique in [8] employs the first and second-order moments of wavelet coefficients of local regions as features, then a classification is performed by means of neural networks. In the same vein, the proposed method in [30] applies frequency domain coefficients obtained by discrete cosine transform (DCT) to extract features. By thresholding filter responses, text-free regions are discarded and the remaining regions are grouped as segmented text regions.

CC-based methods stem from observations that text regions share similar properties such as color and distinct geometric features. At the same time, text regions have close spatial relationship. Therefore, based on such properties they are grouped together and form CCs. The method introduced in [25] finds candidate text regions utilizing Canny edge detector, then a region pruning step is carried out by means of an adjacency graph and some heuristic rules based on local components features. Candidate CCs are extracted by the proposed method in [10] based on edge contour properties, then text-free components are pruned by analysis of wavelet coefficients. In order to find CCs, an adaptive binarization is applied in [3]. Statistical analysis of text regions is performed to determine which image features are reliable indicators of text. This is done by considering a large training set which consists of text images. In fact, the feature response of the candidate CCs must be similar to the text images. The effectiveness of this method is reported in ICDAR 2005 results [11]. A useful operator is defined in [4] to find stroke width of each image pixel. The stroke width transform (SWT) image is generated by shooting rays along the direction of each edge pixel's gradient. Then the SWT values are grouped based on their ratios in order to produce CCs. The text-candidate CCs are selected by applying some rules such as aspect-ratio, diameter and variance of stroke width of each component. In [23] the CCs are found by k-means clustering in the Fourier-Laplacian domain. Then, the candidate CCs are filtered by test string straightness and edge density features. This method is not only practical for horizontally aligned texts but also for any arbitrary oriented text. A CC-based algorithm is introduced in [2], which employs Maximally Stable Extremal Regions (MSER) as the basic letter candidates. Then, by using geometric and stroke width information non-text CCs are excluded.

A number of existing methods are not categorized in the aforementioned two groups. As an example, the proposed method in [18] is a hybrid technique whose first step detects text regions in each layer of image pyramid and projects the text confidence and scale information back to the original image followed by a local binarization to generate candidate text components. Next, a CRF model filters out non-text components and then a learning-based minimum spanning tree (MST) is used to link the CCs. Sparse representation is also applied in the field of text detection. The introduced method in [29] benefits from two learned

discriminative dictionaries, one for document images, and another for natural images to distinguish between text regions and background ones in an input image. The dictionaries are generated using the platform introduced in [14]. Also, the text and non-text regions are distinguished by the reconstruction error function defined in [14] for the dictionary patches and the original image patches.

The general scheme of our proposed method consists in producing the image edge map and then finding CCs based on SWT guided by the generated edge map. Next, precise feature vectors are formed using the properties of CCs from SWT and pixel domain. An unsupervised clustering is performed on the image CCs to detect the candidate text CCs. Finally, text candidate components are linked to form text-words. The method is considered as a CC-based technique and the contribution is twofold: 1) Since accurate edge maps drastically enhance SWT results, a precise edge detection approach adaptive to text-regions is proposed by employing the bandlet transform. 2) A feature vector based on text properties and stroke width values is employed in k-means clustering in order to detect text CCs.

The rest of this paper is organized as follows: the bandlet-based edge detector is discussed in Section 2. The proposed text detection method is explained in Section 3. In Section 4, the experimental results are provided. Finally, Section 5 presents our conclusions.

2 Edge Detection Using Bandlets

The bandlet framework achieves an effective geometric representation of texture images. It is essential in the case where we need to extract image singularities and strong continuations such as the ones that appear in the image edges. Hence, we employ the bandlet transform and propose a novel edge detector that works well in extracting text edges from images.

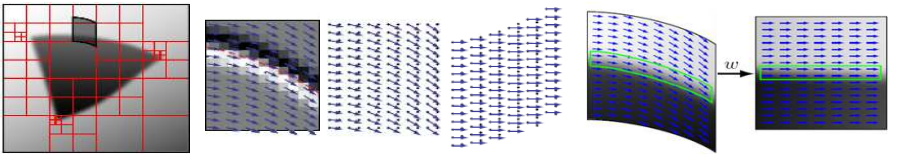
2.1 Bandlet transform

Although geometric regularity along image edges is an anisotropic regularity, conventional wavelet bases can only exploit the isotropic regularity on square domains. An image can be differentiable in the direction of the tangent of an edge curve even though the image may be discontinuous across the curve. Bandlet transform [16] exploits such anisotropic regularity. Bandlet bases construct orthogonal vectors elongated in the direction of the maximum regularity of a function such as the one shown in Fig. 1(b) by a red dash. The earlier bandlet bases [20, 21] have been improved by a multi-scale geometry defined over wavelet coefficients [15, 24]. Indeed, bandlets are anisotropic wavelets warped along the geometric flow.

Considering the Alpert transform as a polynomial wavelet transform adapted to an irregular sampling grid similar to Fig. 1(d), one can obtain vectors that have vanishing moments on this irregular sampling grid. This is the principal need to approximate warped wavelet coefficients similar to Fig. 1(e). Only a few vectors of Alpert basis can efficiently approximate a vector corresponding to a function with anisotropic regularity. The *bandletization* using wavelet coefficients is defined by

$$b^k_{j,l,n}(x) = \sum_p a_{l,n}[p]\,\psi^k_{j,p}(x), \tag{1}$$

where $a_{l,n}[p]$ are the coefficients of the Alpert transform and k represents wavelet orientation. These coefficients strictly depend on the local geometric flow. Bandlet coefficients are generated by inner products $\langle f, b^k_{j,l,n}\rangle$ of the image f with the bandlet functions $b^k_{j,l,n}$. The set of wavelet coefficients are segmented in squares S for polynomial flow approximation of



(a) (b) (c) (d) (e)

Figure 1: Bandlet transform [16, 24]. (a) Dyadic segmentation based on local directionality of the image. (b) A sample bandlet segmentation square that contains a strong regularity function shown by the red dash. (c) Geometric flow and sampling position. (d) Sampling position adapted to the warped geometric flow. (e) Illustration of a warping example.

the geometry. For each scale 2^j and orientation k, the segmentation is carried out using a recursive subdivision in dyadic squares as illustrated in Fig. 1(a). A square S should be further subdivided into four sub-squares if there is still a geometric directional regularity in the square. Apparently, only for the edge squares, the adaptive flow is needed to be computed to obtain the bandlet bases. The geometry of an image evolves through scales. Therefore, for each scale 2^j of the orientation k a different geometry Γ_j^k is chosen. The set of all geometries $\{\Gamma_j^k\}$ represents the overall geometry of an image. At the end of this bandletization process we have a multi-scale, low and high-pass filtering structure similar to the wavelet transform, strictly adapted to local directionality of the image. This elegant platform is suitable to distinguish strong continuations such as edges and eliminate singularities such as foliage areas in natural images. For more details about bandlets the reader is referred to [16].

2.2 Edge detection algorithm

As discussed before, the bandlet transform effectively represents the geometry of an image. We take advantage of this representation and propose an edge detection algorithm that can be used effectively in text-detection techniques. On the other hand, it has been shown in [17, 28] that finding local maxima of wavelet transform coefficients is similar to the multi-scale Canny edge detector operator. Fig. 2(a) presents a sample 1D signal whose wavelet transform (high frequency part) in one scale is shown in Fig. 2(b). Finding the local maxima in the first derivative of high frequency coefficients of wavelet transform (Fig. 2(c)) is equivalent to edge positions in the original signal (image). Since the image coefficients are all warped along local dominant flows in the bandlet transform, the final bandlet coefficients generated for each segmentation square S have the form of approximation, and high-pass filtering values appear in the wavelet transform of a 1D signal like Fig. 2. We benefit from the bandlet-based resulting 1D high-pass frequency coefficients that are adapted to the directionality of the edge that exists in each segmentation square S in order to find a binary map of the edge positions in the image.

The bandlet transform is performed on the original image, and for each segmentation square S the bandlet coefficients are generated. For each S, the resulting coefficients are grouped in low-pass (approximation) and high-pass filtering results similar to the 1D wavelet transform. Since the approximation part consists of coarse information of the original signal, we discard it and only process the high-pass coefficients. The first-order derivatives of the fine-detail bandlet coefficients are computed. By applying a contextual filer, we find local maximum of the resulting gradient signal since many meaningful edges can be found in the local maxima of the gradient not only in the global maxima. Then, in order to improve the quality of the edge image a two level thresholding is employed.

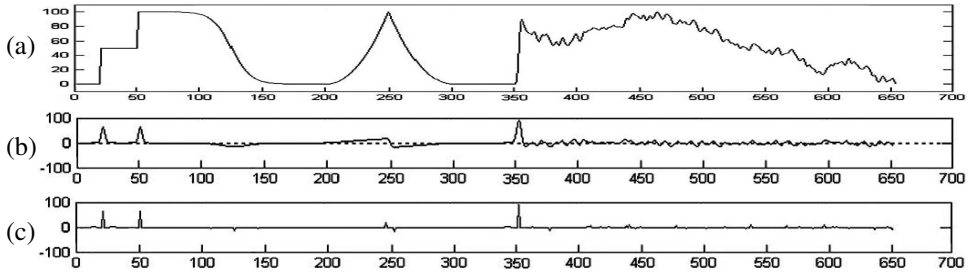


Figure 2: Using wavelets in edge detection. (a) Original signal. (b) One-scale 1D dyadic wavelet transform. (c) Derivatives of the wavelet transform coefficients of (b).

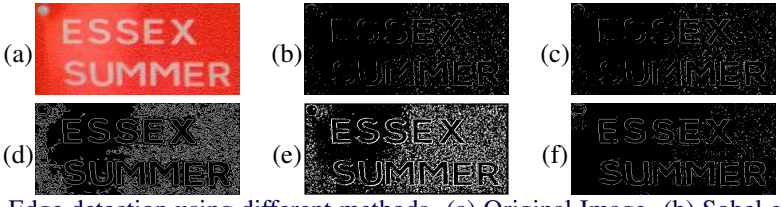


Figure 3: Edge detection using different methods. (a) Original Image. (b) Sobel edges. (c) Prewitt edges. (d) Canny edges. (e) Wavelet-based edges. (f) Bandlet-based edges.

For each point x_i in the gradient signal, we check if x_i is a local maximum and its value is greater than a threshold T. If so, x_i is kept as an edge point coefficient otherwise it will be discarded. Hence, a window with size $2L+1$ centered at x_i is set. Then, the binary indicator of edge points in the gradient signal is generated as follows:

$$M_i = \begin{cases} 1 & \text{if } g_i > T \ \wedge \ g_i > g_j, \ \forall j \in [i-L, i-1] \ \wedge \ g_i > g_j, \ \forall j \in [i+1, i+L] \\ 0 & \text{otherwise,} \end{cases} \tag{2}$$

where g_i represents the gradient value for x_i and g_j indicates gradient value of neighboring pixels of x_i that exist is the window. M is a map of local maxima of the gradient signal. The corresponding locations of 0's of M in the bandlet fine (high-pass) coefficients are set to 0, for all the bandlet squares S. Then, the inverse bandlet transform is performed in order to have the final edge locations of the original image.

Obviously, the quality of the edge map depends on the value of the threshold T. In order to ensure a high quality, a two-level thresholding is employed. First, the edge detection is performed using a low value for T and the edge image E_l is produced. The algorithm is performed another time utilizing a higher value for T to generate the edge image E_h. Apparently, E_l includes more edge pixels than E_h, which only includes significant edges. Also, all the edge pixels of E_h exist in E_l. A combination of E_h and E_l leads to more reasonable results. For each edge component C_{eh} that exists in E_h we inspect E_l and check if there is an edge component C_{el} in E_l that overlaps C_{eh}. If so, C_{el} is taken from E_l and saved in the finial image edge map.

Considering the bandlet transform structure strictly adapted to strong local pixel flows through a geometry-based dyadic segmentation, this edge detection scheme reveals reliable edge pixels. Moreover, since the regions consisting of sparse singularities such as noisy and foliage pixels, and the regions with various pixel intensities are eliminated in the bandlet geometric segmentation, the resulting edges are quite appropriate to localize text-edges embedded in the image. Fig. 3 shows edge detection results of four different methods including Sobel, Prewitt, Canny, wavelet and the proposed bandlet-based technique. The input image includes a text and noisy pixels. Our proposed edge detection approach shows considerably better results compared to the other methods.

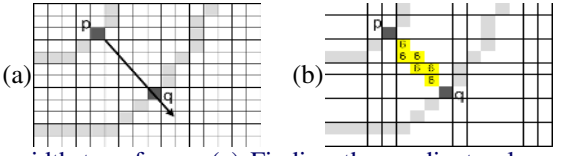


Figure 4: . Stroke width transform. (a) Finding the gradient value of edge pixel p and shooting a ray in its direction and finding an edge pixel q with opposite gradient direction on the ray. (b) Assigning the stroke width value to each pixel that lies on the ray.

Figure 5: Stroke width transform of a sample image. (a) Original Image. (b) SWT output using bandlet-based edges. (c) SWT output using Canny edges.

3 Text Detection Algorithm

Our text detection approach obtains features for CCs produced by SWT, then decides which CC is a text candidate using k-means clustering. The technique applies SWT introduced in [4] in order to obtain reasonable text CCs and text features. Therefore, this section starts with a discussion about SWT.

3.1 Stroke width transform (SWT)

The SWT value of each pixel is roughly the width of the stroke that contains the pixel. A stroke is defined as a part of the image that forms a band of constant width. In the first step, we find the edges of the input image by means of the proposed edge detection method (Sec. 2). Then, the gradient direction d_p of each edge pixel p is determined. A ray starting from p with the direction of d_p is considered and followed until it meets another edge pixel q. If the gradient direction d_q at edge pixel q is approximately opposite to d_p, the distance value of p and q is assigned to all the pixels that lie on the ray. Fig. 4(b) shows SWT values of sample pixels that lie on the ray shown in Fig. 4(a). SWT of a sample image is computed and shown in Fig. 5(b). This figure clearly shows how effective SWT can be in finding text regions in images. In order to demonstrate the effectiveness of the bandlet-based edge detector, SWT is calculated once again using conventional Canny edges as shown in Fig. 5(c). As mentioned before, the bandlet-based edge detector removes noisy and foliage regions from the edge map. Therefore, Fig. 5(b) presents a more practical SWT result compared to Fig. 5(c) in the case of text localization and detection.

Neighboring pixels are grouped together and form CCs if they have similar stroke width values. The traditional CC algorithm is not performed on a binary mask but on the SWT values with a different connection criterion. In the CC algorithm, 4-neighboring pixels are considered. Adjacent pixels are grouped if the ratio of their stroke width values is higher than 0.3 and lower than 3. Features of the produced CCs are used to find text candidates.

3.2 Unsupervised classification and refinement

We need to identify components that very likely contain text. Thus, we employ a set of rules and assumptions in order to make a feature vector for each component. Then, the feature

vectors are fed to k-means clustering to identify text components.

The first property of a text component is that the variance of stroke width values V_{SWT} in all the text components is not too large. A high value of V_{SWT} for a CC means the component consists of the pixels of a foliage region. The mean μ_{SWT} and median M_{SWT} values of each CC are also considered in order to find text components with the same stroke width, since almost all the characters of a word would have the same stroke width. Another important feature of a text component is that it is neither too long nor thin. Therefore, the ratio R_s of the component diameter and its median stroke width M_{SWT} are added to the feature vector.

Considering a sample character as a text CC, one observes that the gradient directions of edge pixels of the component vary significantly. In other words, a text component can have edge pixels with gradient directions ranging from 0 to 90 degree for character "I" for instance or 0 to 180 for "O", indicating a large range of directionality. So, we calculate the variance V_G of gradient directions of all the edge pixels of a CC and save it in the feature vector. Also, a text component has almost a symmetric distribution for the gradient directions of the edge pixels. This is due to the fact that a character has at least two sets of edge pixels roughly parallel to each other with opposite gradient directions. Therefore, we estimate having a symmetric distribution for the direction of edge pixels by computing the skewness SK_G for the gradient directions and add it to the features:

$$SK_G = \frac{\mu_3}{\sigma^3} = \frac{\frac{1}{n}\sum_{i=1}^{n}(g_i - \mu_G)^3}{(\frac{1}{n}\sum_{i=1}^{n}(g_i - \mu_G)^2)^{3/2}}, \tag{3}$$

where n is the total number of edge pixels in a CC, g_i is the gradient direction at edge pixel i and μ_G is the mean of gradient directions of the edge pixels of the CC. In fact, in this equation μ_3 and σ are the third moment about the mean and standard deviation of the gradient directions, respectively.

An important feature attributed to texts in images is their relatively high contrast with the background compared to other regions of the image. This is due to the nature of utilizing texts i.e, catching one's sight and conveying information. A scene text or a caption text in a video frame, for example, must have a strong contrast with the background since the producer of the text wanted them to stand out clearly. Thus, we consider this important property and use it in the feature vector. Typically, contrast is estimated by Weber formula: $C = (L_o - L_b)/L_b$, where L_o and L_b are the luminance of the object and its surrounding background, respectively. More complex contrast analysis can be found in [19, 26] by employing discrete cosine transform and wavelets. We simply use the local mean μ_L and standard deviation σ_L of the image intensity to estimate the contrast value of a CC with its background [22]; $C_L = \sigma_L/\mu_L$. C_L is computed for the intensity pixels that exist in the bounding box of a CC and added to its feature vector.

Finally, the bounding box itself must have a reasonable aspect-ratio for a text CC. Normally, the height of a text component is larger than its width and their aspect-ratio is not too large. So, we find the aspect-ratio R_{asp} of the bounding box of each CC and use it in the feature vector. The final feature vector of each CC has the following form:

$$\overrightarrow{F} = \{V_{SWT},\ \mu_{SWT},\ M_{SWT},\ R_s,\ V_G,\ SK_G,\ C_L,\ R_{asp}\} \tag{4}$$

The produced vectors $\overrightarrow{F}$ of all the CCs of the image are fed to a k-means scheme and consequently clustered into two groups, non-text and text components as shown in Fig. 6(a) for instance. In order to identify which cluster is associated to the texts and which is not, at the beginning of the process we append a sample text to the end of each input image. Hence,

Figure 6: Clustering of CCs. (a) Text and non-text CCs identification. (b) Merging text CCs to generate the final result.

the resulting cluster that contains the sample text components is considered as the group of text components and the rest of the components are discarded. In the last step, the remaining text components which are horizontally aligned and have reasonable distance to each other, for example as far as a character width, are grouped together and form the word components as shown in Fig. 6(b).

4 Experimental Results

We evaluated our approach on the ICDAR text locating contest dataset [11]. In Fig. 7 sample text detection results of our approach on ICDAR dataset are presented. The dataset contains 251 color images in various sizes from 307×93 to 1280×960. Along with the images, the dataset provides ground truth locations of the texts that exist in the images called targets to have a precise evaluation of the results of text detection techniques. The result of a text detection method in the form of a rectangle that bound a text in the image is called estimate hereafter. We followed the same evaluation scheme by means of *Precision* and *Recall* used in ICDAR competitions [11, 12]. *Precision* is the number of correct estimates divided by the total number of estimates. A method has a low precision if the number of text bounding rectangles is too large. *Recall* is defined as a ratio of the number of correct estimates and the total number of targets. Hence, a method that results in a large number of incorrect rectangles has a low recall score. The results of a text locating system are not as exact as human tagged locations. Therefore, a match m_p between two rectangles defined as the area of their intersection divided by the area of the minimum bounding box containing both rectangles is used. The value of m_p is zero for two rectangles without any intersection and one for exactly alike rectangles. For each rectangle in the set of estimates, the closest match in the set of targets is found, and vice versa. Hence, the best match $m(r;R)$ for a rectangle r in a set of rectangles R is defined as

$$m(r;R) = max\{m_p(r;\acute{r})|\acute{r} \in R\}. \tag{5}$$

Then, *precision* and *recall* are defined as

$$Precision = \frac{\sum_{r_e \in E} m(r_e,T)}{|E|}, \quad Recall = \frac{\sum_{r_t \in T} m(r_t,E)}{|T|}, \tag{6}$$

where T and E are the sets of target (ground truth) and estimated boxes, respectively. These two measures are combined into a single quality measure f with a weight factor α set to 0.5:

$$f = \frac{1}{\frac{\alpha}{Precision} + \frac{1-\alpha}{Recall}}. \tag{7}$$

Figure 7: Sample text detection results using the proposed technique on the ICDAR dataset.

Method	Precision	Recall	f
bandlet edges	0.76	0.66	0.71
wavelet edges	0.71	0.59	0.65
Canny edges	0.67	0.51	0.58
Sobel edges	0.53	0.56	0.53

Table 1: Performance of the proposed text detection method employing different edge detectors before STW.

Method	Precision	Recall	f
Our Method	0.76	0.66	0.71
Zhao [29]	0.64	0.65	0.65
Epshtein [4]	0.73	0.60	0.66
Gllavata [5]	0.44	0.46	0.46

Table 2: Performance of different text detection methods performed for the images of the ICDAR dataset.

In the first experiment, we employed other edge detection methods in our text detection scheme instead of the proposed bandlet-based edge detection approach (Sec. 2). Table 1 shows *Precision*, *Recall* and f values obtained by our text detection approach for the images of ICDAR dataset using Sobel, Canny, conventional wavelets and bandlet transform edge detection techniques. In this table the highest values of *Precision*, *Recall* and f are attributed to the method which employs our proposed bandlet-based edge detector. The result of our proposed approach has been compared with other methods as well. Table 2 shows the list of methods used for comparison and their *Precision*, *Recall* and f values for images of ICDAR dataset. The proposed method has a better performance compared to the other listed methods. Specifically, our approach outperforms the SWT-based method introduced in [4] already shown to have a good performance compared to several other existing methods [4] including the participating algorithms in ICDAR 2003 [12] and ICDAR 2005 [11].

5 Conclusions

In this paper, we introduced an unsupervised clustering scheme for text detection that can be considered as a connected component-based text detection technique. A feature vector based on properties extracted from stroke width transform connected components, distinct characteristics of text components that exist in images and their general geometry is formed. The components' feature vectors are fed to k-means clustering in order to separate possible text components from non-text ones. Then, based on the alignments of the found text components, locations of the text-words are determined. The accuracy of our algorithm depends

on how precisely connected components are generated in the SWT domain. At the same time, SWT can be carried-out well only if edge locations are revealed properly. Therefore, we employed the effectiveness of bandlets in representing local geometry properties and introduced a novel edge detection approach which is quite practical in the case of finding edge locations of texts embedded in various types of images and consequently generating stroke width values of texts. The experimental results indicate a considerable performance for both the proposed edge detector and the text detection scheme.

References

[1] M. Cai, J. Song, and M.R. Lyu. A new approach for video text detection. In *IEEE International Conference on Image Processing (ICIP)*, pages I–117–I–120, 2002.

[2] H. Chen, S.S. Tsai G., Schroth, D.M. Chen, R. Grzeszczuk, and B. Girod. Robust text detection in natural images with edge-enhanced maximally stable extremal regions. In *IEEE International Conference on Image Processing (ICIP)*, pages 2609–2612, Sep. 2011.

[3] X. Chen and A.L. Yuille. Detecting and reading text in natural scenes. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages II–366–II–373, June-2 July 2004.

[4] B. Epshtein, E. Ofek, and Y. Wexler. Detecting text in natural scenes with stroke width transform. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2963–2970, June 2010.

[5] J. Gllavata, R. Ewerth, and B. Freisleben. Text detection in images based on unsupervised classification of high-frequency wavelet coefficients. In *International Conference on Pattern Recognition (ICPR)*, volume 1, pages 425–428, Aug. 2004.

[6] K. Jung, K. I. Kim, and A. K. Jain. Text information extraction in images and video: a survey. *Pattern Recognition*, 37(5):977 – 997, 2004.

[7] K.I. Kim, K. Jung, and J. H. 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*, 25(12):1631 – 1639, Dec. 2003.

[8] H. Li, D. Doermann, and O. Kia. Automatic text detection and tracking in digital video. *IEEE Transactions on Image Processing*, 9(1):147–156, Jan 2000.

[9] J. Liang, D. Doermann, and H. Li. Camera-based analysis of text and documents: a survey. *International Journal on Document Analysis and Recognition*, 7(2):84–104, Jul. 2005.

[10] C. Liu, C. Wang, and R. Dai. Text detection in images based on unsupervised classification of edge-based features. In *Eighth International Conference on Document Analysis and Recognition*, pages 610–614, 2005.

[11] S.M. Lucas. Icdar 2005 text locating competition results. In *Eighth International Conference on Document Analysis and Recognition (ICDAR)*, pages 80–84, Aug.-Sep. 2005.

[12] S.M. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, and R. Young. Icdar 2003 robust reading competitions. In *Seventh International Conference on Document Analysis and Recognition (ICDAR)*, pages 682 – 687, Aug. 2003.

[13] 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*, 15(2):243–255, Feb. 2005.

[14] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Discriminative learned dictionaries for local image analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, June 2008.

[15] S. Mallat and G.Peyre. Surface compression with geometric bandelets. *ACM Transactions on Graphics*, 24:601–608, July 2005.

[16] S. Mallat and G. Peyre. A review of bandlet methods for geometrical image representation. *Numerical Algorithms*, 44:205–234, Mar. 2007.

[17] S. Mallat and S. Zhong. Characterization of signals from multiscale edges. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14:710–732, 1992.

[18] Y-F. Pan, X. Hou, and C-L. Liu. A hybrid approach to detect and localize texts in natural scene images. *IEEE Transactions on Image Processing*, 20(3):800 –813, Mar. 2011.

[19] E. Peli. Contrast sensitivity function and image discrimination. *Journal of Optical Society of America*, 18(2):283–293, nov. 2001.

[20] E. Le Pennec and S. Mallat. Sparse geometric image representations with bandelets. 14:423–438, Apr. 2005.

[21] E. Le Pennec and S.Mallat. Bandelet image approximation and compression. *SIAM Multiscale Model. Simul.*, 4:992–1039, 2005.

[22] E. Reinhard, P. Shirley, M. Ashikhmin, and T. Troscianko. Second order image statistics in computer graphics. In *Proceedings of the 1st Symposium on Applied perception in graphics and visualization*, pages 99–106, 2004.

[23] P. Shivakumara, T. Q. Phan, and C. L. Tan. A laplacian approach to multi-oriented text detection in video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(2):412 –419, Feb. 2011.

[24] S.Mallat and G. Peyre. Orthogonal bandlets bases for geometric image approximation. *Commun. Pure Appl. Math.*, 61:1173–1212, Sep. 2008.

[25] H. Takahashi and M. Nakajima. Region graph based text extraction from outdoor images. In *Third International Conference on Information Technology and Applications (ICITA)*, volume 1, pages 680–685, July 2005.

[26] J. Tang, J. Kim, and E. Peli. Image enhancement in the jpeg domain for people with vision impairment. *IEEE Transactions on Biomedical Engineering*, 51(11):2013–2023, nov. 2004.

[27] E. K. Wong and M. Chen. A new robust algorithm for video text extraction. *Pattern Recognition*, 36(6):1397–1406, 2003.

[28] W.G. Zhang, Q. Zhang, and C.S. Yang. Edge detection with multiscale products for sar image despeckling. *Electronics Letters*, 48(4):211–212, 16 2012.

[29] M. Zhao, S. Li, and J. Kwok. Text detection in images using sparse representation with discriminative dictionaries. *Image and Vision Computing*, 28(12):1590–1599, 2010.

[30] Y. Zhong, H. Zhang, and A.K. Jain. Automatic caption localization in compressed video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(4):385–392, Apr 2000.