academicJournals

Vol. 10(3), pp. 105-113, 15 February, 2015
DOI: 10.5897/SRE2014.6146
Article Number:14908D250363
ISSN 1992-2248
Copyright©2015
Author(s) retain the copyright of this article
http://www.academicjournals.org/SRE

Scientific Research and Essays

Full Length Research Paper

A hybrid multilevel text extraction algorithm in scene images

Tahani Khatib*, Huda Karajeh, Hiba Mohammad and Lama Rajab

Department of Computer Information Systems, King Abdullah II School for Information Technology, University of Jordan, 11942 Amman, Jordan.

Received 15 December, 2014; Accepted 22 January, 2015

The textual pieces in scene images might often provide vital semantic data for visual content understanding, indexing and analysis; as a result, text extraction had become a significant research area in image processing and computer vision. In this paper, we propose a new hybrid multilevel algorithm to extract text in various scene images. The algorithm converts the Red – Green –Blue (RGB) image into grayscale for color reduction. Next, it applies edge detection and mathematical morphological operations to extract edges in the image preprocessing phase. The resultant binary image passes through three subsequent levels in a multi layer behavior. Connected components labeling and text candidates' selection take place in each level through different criteria analysis. We used the structural features of connected components as basis criteria for selecting candidate texts, those features include: area, width, length and condense intensity mean of connected components. Afterwards, Horizontal projection profile analysis is used to further refine the candidate text areas and to eliminate non-text regions. The proposed algorithm is evaluated on a set of fifty images chosen from a well known text locating test dataset: KAIST. Extensive experiments show high robustness under different environments such as indoor, outdoor, shadow, night and light, and for different text properties such as various font size, style and complexities of backgrounds and textures. The algorithm effectively extracts textual contents from scenes images with high average of Precision, Recall, and F-Score which are 90.1, 99, and 94.3%, respectively.

Key words: Multilevel text extraction, hybrid text extraction, edge detection, connected components, text candidates, morphological operations, horizontal projection profile.

INTRODUCTION

The development of digital technologies accelerated the rapid growth in digital content. However, as digitalization is expanding in all categories and materials, it becomes important to extract any textual content from digital media to acquire semantic clues to help in visual content illustration and analysis. Digital images, as an essential form of digital media, may include pieces of text that comprise useful information for automatic explanation and structuring of images (Mancas-Thillou et al., 2007). Furthermore, information in embedded text can be used to fully understand images and for specific applications such as page segmentation in (Jain and Zhong, 1996;

*Corresponding author. E-mail: tahani.khatib@ju.edu.jo
Author(s) agree that this article remain permanently open access under the terms of the Creative Commons Attribution License 4.0 International License

Tang et al., 1996), address block location (Yu et al., 1997), license plate location (Cui et al., 1997; Kim and Chien, 2001), and content-based image/video indexing and retrieval (Shim et al., 1998; Zhang et al., 1994). Text appears in images either in the form of documents such as scanned CD/book covers or as video images. The embedded text in video frames can broadly be classified into two categories: overlay text and scene text. Overlay text refers to those characters generated by graphic titling machines and superimposed on video frames/images (Zhang and Chang, 2003), while scene texts are those captured by a recording device such as text in signs, nameplates, food containers, etc. Scene text is more difficult to detect (Gatos et al., 2005; Choksi et al., 2013; Sumathi et al., 2012) and therefore researches and studies in this field are so limited. In contrast to caption texts, scene texts can have any orientation and may be distorted by the perspective projection and may often be affected by variations in scene and camera parameters (Jung et al., 2004 in Kim and Chien, 2004; Mancas-Thillou et al., 2007); they also have several varieties of fonts, sizes, styles, reflections and shadows. As a result, Text extraction in scene images has become a challenging issue due to previous problems in addition to the complicated background in the image itself.

Many algorithms have been developed and improved in scene image text extraction. The majority of text extraction algorithms could be classified either as connected component based technique or as texture based technique (Fu et al., 2006). Connected component-based methods use geometric constraints and information to choose text candidates by creating bounding boxes around connected regions in images (Pan et al., 2009). The algorithm proposed in (Leon et al., 2010) makes use of similarity measure to choose text regions; it combines texture information and geometric information in order to extract text in scene image. In Rajab et al. (2014), we presented a text extraction technique that employs image enhancement, morphological operations and different transformations in order to label text candidates.

Texture-based methods treat the text as a unique object that has some distinguishable features from the background. The researchers in Wen and Chou (2004) used Discrete cosine transform (DCT) based high pass filter to remove constant background. The problem of texture-based methods is the large computational complexity in texture classification; which leads to a confusion when text-like regions appear. The variations on text fonts, sizes, colors and complex backgrounds (Shivakumara et al. 2014; Mao et al., 2013) affect the performance of these algorithms and hence text cannot be extracted by using a single method only. Niti (2014) and Xiaoqing and Jagath (2006) improved a hybrid and multi-scale method that use Support Vector Machine (SVM) transformation along with some pre-processing and post-processing steps in order to extract text in complex images (Chandrasekaran and Chandrasekaran (2011).

In Jung and Han (2004), two methods for text localization in complex images were proposed. The first method was an automatic texture-base method that can increase the recall rates for complex images; while the second one was a connected component-based filtering that took advantage of geometry and shape information to enhance the precision rates.

This paper proposes a hybrid multilevel text extraction algorithm that can locate and extract texts in complex scene images and can resolve problems that some previous systems had. The algorithm uses both connected component-based and texture-based techniques in text candidates' selection. It begins with image preprocessing which includes both color reduction and edge extraction. In color reduction step, the RGB image is converted into a grayscale image. Afterwards, the binary image resulted from edge extraction in the preprocessing phase is sent to three subsequent levels. All levels contain both connected components labeling and text candidates' selection; however every level has its own criteria used in text candidates' selection. Criteria used in candidate selection include analyzing area, width, height and intensity mean of connected components. Adaptive background elimination through logical operations is performed in inner phases in addition to analyzing the horizontal projection profile of the image in order to eliminate tiny non textual areas.

MATERIALS AND METHODS

Proposed algorithm

In this study, we improved a hybrid multi level algorithm for text extraction in scene images. The proposed algorithm uses both connected component-based and texture-based techniques and it includes preprocessing phase in addition to three sequential levels; every level contains inner phases where candidate text regions are labeled gradually within inner phases of each level. The proposed algorithm is discussed methodically in the following:

Image preprocessing

Image preprocessing phase is extremely significant in achieving better performance in text detection and extraction techniques. The scene image may contain some noise or effects such as shadow or light spots; therefore we need to remove those effects before labeling and detecting the text candidates in the image in order to get a better input image for next phases. Preprocessing phase includes both color reduction and edge extraction inner phases:

i) ***Image preprocessing: Color Reduction***
In this phase, the acquired colored image is converted from Red – Green –Blue (RGB) color model into grayscale and passes as the input image to the next phase.
ii) ***Image preprocessing: Edge Extraction***
In this phase, the canny edge detection is applied on the saturation grayscale image. The edge detection is applied to get the edge map of the image. Afterwards, morphological image dilation is used on the resultant binary image with a suitable structuring element.

After image preprocessing, the resultant binary image passes through three subsequent levels as follows:

Level 1

Phase 1.1: Labeling text candidates' regions

In this phase, all elements in the connected components set *S* are labeled and then tested by using some selection criteria in order to find the text candidates set S_T. Those criteria contain mathematical analysis of width, height and area of each element. Connected components with area (A_i) greater than a certain portion of the overall area (A_I) of the image will be eliminated; the analysis of this criteria helps in excluding large connected components that are far away from being textual regions. Width and height of connected components are also tested so all components with width less than twice and half of height (the threshold used in the algorithm) are eliminated. Equations (1) and (2) in the following show the criteria used in eliminating non text regions in this phase.

$$S = S_E \cup S_T \quad (1)$$

$$S_T = \left\{ i \in S : A_i < \frac{A_I}{t_1} \text{ and } W_i < H_i * t_2 \right\} \quad (2)$$

In previous equations, *S* stands for the set of all connected components, S_E is the set of the eliminated components and S_T is the set of candidate text regions. The total area of image *I* is denoted by A_I, while *i* stands for an element in *S* with area, width and height denoted by A_i, W_i and H_i, respectively. Variables t_1 and t_2 are the thresholds used in our algorithm which were obtained from many experiments on large set of images and have values 18 and 2.5, respectively. Figure 1(a) shows a sample scene image used in algorithm testing, Figure 1(b) shows the result image after connected components labeling, while the result of text candidate selection is shown in Figure 1(c).

Phase 1.2: Text Extraction – level 1

In this phase, we apply multi-step operations on the text candidates in order to extract the text from the image. The inner steps of this phase are discussed in the following.

Step 1.2.1: Morphological operations: A set of morphological operations with filling procedures are applied on the image to facilitate edge enhancement; morphological operations include close and open operations followed by holes filling.

Step 1.2.2: Eliminate large non-text areas from the background using adaptive logical operator: In this step, we apply an adaptive (AND) operator between the binary image in Phase 1.1 and the enhanced edged image from Phase 1.2.1; this step gives excellent results in eliminating large non-text regions from the image background. Applying logical (AND) between both enhanced edge image and the adjusted monochrome version from the original helped in studying the foreground and the background of the image. However, if the intensity mean of the resultant image is greater than a certain threshold, an image negation operation is performed to keep the important foreground data; otherwise, the original image is converted to a monochrome version using different threshold. Figure 1(d) shows the result image after applying this phase, while the procedure is shown in the following pseudo code.

Pseudo code: (Adaptive AND Procedure)
Input (G, I, J)
Where: G is the grayscale image, I is the resultant image from Step 1.2.1 and J is the resultant image from Step 1.1

IF the intensity mean of J > 0.5 then

1. Find image negation of J, store the result in Jn
2. Calculate R = AND (I, Jn) ELSE
3. Find binary image of G with a larger threshold (0.75), store the result in Jnb
4. Calculate R = AND (J, Jnb)

End IF
Output (R)

Level 2

Phase 2.1: Labeling the text candidate regions

Text candidates are labeled by studying the condense intensity mean of white pixels (intensity = 1) for each connected component; if the mean is greater than a certain threshold, the region will be eliminated, otherwise it will be labeled as text region as shown in Equation (3).

$$S_T = \{ Mean(i) < t_3 \} \quad (3)$$

In the previous equation, S_T denotes the set of candidate texts, *i* is an element in S (the set of all connected components) and threshold *t3* = .80.

Phase 2.2: Text extraction – level 2

Step 2.2.1: Image post processing: After labeling the text candidates in Phase 2.1, border thinning operations are applied for text candidate regions to remove interior pixels. Filling operation is applied next to reduce gaps between pixels in connected components.

Step 2.2.3: Horizontal projection profile: A horizontal projection profile is defined as the sums of the candidate pixels over image rows (Ye et al., 2005). In this step, the small non textual regions are eliminated by using the horizontal projection profile of the result image after post processing. Pixel rows with intensity sum less than an acceptable threshold will be discarded. The threshold is relative to the total intensity mean of the horizontal projection profile of the image. Figure 2 illustrates eliminating small non-text regions by using the image horizontal projection profile. Figure 2(a) shows the image with small non-text regions, while Figure 2(b) shows the horizontal projection profile of the image with a red circle indicating the candidate image regions to be eliminated. The elimination is applied on all small curves indicating tiny intensity sum of row pixels. Figure 2(c) shows the resultant image after non-text elimination using projection profile analysis.

Level 3

Phase 3.1: Labeling the text candidate Regions

The principal objective of this final level is to ensure that all non-text components are eliminated in the image. In this phase, all criteria features studied earlier are examined for each text candidate again for the last time, those features are:

1. Area of the text candidate,
2. Height and width of the text candidate,
3. Intensity mean of the text candidates.

The text candidate area should be greater than 1/10 of the mean area of connected components with acceptable number of pixels, additionally it should not exceed 4 times of the mean area. The width and height of the text candidate should relatively conform to

Figure 1. The proposed algorithm results from Level 1 (a) Original image; (b) Result image after connected component labeling; (c) Result image after text candidates selection; (d) Result image after text extraction - level 1.

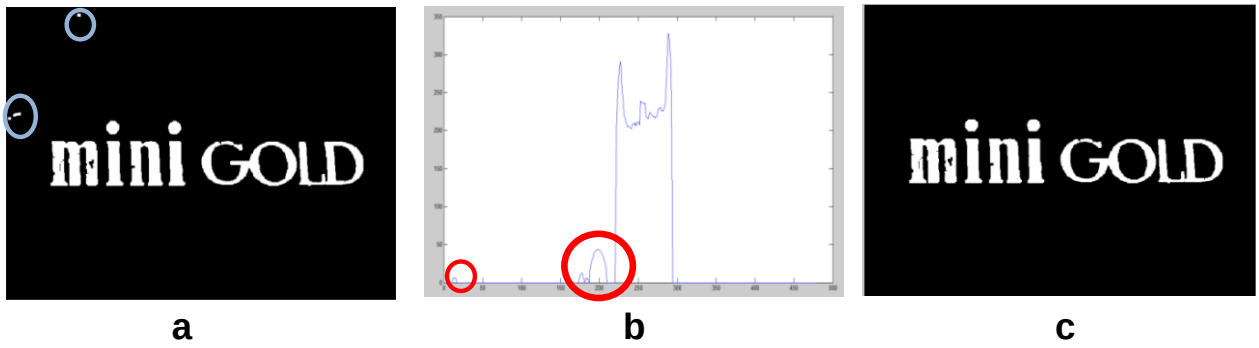

Figure 2. Eliminating small non-text regions by using the image horizontal projection profile (a) Small non-text regions are circled; (b) The horizontal projection profile; (c) Result after non-text elimination.

the English letters rules so that the width should not exceed the height*6 and vice versa. For the intensity mean of text candidate, we eliminated all components with intensity mean greater than 90%, so if the majority of the component texture is white, then the area cannot be a textual region; conversely, it may be a solid region or noise.

Phase 3.2: Final text extraction

After Level 3, the text candidate regions are extracted, and the final textual information in the image is detected. Figure 3 shows a sample final result of the extracted text at the end of this level. All levels, inner phases and steps of the proposed algorithm discussed previously are illustrated in the block diagram shown in Figure 4.

Testing dataset

The selected images from KAIST dataset are used to test the performance of the proposed algorithm (Jin and Seonghun, 2011). This dataset is developed by the Korean Advanced Institute of Science and Technology (KAIST) where the dataset name came from. KAIST dataset consists of scene text images with different properties such as (color, font size, orientation, and alignment) and were captured in five different environments: light, night, shadow, indoor, and outdoor. This dataset is grouped based into the languages: English, Korean, and mixed of English and Korean. Each of these groups is classified according the captured environment condition. All images in the dataset have been resized into 640 × 480. To test the performance of the proposed algorithm, a set of English language scene images have been selected from KAIST set. Testing was based on selecting some scene images with different properties such as (color, font size, orientation, and alignment) and that were captured in different environment.

RESULTS AND DISCUSSION

In this study, proposed text detection algorithm quantitatively and qualitatively were evaluated. The analysis of results is based on various experiments and measurements and is discussed in the subsequent subsection.

Procedure

The performance of the proposed algorithm is evaluated under 50 scene images selected from KAIST dataset with different properties and captured in different environments. The availability of ground truth images in the KAIST dataset provides a better opportunity to compare the proposed algorithm resultant image with the ground truth image quantitatively and qualitatively. An example on a scene image and its ground truth from KAIST dataset is shown in Figure 5.

Selecting optimum values for thresholds used in our algorithm was not an easy task; extensive testing on

Figure 3. The final extracted text.

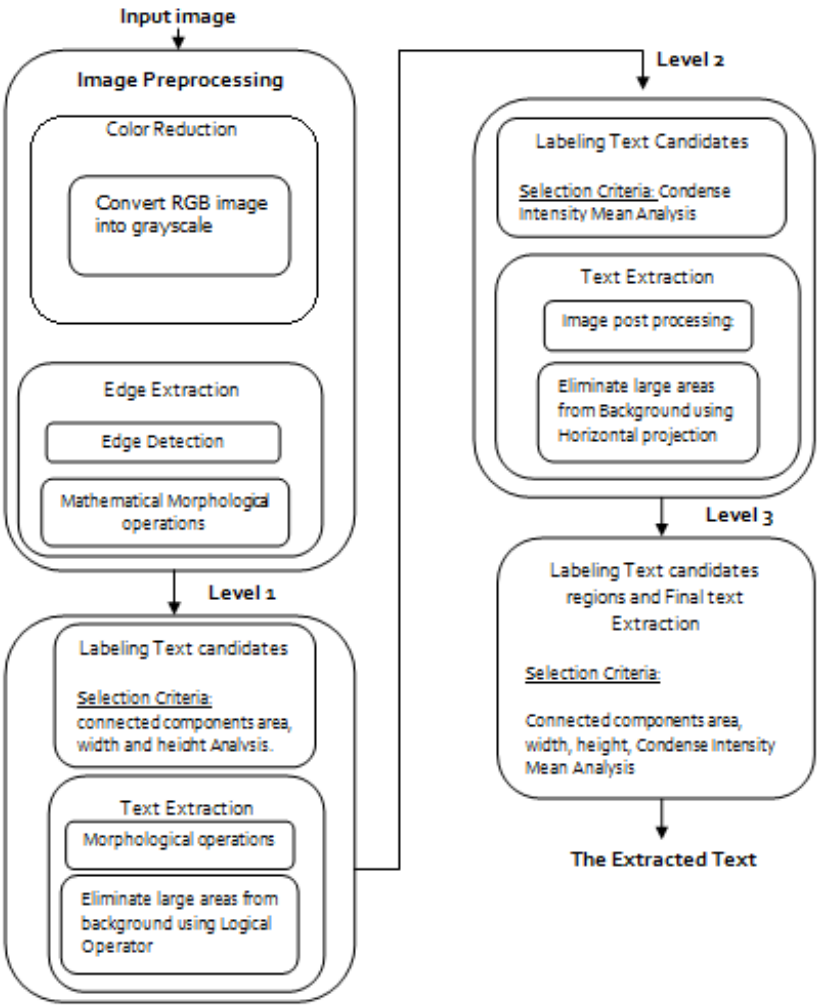

Figure 4. The block diagram for the multilevel text detection algorithm.

Figure 5. A KAIST image and its corresponding ground truth a. The original image; b. Ground truth image.

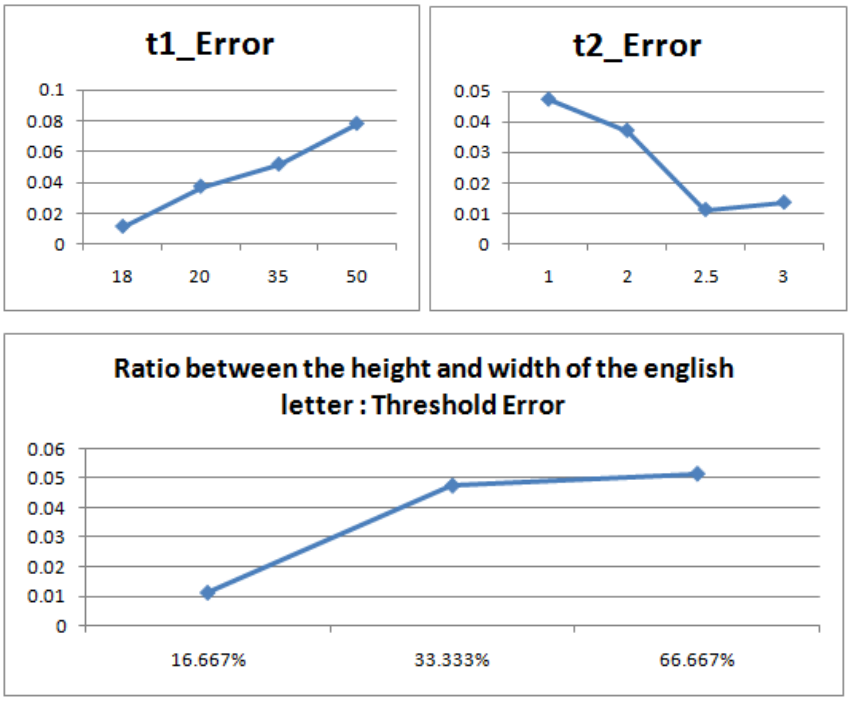

Figure 6. Choosing some thresholds optimum values. The upper charts compare between t1, t2 thresholds probabilities; while the bottom chart compares between threshold probabilities for the ratio between height and width in English letters.

many images had been performed to choose the best value for each one. Figure 6 shows comparisons between the algorithm testing results when applied on one KAIST image with different threshold probabilities. Figure charts compare between error percentages for threshold probabilities. The minimum error values: 18, 2.5 were selected for thresholds, demonstrated previously in this study: t1, t2 respectively. Also, the third chart shows the best threshold for the ratio between height and width for English letters which is 16.667 or 1:6%.

Quantitative metric

The performance of the proposed text extraction algorithm is evaluated quantitatively by calculating three measurement metrics: Precision, Recall, and F-Score. The calculation of these metrics is based on computing the number of corresponding match text between the algorithm's detected text area and the ground truth image. This yields calculating three measurements: true positive *tp*, false positive *fp*, and false negative *fn*. True positive (*tp*) represents the number of pixels that are truly classified as text in the algorithm's detection result, and false positive (*fp*) represents the number of pixels that are falsely classified as text in the algorithm's detection result while it is a background in the ground truth. False negative (*fn*) represents the number of pixels that are falsely classified as background in the algorithm's detection result while it is a text in the ground truth. Based on these measurements, Precision, Recall, and F-Score are calculated as in the following equations:

$$Precision = \frac{tp}{tp + fp} \quad (4)$$

$$Recall = \frac{tp}{tp + fn} \quad (5)$$

$$F - Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (6)$$

Precision value measures the percentage of the correctly detected text from the whole detected text area while the Recall value measures the probability of the text detection algorithm of correctly detecting the text area. F-Score value represents a harmonic mean of the precision and recall to give a single value to measure the effectiveness of the detection results.

Analysis

The proposed text extraction algorithm was improved to resolve the problems encountered in our system proposed

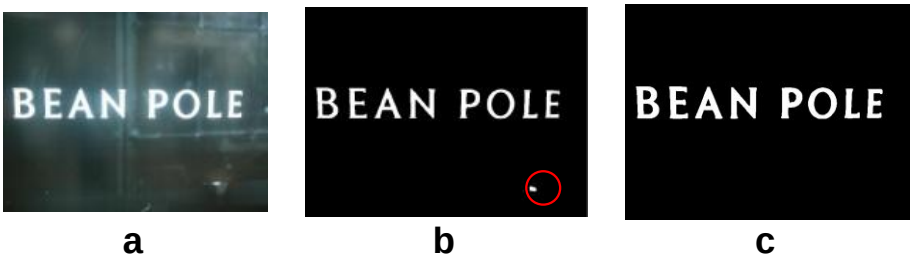

Figure 7. False text detection in the old algorithm and the improvement in the new one a. Original image; b. Lama Rajab et.al (2014) Algorithm; c. Proposed algorithm.

in (Rajab et al., 2014). The previous system presented a connected component-based text extraction technique that employs image enhancement, morphological operations and different transformations such as Hough transform in order to label and extract text candidates. However in this paper, a hybrid multi-level text extraction that uses both connected component-based and texture-based techniques in text candidates' selection was present. The algorithm applies color reduction and edge enhancement on input image followed by three subsequent levels. Each level applies multiple inner phases like connected components labeling and text candidates' selection based on criteria analysis of connected components' area, width, height and intensity mean in addition to image horizontal projection profile analysis.

The old system proved its robustness in text extraction on many images but unfortunately it failed in extracting the text from shadow images and it was detecting the light spot falsely as a text. Figure 7 shows a sample image with a light spot detected as text in the old technique. However, the current Algorithm utilized completely different techniques and presented a novel methodology to improve the performance of text extraction in such images.

The improvement of the current algorithm performance was experimented by testing both algorithms on a set of common images from about 15% of the overall test set. The precision, Recall and F_score values of the proposed algorithm for this set of images are 0.868, 0.991, 0.924, respectively, while they are 0.853, 0.955 and 0.889, respectively for algorithm (Rajab et al., 2014). Comparisons between these metrics for both algorithms are shown in the chart (Figure 8).

Relatively to the comparative analysis above, the effectiveness of the proposed text extraction algorithm is tested also individually on fifty selected scene images from KAIST dataset that have different properties and were captured in different environments. The results under five different environments: indoor, outdoor, light, night, and shadow are shown in Figure 9; two images were selected from each environment.

Obviously, the detection results from the proposed technique are very accurate and robust in detecting text from scene images that have different properties such as font size and type, color, orientation, and alignment. Moreover, the proposed algorithm detects the text accurately from images that have been affected to strong light or those which have dark or bright illumination spots (Figure 9c and d). It also detects the large characters accurately as well as the small ones in both indoor and outdoor environments (Figure 9a and i), as well as in images which have shadow areas as in Figure 9g and h. Moreover, it proved to be robust and effective in detecting images with curved texts (Figure 9b).

As stated previously, the performance of the proposed technique is evaluated quantitatively using three metrics: Precision, Recall, and F-Score that obtained from comparing the output image from the proposed algorithm with the ground truth. These three measurements are calculated for 50 images that were selected from KAIST dataset. The average of Precision, Recall, and F-Score on this set consisting of fifty KAIST images is 90.1, 99, and 94.3%, respectively.

Noticeably, the average of the Recall metric is very high (99%) due to the high probability of our text detection algorithm of correctly detecting the text area in the scene image and this is obvious in the Figure. As a result, our proposed algorithm is robust and consistent under the different environments and under variant properties.

Unfortunately in some cases, the algorithm detects some small areas falsely if they have similar properties to texts. Therefore, the algorithm will label these background areas as candidate text regions which will be detected as textual contents in the further steps. Thus, the existence of some small areas which are similar to text properties will decrease the precision since these areas were extracted falsely to be texts while they are in fact background areas as we can see in Figure 9b, e, and h, and that affects the value of Precision metric.

Conclusion

Text extraction in scene images is a significant and promising research area in computer vision. In this paper, we propose a new and improved multilevel and hybrid algorithm that can detect and extract the textual content

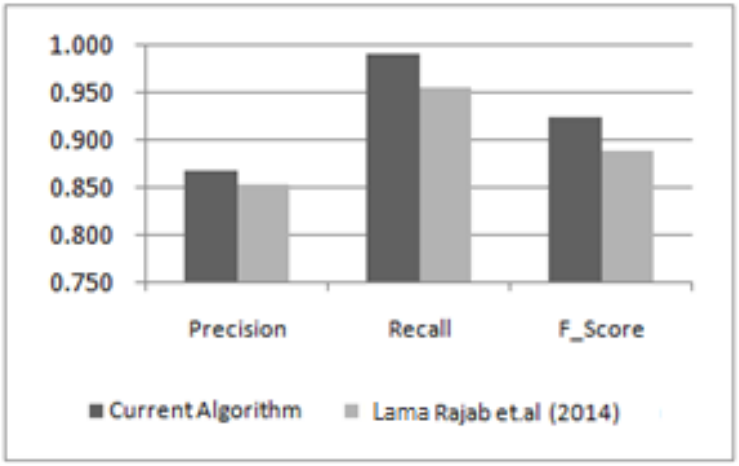

Figure 8. Comparisons of Precision, Recall and F_Score between both: our new and old techniques.

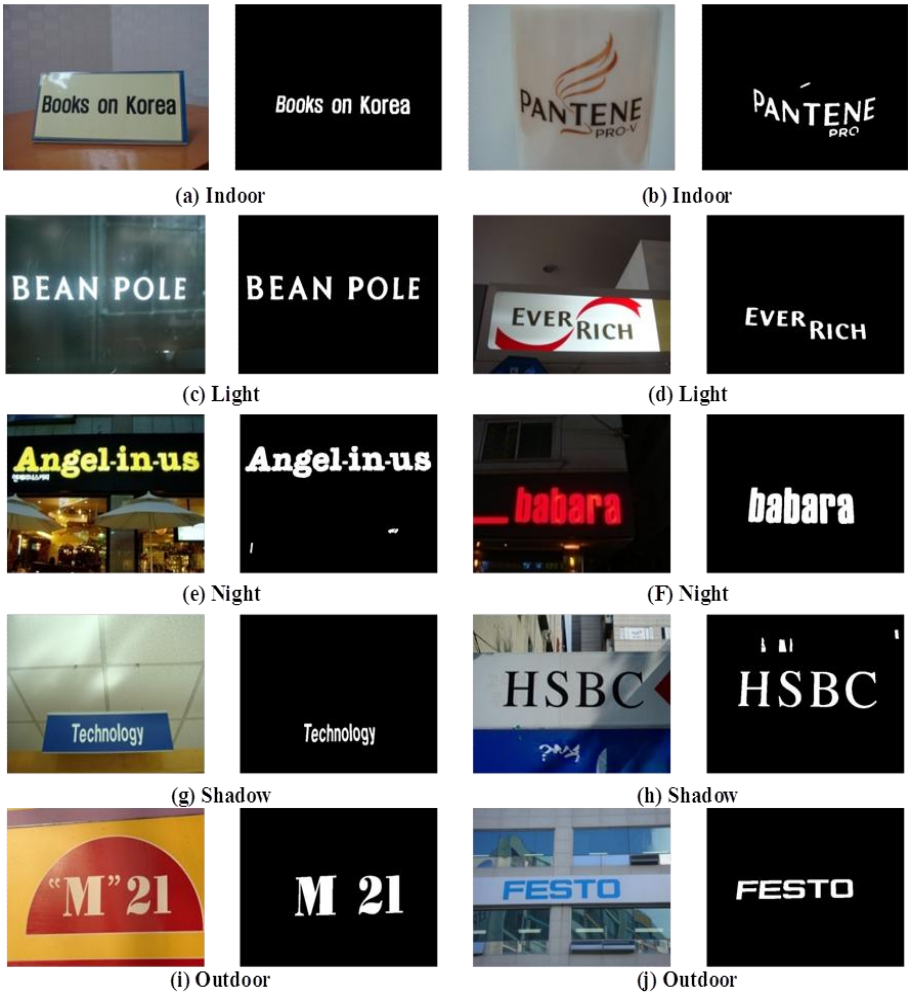

Figure 9. Set of images from KAIST dataset with detected text area using the proposed technique. (a) and (b) were captured on a indoor environment. (c) and (d) were captured on light. (e) and (f) were captured on night environment. (g) and (h) are two images that were captured at shadow. (i) and (j) are two images that were captured at outdoor.

in various scene images. The algorithm uses both connected component-based and texture-based techniques in text candidates' selection. The improvement is represented by using a hybrid multilevel detection method with subsequent multi phases in order to extract the text progressively. We have anticipated through the proposed algorithm to resolve problems that our some previous systems had in text extraction field.

In the preprocessing step, the algorithm used the grayscale version from the RGB image and then applied edge extraction on the resultant image. Various techniques in three subsequent levels are applied after preprocessing such as connected component labeling, text candidate selection with different criteria testing, morphological operations, and projection profile based technique for non-text regions elimination. As a result, candidate text areas will be labeled, detected and extracted in a multi layer behavior in inner phases.

The effectiveness of the proposed technique is tested on 50 images from KAIST dataset that were captured in different environments (shadow, light, outdoor, indoor, and night). Precision, Recall, and F-Score metrics are used to test the accuracy of the text detection rate for the proposed technique quantitatively. The results show that the proposed algorithm detects the text with high average of Precision, Recall, and F-Score to be 90.1, 99, and 94.3%, respectively. The algorithm also proved to be robust and consistent in terms of detecting the textual content from scene images that have various properties and which were captured in different environments.

Conflict of Interest

The authors have not declared any conflict of interest.

REFERENCES

Jain AK, Zhong Y (1996). Page segmentation using texture analysis. Pattern Recognition 29(5):743-770. DOI: 10.1016/0031-3203(95)00131-X

Chandrasekaran R, Chandrasekaran RM (2011). Morphology based Text Extraction in Images. Int. J. Comput. Sci. Technol. 2:4.

Choksi A, Desai N, Chauhan A, Revdiwala V, Patel K (2013). Text Extraction from Natural Scene Images using Prewitt Edge Detection Method. Int. J. Adv. Res. Comput. Sci. Software Eng. 3:12 DOI 10.1007/11595755_59.

Cui Y, Huang Q (1997). Character extraction of license plates from video. In Computer Vision and Pattern Recognition, Proceedings IEEE Computer Society Conference. pp. 502-507. DOI :10.1109/CVPR.1997.609372

Kim DS, Chien SI (2001). Automatic car license plate extraction using modified generalized symmetry transform and image warping. In Industrial Electronics, 2001. Proceedings. ISIE 2001. IEEE International Symposium 3:2022-2027. DOI:10.1109/ISIE.2001.932025

Fu H, Liu X, Jia Y (2006). Maximum-Minimum similarity training for text extraction, King et al. (Eds.): ICONIP 2006, Part III, LNCS 4234:268-277. DOI: 10.1007/11893295_31.

Gatos B, Pratikakis I, Perantonis S (2005). Towards text recognition in natural scene images. Proc. Int. Conf. Automation Technol. pp. 354-359, doi:10.1.1.76.9618

Jin HK, Seonghun L (2011) http://www.iapr-tc11.org/mediawiki/index.php/KAIST_Scene_Text_Database accessed on (2014-07)

Jung K, Han J (2004). Hybrid approach to efficient text extraction in complex color images. Pattern Recognit. Lett. 25(6):679-699. DOI:10.1016/j.patrec.2004.01.017.

Jung, K, Kim K, Jain KA (2004). Text information extraction in images and video: A survey. Pattern recognit. 37(5):977-997. http://dx.doi.org/10.1016/j.patcog.2003.10.012.

Lama R, Mohammad H, Karajeh H, Al Khatib T (2014). An improved text extraction technique based on linear transformation. Life Sci. J. 11(7):83-88.

Leon M, Vilaplana V, Gasull A, Marques F (2010). Region-based caption text extraction. In Image Analysis for Multimedia Interactive Services (WIAMIS), 2010 11th International Workshop on IEEE. ISBN: 978-1-4244-7848-4. pp. 1-4.

Mancas-Thillou C, Gosselin B (2006). Natural scene text understanding. na. Vision Systems: Segmentation and Pattern Recognition, Goro Obinata and Ashish Dutta (Ed.), ISBN: 978-3-902613-05-9, In Tech, DOI: 10.5772/4966. Available from: http://www.intechopen.com/books/vision_systems_segmentation_and_pattern_recognition/natural_scene_text_understanding

Mao J, Li H, Zhou W, Yan S, Tian Q (2013). Scale based region growing for scene text detection. Proceedings of the 21st ACM international conference on Multimedia ACM Multimedia, pp.1007-1016 , DOI:10.1145/2502081.2502108.

Niti S, Naresh K, (2014).Text extraction in images using dwt, gradient method and svm classifier. Int. J. Emerging Technol. Adv. Eng. 4:6 ISSN 2250-2459, ISO 9001:2008 Certified Journal.

Pan Y, Hou X, Liu C (2009). Text Localization in Natural Scene Images based on conditional Random Field .10th International Conference on Document Analysis and Recognition. DOI 10.1109/ICDAR.2009.97

Shim J, Dorai C, Bolle R (1998). Automatic text extraction from video for content-based annotation and retrieval. Proc. Int. Conf. Pattern Recognit. (1):618-620. DOI: 10.1109/ICPR.1998.711219

Shivakumara P, Kumar NV, Guru DS, Tan CL (2014). Separation of Graphics (Superimposed) and Scene Text in Video Frames. In Document Analysis Systems (DAS), 2014 11th IAPR International Workshop on IEEE. pp. 344-348. DOI :10.1109/DAS.2014.20.

Sumathi C, Santhanam T, Gayathri G (2012). A survey on various approaches of text extraction in images. Int. J. Comput. Sci. Eng. Survey. 3:4. DOI: 10.5121/ijcses.2012.3403.

Tang Y, Lee S, Suen C (1996). Automatic document processing: A Survey. Pattern Recognit. 29(12):1931-1952. DOI: 10.1016/S0031-3203(96)00044-1.

Xiaoqing L, Jagath S (2006). Multiscale edge-based text extraction from complex images. Conference: Proceedings of the 2006 IEEE International Conference on Multimedia and Expo, Toronto, Ontario, Canada. DOI: 10.1109/ICME.2006.262882

Ye Q, Huang Q, Gao W, Zhao D (2005). Fast and robust text detection in images and video frames. Image Vision Comput. 23(6):565-576. DOI: 10.1145/1101149.1101250

Yu B, Jain AK, Mohiuddin M (1997). Address block location on complex mail pieces. In Document Analysis and Recognition, Proceedings of the Fourth International Conference on IEEE. 2:897-901. DOI: 10.1109/ICDAR.1997.620641.

Zhang D, Chang S (2003). Accurate overlay text extraction for digital video analysis, in international conference on information technology: Research and Education (ITRE) DOI:10.1.1.134.7724

Zhang H, Gong Y, Smoliar SW, Tan SY (1994). Automatic parsing of news video. In Multimedia Computing and Systems. Proc. Int. Conf. IEEE. pp. 5-54. DOI :10.1109/MMCS.1994.292432.