International Journal of Computing Sciences Research (ISSN print: 2546-0552; ISSN online: 2546-115X)
Vol. 7, pp. 1699-1722
doi: 10.25147/ijcsr.2017.001.1.129
https://stepacademic.net

Long Paper

Towards Improved Performance of Text Detection Algorithms: Exploring the Impact of Automatic Image Classification and Blind Deconvolution

Clarisa V. Albarillo
College of Computer Science, Don Mariano Marcos Memorial State University, Philippines
calbarillo@dmmmsu.edu.ph
(corresponding author)

Proceso L. Fernandez, Jr.
Department of Information Systems and Computer Science, Ateneo de Manila University, Philippines
pfernandez@ateneo.edu

Date received: October 5, 2022
Date received in revised form: November 16, 2022; November 24, 2022
Date accepted: December 1, 2022

Recommended citation:

Albarillo, C. V. & Fernandez, P. L. (2023). Towards improved performance of text detection algorithms: Exploring the impact of automatic image classification and blind deconvolution. *International Journal of Computing Sciences Research, 7,* 1699-1722. https://doi.org/10.25147/ijcsr.2017.001.1.129

Abstract

Purpose – This study aims to investigate the impact of the proposed BD-CRAFT, a variant of the CRAFT algorithm applying preprocessing steps as blurry or non-blurry image classification using Laplacian and a deblurring technique known as blind deconvolution, in improving the performance of the top three state-of-the-art scene text detection algorithms – SenseTime, TextFuseNet, and TencentAILab.

Methodology – The researchers utilized the ICDAR 2013 Focused Scene Text Competition Challenge 2 dataset and the Intersection over Union (IoU) to determine the performance of the proposed BD-CRAFT. The IoU h-mean of the top three algorithms was compared against those of the modified versions.

Results – Each algorithm variant significantly improves the overall h-mean and some of the precision and recall values. TextFuseNet + BD-CRAFT yields 93.55% h-mean, while the precision shows an impressive improvement of over 4% to increase the precision to 95.71%.

Meanwhile, TencentAILab + BD-CRAFT achieved an h-mean result of 94.77% with precision and recall improvement. Furthermore, SenseTime + BD-CRAFT ranked first with a very impressive 95.22% h-mean and showed a significant improvement of over 4%, which made it the top-ranked algorithm.

Conclusion – Evidence shows that when BD-CRAFT is combined with other algorithms, their performances are improved; hence BD-CRAFT has a significant impact on the text detection performance of these algorithms.

Recommendation – As possibilities for further studies, it would be interesting to investigate the other state-of-the-art algorithms for scene text detection that would also benefit from BD-CRAFT. Exploring other preprocessing techniques that can be incorporated into text detection algorithms in general, may be suitable.

Research Implication – Though the performances of current state-of-the-art algorithms are already commendable, the use of image classification and blind deconvolution as preprocessing techniques helps the top-performing text detection algorithms perform better in natural scene images hence the proposed method can be utilized in improving scene text detection.

Keywords – blind deconvolution, image classification, image deblurring, image processing, text detection

INTRODUCTION

Text detection in natural scenes has been a significant and active research subject in computer vision and document analysis because of its wide range of applications. These applications include image search (Zu et al., 2016; Gordo et al., 2016), target geolocation (Namazi et al., 2022; Renshaw et al., 2022), human-computer interaction (Chakraborty et al., 2018; Brodić & Amelio 2016), robot navigation (Nguyen et al., 2016; Panchpor et al., 2018), and industrial automation (Sharma et al., 2022; Gao et al., 2021), can greatly benefit from the detailed information included in the text. Despite advances in the field, several challenges still exist, such as noise, blur, distortion, occlusion, and variance.

In 2011, the Robust Reading Competition emerged in conjunction with the International Conference on Document Analysis and Recognition (ICDAR). The competition is structured around several challenging computer vision tasks that cover a broad range of actual conditions. The Focused Scene Text is one such challenge, and it focuses on reading texts in actual scenarios, with "focused text"—images of text that are primarily focused around the text content of interest—as the scenario under examination. Furthermore, ICDAR 2011, ICDAR 2013, and ICDAR 2015 are the three editions of the Focused Scene Text

Challenge. For the tasks of text localization, text segmentation, and word recognition, the ICDAR 2013 is the definitive one (Robust Reading Competition, 2020).

One of the most commonly used evaluation metrics in text detection is the Intersection-over-Union (IoU). The algorithms having the best IoU h-mean in the above-mentioned competition include: SenseTime (the top-ranked algorithm) whose h-mean is 93.62%, TextFuseNet with 93.11% h-mean, TencentAIlLab with 93.05%, VARCO with 91.71%, HIT with 91.48% h-mean and CRAFT with 91.41%. In SenseTime, a single end-to-end trainable Fast Oriented Text Spotting (FOTS) network that is designed for simultaneous detection and recognition is used. To share convolutional features across detection and identification, it specifically introduced RoIRotate (Kim & Park, 2020).

Most scene text detectors train their networks to locate bounding boxes at the word level. In complex situations, such as texts with arbitrary font and size, and varied text scales and shapes, such as curved, distorted, or exceedingly long texts, this level may be challenging. In these situations, detectors frequently return bounding boxes at the character level rather than the entire word. One algorithm that can handle these cases is the Character Region Awareness for Text detection (CRAFT) (Baek et al., 2019). It is a text detector that localizes the specific character regions and then connects the detected characters to a text instance. Both the affinity score, which merges all of the characters into one instance, and the character region score, which is used to localize specific characters within an image, are generated using a convolutional neural network.

Currently, the CRAFT algorithm ranks sixth in the competition, with an IoU h-mean of 91.42%. Though CRAFT's performance is already commendable, there is still much room for improvement because it assumes that the images of ICDAR 2013 are free from any blur or image distortion. Likewise, current text detection algorithms also treat the input images to be clear and do not employ image preprocessing before running the text detection algorithm hence the idea of imploring image classification and image deblurring through Blind Deconvolution as image preprocessing was conceptualized.

A recently concluded study (Albarillo & Fernandez, 2022) showed an improved text detection performance of CRAFT by adding some preprocessing steps that include automatically detecting blurry images and then attempting to reduce the blur of the identified blurry images before running the CRAFT algorithm. The resulting technique is referred to as BD-CRAFT, a CRAFT variant. BD-CRAFT was shown to be not only significantly better than CRAFT but also outperformed the current best state-of-the-art algorithms for scene text detection. In this study, combining BD-CRAFT with some state-of-the-art algorithms to further improve their text detection performances is explored.

LITERATURE REVIEW

Text Detection

Text detection methods in scene images can be classified into two groups: sliding window-based approach and component-connected approach.

The sliding window checks the image densely by employing a multi-scale sub-window with a pre-designed classifier. Several sliding window-based methods have been developed. One of its strengths is that it calculates a global feature from the traversed windows making the said feature to be invariant to numerous low-level distortions or transformations. Another strong point of this method includes robustness to noise and blur since it exploits features amassed over the whole region of interest.

The Sliding Window works by either localizing individual characters or by localizing the complete words and traversing the entire image through the use of a classification window. The said strategy usually deals with exhaustive search (Nguyen et al., 2017; Neumann & Matas, 2015) as evidenced by the number of windows which grow to $O(n^2)$ for an image of n pixels, making it less practical.

Another drawback is that the number of rectangles that ought to be assessed grows rapidly when text with diverse scale, perspective, rotation, and other distortions should be found – an impact that does not occur in common object detection tasks where the variance of sliding window parameters is lower (Nguyen et al., 2017). Likewise, it has high computational cost since it requires multi-scale windows to handle texts of different font sizes resulting in a large number of scanning windows. Finally, it is difficult to design a discriminative feature and train a powerful classifier for whether text or non-text.

Existing approaches on sliding windows are usually developed on character level detection which is unreliable and is not robust. Numerous bottom-up steps such as identifying and grouping character candidates into text lines are also required which results in increased complexity.

A prevalent approach that has achieved promising performance is the connected-components-based algorithm. This technique finds individual characters using local properties of an image such as color, intensity, stroke width, and the like. It detects textual information at the pixel level by employing a fast low-level detector and groups the detected pixels into text-candidate text components.

Among the connected component-based approaches, Maximally Stable Extremal Region (MSER) (Huang et al., 2013) and Stroke Width Transform (SWT) (Epshtein, Ofek & Wexler, 2010) are the popular low-level methods for detecting text component candidates. The MSERs algorithm can distinguish challenging text patterns, causing the said algorithm to have a good recall in character detection, thus leading to several high-performance systems, Stroke Feature Transform (SFT) (Huang et al., 2013), Edge Box (Zitnick & Dollár, 2014) and Characterness (Li et al., 2014) are some of the recently developed low-level text detectors.

The complexity of these detectors is not dependent on the textual content parameters as characters of all scales and orientations can be detected in one pass, thus having a great impact on its speed. However, these detectors have main difficulties in filtering non-text components and combining components into text lines because these detectors generate a huge amount of non-text components due to their low-level nature, thus requiring some bottom-up post-processing steps to yield good performance.

Recently, automatic text detection has emerged as an active research area in computer vision and document analysis, providing a way to access and make use of textual information in images. The discipline of computer vision has witnessed a significant amount of research in the extraction of accurate texts from scene images (Tian et al., 2016; Shi et al., 2017; Shi et al., 2016, Zhou et al., 2017; He et al., 2016). This is because it has numerous real-world uses in image retrieval, scene understanding, robot navigation, and document analysis. Many multimedia applications, including visual classification (Karaoglu et al., 2016; Bai et al., 2017) and video analysis (Yin et al., 2016), require text detection as a prerequisite. Due to the inherent issues and difficulties, traditional text identification algorithms frequently involve numerous processing phases, such as character/word candidate generation (Busta et al., 2015; Jaderberg et al. 2016), and candidate filtering and grouping (Baek et al., 2019).

Image Classification

An essential foundation for image depth processing and the use of computer vision technologies in related domains involves image classification. To classify images traditionally, a process called image preprocessing, feature extraction, classifier development, and learning training are involved (Wang, et al, 2019). Traditional image classification techniques primarily rely on the extracted fundamental image features to achieve image classification, which can serve as a foundation for the future computer-based acquisition of the semantic data of images. Traditional image classification often uses support vector machines and logistic regression to perform image classification and uses image color, texture, and other information to calculate image features (Kechagias-Stamatis & Aouf, 2017). The results of image classification are influenced by knowledge and expertise in related domains, in addition to a significant degree of dependence on the extracted features.

In addition to being challenging to apply manually acquired features to image classification, feature data analysis takes a lot of time. Furthermore, traditional machine learning cannot be used to analyze huge datasets, and it is difficult to achieve the optimization of feature design, feature selection, and model training, which reduces the model's ability to accurately classify data. As a result, traditional machine learning-based image categorization techniques are impacted in numerous application sectors (Ding, et al, 2019). According to research, low-level essential features can serve as the foundation for image classification since texture, shape, and color features can be used for image classification and recognition. Traditional image classification techniques often involve the extraction of a single feature or a set of features, while support vector machines use the extracted features as input values.

Artificial neural network classifiers have helped to advance image classification in recent years. Moreover, deep learning has employed the technique of layer-by-layer feature extraction to acquire the high-level attributes of the image and achieves the training of massive datasets using multilayer network models.

Deblurring Techniques

Many deconvolution techniques have emerged and have been implemented to deblur images. These include no neighbor approach, linear deconvolution methods, nonlinear methods, statistical methods, blind deconvolution, and image sharpening.

Among the most prevalent deblurring technique is blind deconvolution. Blind deconvolution is a well-established image restoration technique, where the point nature of the objects photographed exposes the PSF thus making it more feasible. In most blind deconvolution methods, motion blur kernels and the latent image are alternatingly optimized in an iterative process. From the estimated latent image and the given blurred image, blur kernels are obtained. These kernels are then used to produce a new estimated latent image by applying non-blind deconvolution to the given blurred image. This new estimated latent image is then used in the next iteration of kernel estimation. As the estimated PSF better describes the system optics and the sample image is better isolated from the acquired image, the error decreases as the number of cycles increases.

Applying constraints, such as the result cannot be negative, or that the PSF must be symmetrical, aids the selection of useful PSF models. A type of constrained iterative approach, blind deconvolution, allows N iterations to be directed at improving the PSF, followed by N iterations directed at improving the image before repeating: PSF, image, PSF, image.

Blind deconvolution, as one of the most common deblurring methods, has been considered one of the leading research topics. As proof, several algorithms have been proposed. Single-image deblurring has recently attracted great attention in computer vision and numerous methods have been developed (Koh et al, 2021; Lai et al., 2016; Park et al., 2020). With this, a proposed novel low-rank prior for blind image deblurring. It is observed that a simple low-rank model can significantly improve the quality of an input image and reduce the blur even without using any kernel information while preserving important edge information. Likewise, a gradient map of blurry input can also be enhanced through a low-rank model. With this, an enhanced prior for image processing was introduced which combines the low-rank prior of similar patches from the blurry image and its gradient map. To further enhance the effectiveness of low-rank prior, the weighted nuclear norm minimization method was employed wherein the dominant edges were retained while the fine texture and slight edges were eliminated thus allowing for better kernel estimation.

METHODOLOGY

The Dataset

The Focused Scene Text Competition Challenge 2 dataset from the 2013 International Conference on Document Analysis and Recognition (ICDAR) was used in the study (Karatzas et al., 2013). Because this study is primarily concerned with scene images, recent computer vision datasets like 2017 COCO-Text (Veit et al., 2016), deTEXT (Yang et al., 2017), DOST (Iwamura et al., 2016), FSNS (Smith et al., 2016, MLT (Nayef et al., 2017), and IEHHR (Fornés et al., 2017) were not employed.

The ICDAR 2013 Challenge 2 dataset is composed of 299 training images and 233 test images. Training images were used to train the text detection algorithms while the test images were used to determine the performance of the said detection algorithms. Furthermore, the dataset is composed of images explicitly focused on the text content of interest. This represents the use case wherein a person focuses a camera on a scene text for text reading and translation applications. As such, the focus text is horizontal in most cases. Different cameras were used to capture the images in varying environments, capturing images in .jpg format but with various sizes (such as 16KB up to 5.82MB), dimensions (350x200 up to 3888x2592), orientations (portrait and landscape) and light conditions. With this, the ICDAR 2013 was used in the study as it is the definitive one for the task of text detection.

Implementation

In the implementation of the CRAFT algorithm, open-source frameworks were utilized such as a) Python 3.6, the programming language; b) OpenCV 4.1.0, the image processing library; c) Numpy 3.7, Python's fundamental package for scientific computing; and d) Anaconda, a distribution of the Python programming. Furthermore, CRAFT uses a fully convolutional network architecture based on the VGG-16 backbone.

Meanwhile, Matlab R2018a was used in the implementation of blind deconvolution. It combines a desktop environment that is optimized for iterative analysis and design with a programming language that natively represents matrix and array mathematics.

All the numerous computational experiments in this research were performed using a Dell Inspiron 7460 laptop with Intel Core i7-7500U CPU at 2.70GHz and NVIDIA GeForce 940MX.

Experimentations with the BD-CRAFT Algorithm

BD-CRAFT is a recently proposed technique for text detection (Albarillo & Fernandez, 2022). It is a variant of the CRAFT algorithm, with a significantly improved performance due to primarily two image preprocessing techniques that are executed before running the main method of CRAFT:

1. It employs the Laplacian operator, with a threshold set to 100, to automatically detect blurry input images
2. It deblurs the detected blurry images using Blind Deconvolution, with the point spread function (PSF) set to 1,3 since this set of PSF values yielded the best results among the other explored PSFs.

A flowchart describing the main operations of BD-CRAFT is provided in Figure 1.

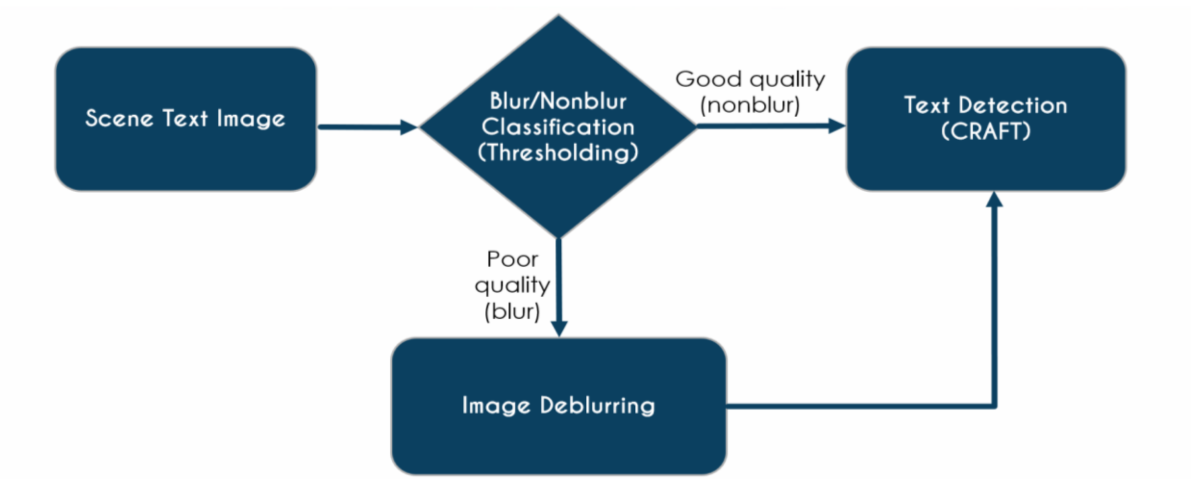

Figure 1. Flowchart of the proposed BD-CRAFT

The main part of this study involves the exploration of combining BD-CRAFT with some state-of-the-art algorithms. The top three state-of-the-art algorithms which include SenseTime, TextFuseNet, and TencentAILab were experimented on. Since their actual codes are not available for download, the published results in the Robust Reading Competition including the actual performances (precision, recall, and h-mean) of each image in the ICDAR 2013 were used. The method involves only the 63 identified blurry images and the performance results of the target algorithm (SenseTime, TextFuseNet, or TencentAILab) are collected and compared with the results of each image when running BD-CRAFT.

The Evaluation Metrics

Intersection over Union (IoU), a well-known similarity metric that is evaluated as the ratio of two entities - the overlapping area and the union area - is used to determine the accuracy of the proposed method. IoU measures how well the predicted bounding box overlaps with the actual box in the context of our challenge. It ranges from 0 (no overlap) to 1, which is the optimal value (perfect overlap). An accurate localization of the texts is indicated by an IoU value of 0.5 or higher, which is regarded as a good prediction (Rosebrock, 2016) whereas IoU scores below 0.5 are considered poor prediction.

To assess the IoU performance of the method, the precision, recall, and eventually, the h-mean are computed from this prediction. Precision is a metric that quantifies the number of correct positive predictions made. It evaluates the fraction of correctly classified instances among the ones classified as positive. Recall quantifies the number of correct positive predictions made out of all positive predictions that could have been made. It measures the proportion of valid positive predictions out of all possible positive predictions. Unlike precision which only comments on the correct positive predictions out of all positive predictions, recall indicates missed positive predictions. Precision and recall can be combined into one metric using H-mean, which covers both characteristics. The formulas for these well-known measures are provided below for completeness:

$$Precision = \frac{TP}{TP+FP}; \quad \textit{Equation 1}$$

$$Recall = \frac{TP}{TP+FN}; \text{ and} \quad \textit{Equation 2}$$

$$Hmean = 2\frac{(Recall*Precision)}{(Recall+Precision)} \quad \textit{Equation 3}$$

The Precision and Recall can be further represented by the equations below. Precision is determined by dividing the total number of correctly anticipated positive examples by the ratio of correctly predicted positive examples while recall is calculated as the sum of true positives across all classes divided by the sum of true positives and false negatives across all classes.

$$Precision(G,D) = \frac{\sum_{j=1}^{|D|} Bestmatch_D(D_j)}{|D|} \quad \textit{Equation 4}$$

$$Recall(G,D) = \frac{\sum_{i=1}^{|G|} Bestmatch_G(G_i)}{|G|} \quad \textit{Equation 5}$$

where $Bestmatch_D$ and $Bestmatch_G$ indicate the closest match between detection and ground truth as defined below:

$$Bestmatch_G(G_i) = \frac{2 \cdot Area(G_i \cap D_j)}{Area(G_i) + Area(D_j)} \quad \textit{Equation 6}$$

$$Bestmatch_D(D_j) = \frac{2 \cdot Area(D_j \cap \; G_i)}{Area(D_j) + \; Area(G_i)}$$ *Equation 7*

Note that H-mean refers to the harmonic mean of the Precision and Recall and therefore takes into account both false positives and false negatives.

Comparison with State-of-the-Art Algorithms

Finally, the IoU h-mean results of the mentioned top three algorithms were compared against those of the modified versions (i.e., the versions that incorporate BD-CRAFT). The contributory value of BD-CRAFT is established after showing that each of these three algorithms yields better IoU h-means after incorporating BD-CRAFT.

RESULTS AND DISCUSSION

Blind deconvolution is a technique for recovering a scene from a blurred image using a point spread function (PSF) that is poorly recognized or unknown. The PSF describes how much a point of light is spread out (blurred) by an optical system. blind deconvolution maximizes the likelihood that the output recovered image, when convolved with a specific PSF, is an instance of the input blurry image. PSF reconstruction begins with a uniform array (array of ones), a pair of parameter values, each time. The number of pixels applied in each dimension (x, y) during restoration will depend on the combination of parameter values.

Figure 2 shows the original image (a) and a selected set of images (b) to (f) restored through blind deconvolution using different PSFs as indicated. A deblurred image first undergoes blind deconvolution, which is followed by standard text detection with CRAFT. Figure 3 further demonstrates how analyzing the reconstructed PSFs may contribute to determining the appropriate PSF values for the image.

A technique to pre-classify images as either blurry or non-blurry is included wherein the deblurring technique is applied only to images that are automatically detected as blurry. This is accomplished by utilizing the Laplacian operator, a differential operator produced by the divergence of the gradient of a scalar function in Euclidean space. The Laplacian can be used to highlight areas of an image with rapid changes in intensity. Thus, it is frequently used in edge detection.

The average variance of the Laplacian is used to describe the blurriness of an image as a single-point value. The higher the number, the sharper the edges in the image are. Therefore, this value is simply computed for the input image and is compared against a specified threshold to automatically identify an image as either blurry or non-blurry.

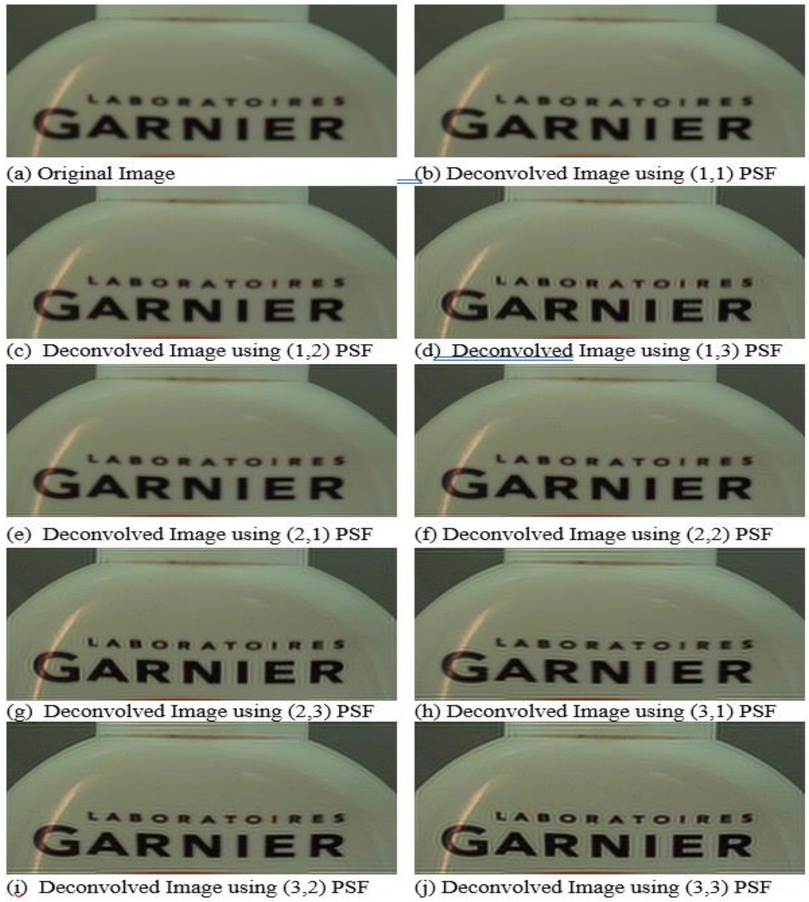

Figure 2. Sample image which underwent blind deconvolution using different PSFs as indicated.

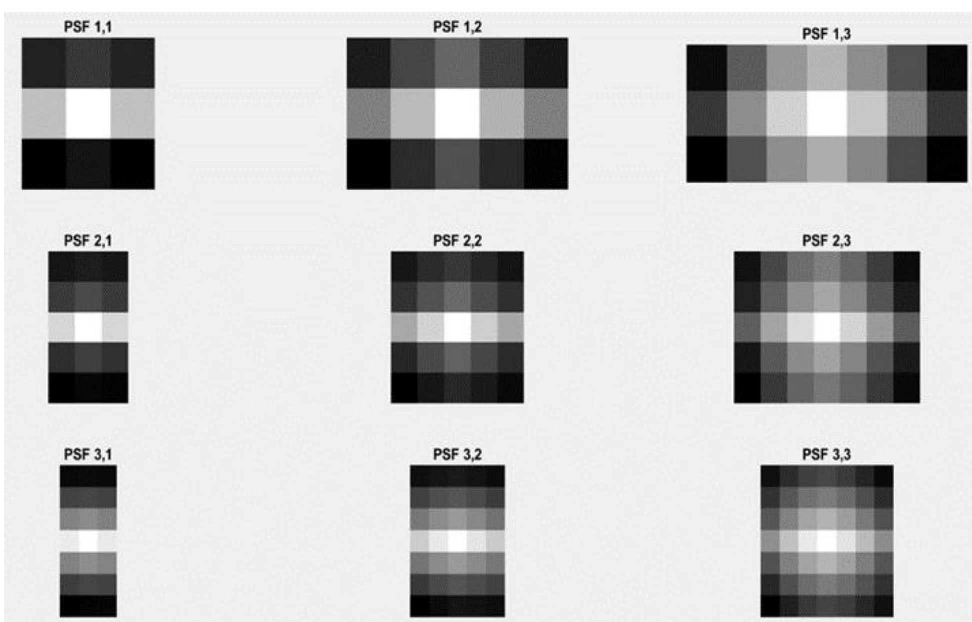

Figure 3. Reconstructed PSFs of the sample image using different PSFs as indicated.

After assessing the input scene image, if the image has a focus measure higher than the supplied threshold of 100, it is considered non-blurry and the image will then immediately undergo scene detection using CRAFT. Otherwise, the scene image further undergoes another preprocessing step (blind deconvolution) first. From the 233 images in the ICDAR 2013 dataset, exactly 63 were classified as blurry and 170 as non-blurry when the threshold 100 was used. Refer to Figure 4 for some example images. The two images on the top, with blur measures of 7.97 and 28.99, are regarded to be blurry. The last two images (bottom) are considered non-blurry, with measures of 265.99 and 693.66.

Figure 4. Example images from the ICDAR 2013 dataset, with their focused measures, indicated.

The Effect of Blind Deconvolution the Identified Blurry Images

Table 1 presents the evaluation results for the 63 identified blurry images together with their corresponding threshold of blurriness when using CRAFT and BD-CRAFT. The precision, recall, and h-mean of the said images are also presented.

Using (1,3) as the PSF for BD-CRAFT, the updated ranking of the state-of-the-art algorithms for Text Detection shows BD-CRAFT ranked top. Observe that BD-CRAFT outperforms the first proposed method (which employs Blind Deconvolution in all images in the dataset) across all the performance metrics. Its IoU h-mean is 3.05% greater than that of the original CRAFT and 0.85% (absolute) higher than that of SenseTime. The top-ranking performance was achieved because of the very impressive precision of 95.24%, which tops all precision results in the table, and is significantly higher than SenseTime, by over 3%.

Table 1. Evaluation Performance of the 63 Blurry Images using BD-CRAFT vs CRAFT

Blurry Images	Blurriness	CRAFT			BD-CRAFT		
Img_no	**Threshold**	**Precision**	**Recall**	**H-Mean**	**Precision**	**Recall**	**H-Mean**
1	12.8405571	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
4	16.4762852	80.00%	57.14%	66.67%	**100.00%**	**85.71%**	**92.31%**
10	23.4311781	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
12	36.653517	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
15	28.9903253	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
16	16.1522402	0.00%	0.00%	0.00%	**100.00%**	**50.00%**	**66.67%**
20	57.4975666	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
21	81.7039225	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
23	15.9194037	100.00%	100.00%	100.00%	**94.12%**	**100.00%**	**96.87%**
25	7.97007126	33.33%	50.00%	40.00%	**100.00%**	**100.00%**	**100.00%**
26	35.1926793	90.91%	76.92%	83.33%	**100.00%**	**100.00%**	**100.00%**
29	50.7909314	100.00%	60.00%	75.00%	**75.00%**	**60.00%**	**66.67%**
34	16.1037522	100.00%	100.00%	100.00%	**45.45%**	**100.00%**	**62.50%**
38	14.3117793	100.00%	100.00%	100.00%	**100.00%**	**100.00%**	**100.00%**
39	15.4544795	88.89%	80.00%	84.21%	**83.33%**	**100.00%**	**90.91%**
44	27.4993814	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
45	21.6460108	76.47%	86.67%	81.25%	**100.00%**	**93.33%**	**96.55%**
48	17.6449657	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
49	56.3667426	100.00%	66.67%	80.00%	**100.00%**	**83.33%**	**90.90%**
52	99.1999698	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
54	39.2921302	71.43%	83.33%	76.92%	**100.00%**	**100.00%**	**100.00%**
55	83.6990885	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
58	41.8884998	100.00%	77.78%	87.50%	100.00%	77.78%	87.50%
59	97.4825748	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
61	61.5662565	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
62	64.4650431	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
63	66.767582	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
64	18.0964371	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
65	12.5359277	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
69	28.2970963	100.00%	100.00%	100.00%	**50.00%**	**100.00%**	**66.67%**
73	78.597045	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
75	62.6589596	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
77	88.4263075	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
82	56.0501649	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
84	22.8809663	100.00%	83.33%	90.91%	**71.43%**	**83.33%**	**76.92%**
85	68.5711141	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
87	49.7199117	100.00%	100.00%	100.00%	**80.00%**	**100.00%**	**88.89%**
88	81.8170171	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
89	14.206154	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
93	13.5899878	50.00%	100.00%	66.67%	**100.00%**	**100.00%**	**100.00%**
95	57.426799	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
96	11.8740355	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
98	61.3983121	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
123	54.5686799	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
125	29.4063831	83.33%	71.43%	76.92%	**100.00%**	**100.00%**	**100.00%**
131	65.3736027	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
134	45.5904438	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%

Table 1. Evaluation Performance of the 63 Blurry Images using BD-CRAFT vs CRAFT (cont.)

Blurry Images	Blurriness	CRAFT			BD-CRAFT		
Img_no	Threshold	Precision	Recall	H-Mean	Precision	Recall	H-Mean
138	72.2330041	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
142	73.576066	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
143	24.0986572	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
180	10.3440649	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
181	84.1780428	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
183	81.9879789	100.00%	100.00%	100.00%	**100.00%**	**50.00%**	**66.67%**
184	26.0136719	80.00%	100.00%	88.89%	**100.00%**	**100.00%**	**100.00%**
186	35.5071598	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
211	22.1367073	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
217	95.5176921	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
222	65.4059773	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
226	43.1260097	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
227	44.3401323	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
229	16.0993804	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
231	60.5148326	50.00%	33.33%	40.00%	**83.33%**	**100.00%**	**90.91%**
232	17.4316837	75.00%	100.00%	85.71%	**100.00%**	**100.00%**	**100.00%**

The Proposed BD-CRAFT Algorithm for Text Detection

Using the insights gathered from previous experiments, BD-CRAFT, an improved technique for scene text detection is proposed.

Blurry/Non-blurry Classification. Using the Laplacian operator and a threshold of 100, the scene text image is automatically classified in this stage as either blurry or non-blurry.

Blind deconvolution. Blind deconvolution is used to deblur images that have been identified as blurry. Notably, images that are classified as non-blurry do not undergo this step.

Text Detection using CRAFT. After classifying images as blurry or non-blurry and then preprocessing the blurry images using blind deconvolution, the text detection step takes place. The Character Region Awareness for Text Detection (CRAFT) algorithm was utilized in this experiment to detect text areas or regions.

To establish that the proposed method is effective in improving text detection performance, the said method was also used in the top three algorithms of the ICDAR 2013 competition. Table 2 shows the performance of BD-CRAFT and SenseTime, respectively. The table illustrates that when BD-CRAFT was applied, 11 images performed better in terms of h-mean than using SenseTime, whereas 8 and 9 images performed better in terms of precision and recall. However, when BD-CRAFT is used, nine images obtained lower h-mean results. The images which have improved results when using BD-CRAFT outnumbered the images which had lower results than SenseTime. This partly explains why BD-CRAFT has better detection resulting in 94.47% than SenseTime which only garnered a 93.62% h-mean result.

Table 2. Evaluation Performance of the 63 Blurry Images using BD-CRAFT vs SenseTime

Blurry Images	Blurriness	BD-CRAFT			SenseTime		
Img_no.	Threshold	Precision	Recall	H-Mean	Precision	Recall	H-Mean
1	12.8405571	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
4	16.4762852	**100.00%**	**85.71%**	**92.31%**	71.43%	71.43%	71.43%
10	23.4311781	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
12	36.653517	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
15	28.9903253	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
16	16.1522402	**100.00%**	**50.00%**	**66.67%**	100.00%	100.00%	100.00%
20	57.4975666	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
21	81.7039225	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
23	15.9194037	**94.12%**	**100.00%**	**96.87%**	100.00%	100.00%	100.00%
25	7.97007126	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
26	35.1926793	**100.00%**	**100.00%**	**100.00%**	100.00%	84.62%	91.67%
29	50.7909314	**75.00%**	**60.00%**	**66.67%**	80.00%	80.00%	80.00%
34	16.1037522	**45.45%**	**100.00%**	**62.50%**	100.00%	100.00%	100.00%
38	14.3117793	**100.00%**	**100.00%**	**100.00%**	50.00%	100.00%	66.67%
39	15.4544795	**83.33%**	**100.00%**	**90.91%**	66.67%	80.00%	72.73%
44	27.4993814	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
45	21.6460108	**100.00%**	**93.33%**	**96.55%**	100.00%	100.00%	100.00%
48	17.6449657	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
49	56.3667426	**100.00%**	**83.33%**	**90.90%**	75.00%	75.00%	75.00%
52	99.1999698	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
54	39.2921302	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
55	83.6990885	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
58	41.8884998	100.00%	77.78%	87.50%	100.00%	77.78%	87.50%
59	97.4825748	**100.00%**	**100.00%**	**100.00%**	75.00%	75.00%	75.00%
61	61.5662565	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
62	64.4650431	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
63	66.767582	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
64	18.0964371	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
65	12.5359277	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
69	28.2970963	**50.00%**	**100.00%**	**66.67%**	100.00%	100.00%	100.00%
73	78.597045	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
75	62.6589596	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
77	88.4263075	**100.00%**	**100.00%**	**100.00%**	50.00%	50.00%	50.00%
82	56.0501649	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
84	22.8809663	**71.43%**	**83.33%**	**76.92%**	80.00%	66.67%	72.73%
85	68.5711141	**100.00%**	**100.00%**	**100.00%**	50.00%	100.00%	66.67%
87	49.7199117	80.00%	100.00%	88.89%	80.00%	100.00%	88.89%
88	81.8170171	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
89	14.206154	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
93	13.5899878	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
95	57.426799	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
96	11.8740355	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
98	61.3983121	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%

Table 2. Evaluation Performance of the 63 Blurry Images using BD-CRAFT vs SenseTime (cont.)

Blurry Images	Blurriness	BD-CRAFT			SenseTime		
Img_no.	Threshold	Precision	Recall	H-Mean	Precision	Recall	H-Mean
123	54.5686799	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
125	29.4063831	**100.00%**	**100.00%**	**100.00%**	100.00%	71.43%	83.33%
131	65.3736027	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
134	45.5904438	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
138	72.2330041	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
142	73.576066	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
143	24.0986572	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
180	10.3440649	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
181	84.1780428	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
183	81.9879789	**100.00%**	**50.00%**	**66.67%**	75.00%	75.00%	75.00%
184	26.0136719	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
186	35.5071598	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
211	22.1367073	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
217	95.5176921	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
222	65.4059773	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
226	43.1260097	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
227	44.3401323	**100.00%**	**100.00%**	**100.00%**	100.00%	75.00%	85.71%
229	16.0993804	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
231	60.5148326	**83.33%**	**100.00%**	**90.91%**	100.00%	100.00%	100.00%
232	17.4316837	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%

Meanwhile, Table 3 shows the evaluation performance on the 63 identified blurry images using BD-CRAFT and TextFuseNet. TextFuseNet ranks second based on the published results of the ICDAR 2013 Focused Scene Text Detection Challenge. When BD-CRAFT is applied, 9 images had better results than that of TextFuseNet while 5 images had improved results in terms of precision and 8 images had better recall results.

However, 8 images have higher h-mean results when using TextFuseNet. Comparing the number of images that gained better performance, it is evident that BD-CRAFT has better results than that of TextFuseNet. This resulted in a 93.11% detection performance of TextFuseNet whose h-mean score is compared to the 94.47% h-mean of BD-CRAFT.

Further experiments on the evaluation of the text detection performance of BD-CRAFT and TencentAILab were also conducted. Results in Table 4 show that 9 images yielded better h-mean results when using CRAFT compared to TencentAILab. In terms of precision, 10 images had better results while only 3 images had improved recall results when BD-CRAFT is employed.

On the other hand, 13 images show better h-mean results while 10 images garnered lower h-mean results when using TencentAILab. Meanwhile, 8 images have better results in terms of precision, and six have garnered higher recall results than that BD-CRAFT. Even though TencentAILab has more images which yielded better h-mean results on the 63 identified blurry images, BD-CRAFT still got a higher h-mean resulting in 94.47% while TencentAILab only yielded 93.05%. This result is influenced by the images which are identified as non-blurry which yielded poor detection results when using TencentAILab.

Table 3. Evaluation Performance of the 63 Blurry Images using BD-CRAFT vs TextFuseNet (cont.)

Blurry Images	**Blurriness**	**BD-CRAFT**			**TextFuseNet**		
Img_no.	**Threshold**	**Precision**	**Recall**	**H-Mean**	**Precision**	**Recall**	**H-Mean**
1	12.8405571	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
4	16.4762852	**100.00%**	**85.71%**	**92.31%**	100.00%	57.14%	72.73%
10	23.4311781	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
12	36.653517	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
15	28.9903253	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
16	16.1522402	**100.00%**	**50.00%**	**66.67%**	100.00%	100.00%	100.00%
20	57.4975666	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
21	81.7039225	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
23	15.9194037	**94.12%**	**100.00%**	**96.87%**	100.00%	100.00%	100.00%
25	7.97007126	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
26	35.1926793	**100.00%**	**100.00%**	**100.00%**	100.00%	84.62%	91.67%
29	50.7909314	**75.00%**	**60.00%**	**66.67%**	100.00%	60.00%	75.00%
34	16.1037522	**45.45%**	**100.00%**	**62.50%**	100.00%	100.00%	100.00%
38	14.3117793	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
39	15.4544795	**83.33%**	**100.00%**	**90.91%**	0.00%	0.00%	0.00%
44	27.4993814	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
45	21.6460108	**100.00%**	**93.33%**	**96.55%**	81.25%	86.67%	83.87%
48	17.6449657	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
49	56.3667426	**100.00%**	**83.33%**	**90.90%**	100.00%	75.00%	85.71%
52	99.1999698	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
54	39.2921302	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
55	83.6990885	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
58	41.8884998	**100.00%**	**77.78%**	**87.50%**	100.00%	88.89%	94.12%
59	97.4825748	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
61	61.5662565	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
62	64.4650431	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
63	66.767582	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
64	18.0964371	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
65	12.5359277	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
69	28.2970963	**50.00%**	**100.00%**	**66.67%**	0.00%	0.00%	0.00%
73	78.597045	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
75	62.6589596	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
77	88.4263075	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
82	56.0501649	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
84	22.8809663	**71.43%**	**83.33%**	**76.92%**	100.00%	83.33%	90.91%
85	68.5711141	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
87	49.7199117	**80.00%**	**100.00%**	**88.89%**	100.00%	100.00%	100.00%
88	81.8170171	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
89	14.206154	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
93	13.5899878	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
95	57.426799	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%

Table 3. Evaluation Performance of the 63 Blurry Images using BD-CRAFT vs TextFuseNet (cont.)

Blurry Images	Blurriness	BD-CRAFT			TextFuseNet		
Img_no.	Threshold	Precision	Recall	H-Mean	Precision	Recall	H-Mean
96	11.8740355	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
98	61.3983121	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
123	54.5686799	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
125	29.4063831	**100.00%**	**100.00%**	**100.00%**	62.50%	71.43%	66.67%
131	65.3736027	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
134	45.5904438	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
138	72.2330041	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
142	73.576066	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
143	24.0986572	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
180	10.3440649	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
181	84.1780428	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
183	81.9879789	**100.00%**	**50.00%**	**66.67%**	100.00%	100.00%	100.00%
184	26.0136719	100.00%	100.00%	100.00%	80.00%	100.00%	88.89%
186	35.5071598	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
211	22.1367073	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
217	95.5176921	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
222	65.4059773	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
226	43.1260097	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
227	44.3401323	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
229	16.0993804	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
231	60.5148326	**83.33%**	**100.00%**	**90.91%**	80.00%	66.67%	72.73%
232	17.4316837	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%

To establish that the proposed method has an impact in improving the performance of the different scene text detection algorithms, those images which obtained better results among the 63 identified blurry images in BD-CRAFT compared to the results using the other algorithms such as SenseTime, TextFuseNet and TencentAILab are then selected and eventually used to calculate the average h-mean result of the said algorithm.

An updated ranking of the top-performing algorithms would show an impressive ranking of this proposed algorithm when integrated into the original algorithm (see Table 5). It can be observed that when BD-CRAFT is applied to other algorithms, their h-mean results were improved. When using TextFuseNet alone, the h-mean result is 93.11% but when BD-CRAFT is (TextFuseNet + BD-CRAFT) applied, it yields 93.55% h-mean. Furthermore, when BD-CRAFT (TextFuseNet + BD-CRAFT) is applied, the precision shows an impressive improvement of over 4% as evidenced by its 95.71% precision.

Table 4. Evaluation Performance of the 63 Blurry Images using BD-CRAFT vs TencentAILab

Blurry Images	Blurriness	BD-CRAFT			TencentAIlab		
Img_no.	Threshold	Precision	Recall	H-Mean	Precision	Recall	H-Mean
1	12.8405571	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
4	16.4762852	**100.00%**	**85.71%**	**92.31%**	100.00%	87.50%	93.33%
10	23.4311781	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
12	36.653517	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
15	28.9903253	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
16	16.1522402	**100.00%**	**50.00%**	**66.67%**	100.00%	100.00%	100.00%
20	57.4975666	**100.00%**	**100.00%**	**100.00%**	**85.71%**	**100.00%**	**92.31%**
21	81.7039225	100.00%	100.00%	100.00%	**93.33%**	**93.33%**	**93.33%**
23	15.9194037	**94.12%**	**100.00%**	**96.87%**	100.00%	100.00%	100.00%
25	7.97007126	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
26	35.1926793	**100.00%**	**100.00%**	**100.00%**	100.00%	84.62%	91.67%
29	50.7909314	**75.00%**	**60.00%**	**66.67%**	**83.33%**	**100.00%**	**90.91%**
34	16.1037522	**45.45%**	**100.00%**	**62.50%**	**100.00%**	**100.00%**	**100.00%**
38	14.3117793	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
39	15.4544795	**83.33%**	**100.00%**	**90.91%**	**100.00%**	**100.00%**	**100.00%**
44	27.4993814	100.00%	100.00%	100.00%	**90.00%**	**100.00%**	**94.74%**
45	21.6460108	**100.00%**	**93.33%**	**96.55%**	**100.00%**	**100.00%**	**100.00%**
48	17.6449657	**100.00%**	**100.00%**	**100.00%**	**83.33%**	**100.00%**	**90.91%**
49	56.3667426	**100.00%**	**83.33%**	**90.90%**	**90.91%**	**83.33%**	**86.96%**
52	99.1999698	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
54	39.2921302	100.00%	100.00%	100.00%	**83.33%**	**83.33%**	**100.00%**
55	83.6990885	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
58	41.8884998	**100.00%**	**77.78%**	**87.50%**	**88.89%**	**88.89%**	**88.89%**
59	97.4825748	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
61	61.5662565	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
62	64.4650431	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
63	66.767582	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
64	18.0964371	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
65	12.5359277	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
69	28.2970963	**50.00%**	**100.00%**	**66.67%**	**100.00%**	**100.00%**	**100.00%**
73	78.597045	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
75	62.6589596	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
77	88.4263075	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
82	56.0501649	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
84	22.8809663	**71.43%**	**83.33%**	**76.92%**	100.00%	83.33%	90.91%
85	68.5711141	**100.00%**	**100.00%**	**100.00%**	**50.00%**	**100.00%**	**66.67%**
87	49.7199117	**80.00%**	**100.00%**	**88.89%**	**100.00%**	**100.00%**	**100.00%**
88	81.8170171	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
89	14.206154	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
93	13.5899878	**100.00%**	**100.00%**	**100.00%**	**50.00%**	**100.00%**	**66.67%**
95	57.426799	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
96	11.8740355	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
98	61.3983121	**100.00%**	**100.00%**	**100.00%**	**77.78%**	**100.00%**	**87.5%**
123	54.5686799	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
125	29.4063831	**100.00%**	**100.00%**	**100.00%**	**100.00%**	**71.43%**	**83.33%**
131	65.3736027	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
134	45.5904438	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
138	72.2330041	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
142	73.576066	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
143	24.0986572	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
180	10.3440649	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
181	84.1780428	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
183	81.9879789	**100.00%**	**50.00%**	**66.67%**	**75.00%**	**75.00%**	**75.00%**
184	26.0136719	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
186	35.5071598	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
211	22.1367073	100.00%	100.00%	100.00%	66.67%	66.67%	66.67%
217	95.5176921	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
222	65.4059773	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
226	43.1260097	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
227	44.3401323	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
229	16.0993804	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
231	60.5148326	**83.33%**	**100.00%**	**90.91%**	**100.00%**	**100.00%**	**100.00%**
232	17.4316837	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%

Moreover, TencentAILab has an h-mean result of 93.05% and its detection performance was improved when BD-CRAFT (TencentAILab + BD-CRAFT) was employed. TencentAILab + BD-CRAFT has yielded an impressive h-mean result of 94.77% which ranked second and outperformed the state-of-the-art SenseTime with 93.62%. In addition, both the precision and recall of TencentAILab were improved.

Finally, the state-of-the-art algorithm SenseTime is also explored. When applying BD-CRAFT (SenseTime + BD-CRAFT), it resulted in a very impressive 95.22% h-mean and showed a huge precision improvement of over 4% which made it to be the top-ranked algorithm.

Table 5. Comparison of the State-of-the-art algorithms ranked by IoU h-mean

Method	Precision	Recall	H-Mean
SenseTime + BD-CRAFT	**96.79 %**	**94.65 %**	**95.22 %**
TencentAILab + BD-CRAFT	***94.04%***	***96.46%***	***94.77%***
SenseTime (2016)	91.87%	95.45%	93.62%
TextFuseNet + BD-CRAFT	***95.71%***	***94.45%***	***93.55%***
BD-CRAFT	94.32%	92.44%	93.37%
TextFuseNet (2020)	90.78%	95.58%	93.11%
TencentAILab (2017)	94.79%	91.37%	93.05%
VARCO (2020)	89.86%	93.63%	91.71%
HIT (2020)	89.22%	93.85%	91.48%
CRAFT (2018)	89.04%	93.93%	91.42%

CONCLUSIONS AND RECOMMENDATIONS

In this study, we improve three (3) state-of-the-art algorithms for text detection – SenseTime, TextFuseNet, and TencentAILab – by incorporating the BD-CRAFT, a variant of the CRAFT algorithm that involves preprocessing steps where images are automatically classified as blurry or non-blurry using a Laplacian operator, followed by applying the blind deconvolution deblurring technique. Each of the resulting algorithm variants shows significant improvement as evidenced by the increase not only in the overall h-mean but also in some of the precision and recall values. TextFuseNet + BD-CRAFT yields 93.55% h-mean, while the precision is 95.71%, which is an impressive improvement of over 4%. Meanwhile, TencentAILab + BD-CRAFT showed an impressive h-mean result of 94.77% (which ranked second and had outperformed the state-of-the-art SenseTime's 93.62%), with both the precision and recall improving also. Furthermore, SenseTime + BD-CRAFT ranked first with a very impressive 95.22% h-mean and showed a huge precision improvement of over 4%, which made it to be the top-ranked algorithm. Evidence shows that when BD-CRAFT is combined with other algorithms, their performances are improved, hence BD-CRAFT has a significant impact on the text detection performance of these algorithms.

As possibilities for further studies, it would be interesting to investigate the other state-of-the-art algorithms for scene text detection that would benefit also from the discussed method. It may also be good to investigate other preprocessing techniques that can be incorporated into text detection algorithms in general.

ACKNOWLEDGEMENT

The completion of this undertaking could not have been possible without the participation and assistance of many individuals whose names may not all be enumerated, their contributions are sincerely appreciated and gratefully acknowledged. We have been blessed in the preparation of this work with the blessings and graces of the Lord Jesus Christ. Through His guidance, we had certainly overcome great struggles and deprivations. We also would like to express our appreciation to CHED, Ateneo de Manila University, and DMMMSU for their support towards the fulfillment of all our scholarly endeavors. It is the kindness of these acknowledged that this research sees the light of day. We submit this work with great humility and utmost regard.

DECLARATIONS

Conflict of Interest

All authors declare that they have no conflicts of interest.

Informed Consent

The study did not involve humans as participants and only used scene text datasets which are already available online hence this is not applicable.

Ethics Approval

The conducted research uses only scene text datasets which are already available online and did not include humans as participants hence this is not applicable.

REFERENCES

Albarillo, C. and Fernandez, P. (2022). BD-CRAFT: Applying Blind Deconvolution towards Improved Scene Text Detection of the CRAFT Algorithm. *Submitted.*

Baek, Y., Lee, B., Han, D., Yun, S., & Lee, H. (2019). Character region awareness for text detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 9365-9374).

Bai, X., Yang, M., Lyu, P., & Xu, Y. (2017). Integrating scene text and visual appearance for fine-grained image classification with convolutional neural networks. *arXiv preprint arXiv:1704.04613*, *1*(4).

Barber, D. B., Redding, J. D., McLain, T. W., Beard, R. W., & Taylor, C. N. (2006). Vision-based target geo-location using a fixed-wing miniature air vehicle. *Journal of Intelligent and Robotic Systems*, *47*(4), 361-382.

Brodić, D., & Amelio, A. (2016, September). Analysis of the human-computer interaction on the example of image-based CAPTCHA by association rule mining. In *International Workshop on Symbiotic Interaction* (pp. 38-51). Springer, Cham.

Busta, M., Neumann, L., & Matas, J. (2015). Fastext: Efficient unconstrained scene text detector. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 1206-1214).

DeSouza, G. N., & Kak, A. C. (2002). Vision for mobile robot navigation: A survey. *IEEE transactions on pattern analysis and machine intelligence*, *24*(2), 237-267.

Ding, M., Sun, Z., Wei, L., Cao, Y., & Yao, Y. (2019). Infrared target detection and recognition method in airborne photoelectric system. *Journal of Aerospace Information Systems*, *16*(3), 94-106.

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

Fornés, A., Romero, V., Baró, A., Toledo, J. I., Sánchez, J. A., Vidal, E., & Lladós, J. (2017). ICDAR2017 competition on information extraction in historical handwritten records. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)* (Vol. 1, pp. 1389-1394). IEEE.

Gao, Y., Li, X., Wang, X. V., Wang, L., & Gao, L. (2021). A review on recent advances in vision-based defect recognition towards industrial intelligence. *Journal of Manufacturing Systems*.

Gordo, A., Almazán, J., Revaud, J., & Larlus, D. (2016, October). Deep image retrieval: Learning global representations for image search. In *European Conference on Computer Vision* (pp. 241-257). Springer, Cham.

Ham, Y. K., Kang, M. S., Chung, H. K., Park, R. H., & Park, G. T. (1995). Recognition of raised characters for automatic classification of rubber tires. *Optical Engineering*, *34*(1), 102-109.

He, P., Huang, W., Qiao, Y., Loy, C. C., & Tang, X. (2016). Reading scene text in deep convolutional sequences. In *Thirtieth AAAI conference on artificial intelligence*.

Huang, W., Lin, Z., Yang, J., & Wang, J. (2013). Text localization in natural images using stroke feature transform and text covariance descriptors. In Proceedings of the IEEE International Conference on Computer Vision (pp. 1241-1248).

Iwamura, M., Matsuda, T., Morimoto, N., Sato, H., Ikeda, Y., & Kise, K. (2016). Downtown Osaka scene text dataset. In *European Conference on Computer Vision* (pp. 440-455). Springer, Cham.

Jaderberg, M., Simonyan, K., Vedaldi, A., & Zisserman, A. (2016). Reading text in the wild with convolutional neural networks. *International Journal of Computer Vision*, *116*(1), 1-20.

Karaoglu, S., Tao, R., Gevers, T., & Smeulders, A. W. (2016). Words matter: Scene text for image classification and retrieval. *IEEE transactions on multimedia*, *19*(5), 1063-1076.

Karatzas, D., Shafait, F., Uchida, S., Iwamura, M., i Bigorda, L. G., Mestre, S. R., ... & De Las Heras, L. P. (2013). ICDAR 2013 robust reading competition. In *2013 12th International Conference on Document Analysis and Recognition* (pp. 1484-1493). IEEE.

Kechagias-Stamatis, O., & Aouf, N. (2017). Evaluating 3D local descriptors for future LIDAR missiles with automatic target recognition capabilities. *The Imaging Science Journal*, 65(7), 428-437.

Kim, S., & Park, T. (2020). Learning Robust Feature Representations for Scene Text Detection. *arXiv preprint arXiv:2005.12466*.

Kisacanin, B., Pavlovic, V., & Huang, T. S. (Eds.). (2005). *Real-time vision for human-computer interaction*. Springer Science & Business Media.

Li, Y., Jia, W., Shen, C., & van den Hengel, A. (2014). Characterness: An indicator of text in the wild. *IEEE transactions on image processing*, *23*(4), 1666-1677.

Macesanu, G., & Moldoveanu, F. (2010). Computer vision-based Mobile robot navigation in unknown environments. *Bulletin of the Transilvania University of Brasov. Engineering Sciences. Series I*, *3*, 259.

Nayef, N., Yin, F., Bizid, I., Choi, H., Feng, Y., Karatzas, D., & Ogier, J. M. (2017). Icdar2017 robust reading challenge on multi-lingual scene text detection and script identification-rrc-mlt. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)* (Vol. 1, pp. 1454-1459). IEEE.

Neumann, L., & Matas, J. (2015, August). Efficient scene text localization and recognition with local character refinement. In *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on* (pp. 746-750). IEEE.

Nguyen, D., Shijian, L., Ouarti, N., & Mokhtari, M. (2017, March). Text-Edge-Box: An Object Proposal Approach for Scene Texts Localization. In *Applications.*

Panchpor, A. A., Shue, S., & Conrad, J. M. (2018, January). A survey of methods for mobile robot localization and mapping in dynamic indoor environments. In *2018 Conference on Signal Processing And Communication Engineering Systems (SPACES)* (pp. 138-144). IEEE.

Pech-Pacheco, J. L., Cristóbal, G., Chamorro-Martinez, J., & Fernández-Valdivia, J. (2000). Diatom autofocusing in brightfield microscopy: a comparative study. In *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000* (Vol. 3, pp. 314-317). IEEE.

Renshaw, C. K., Hewitt, J., Grimming, R., & Driggers, R. G. (2022, July). Sensor Optimization for Image-Based Geolocation. In *Computational Optical Sensing and Imaging* (pp. JF2D-6). Optica Publishing Group.

Robust Reading Competition - Focused Scene Text Overview. 2020. https://rrc.cvc.uab.es/?ch=2&com=introduction, Online. accessed October 3, 2021.

Robust Reading Competition - IoU Evaluation of Task 1. (2020). Retrieved from https://rrc.cvc.uab.es/?ch=2&com=evaluation&task=1&f=1&e=3.

Rosebrock, A. (2016). Intersection over Union (IoU) for object detection. *línea].[Consultado: 20 de enero de 2020]. Disponible en: https://www. pyimagesearch. com/2016/11/07/intersection-over-union-iou-forobject-detection*.

Shi, B., Bai, X., & Belongie, S. (2017). Detecting oriented text in natural images by linking segments. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2550-2558).

Shi, B., Bai, X., & Yao, C. (2016). An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE transactions on pattern analysis and machine intelligence*, *39*(11), 2298-2304.

Smith, R., Gu, C., Lee, D. S., Hu, H., Unnikrishnan, R., Ibarz, J., & Lin, S. (2016). End-to-end interpretation of the French street name signs dataset. In *European Conference on Computer Vision* (pp. 411-426). Springer, Cham.

Tian, Z., Huang, W., He, T., He, P., & Qiao, Y. (2016, October). Detecting text in natural image with connectionist text proposal network. In European conference on computer vision (pp. 56-72). Springer, Cham.

Tsai, S. S., Chen, H., Chen, D., Schroth, G., Grzeszczuk, R., & Girod, B. (2011, September). Mobile visual search on printed documents using text and low bit-rate features. In *2011 18th IEEE International Conference on Image Processing* (pp. 2601-2604). IEEE.

Veit, A., Matera, T., Neumann, L., Matas, J., & Belongie, S. (2016). Coco-text: Dataset and benchmark for text detection and recognition in natural images. *arXiv preprint arXiv:1601.07140*.

Wang, S., Jiang, F., Zhang, B., Ma, R., & Hao, Q. (2019). Development of UAV-based target tracking and recognition systems. *IEEE Transactions on Intelligent Transportation Systems*, *21*(8), 3409-3422.

Yang, C., Yin, X. C., Yu, H., Karatzas, D., & Cao, Y. (2017). ICDAR2017 robust reading challenge on text extraction from biomedical literature figures (DeTEXT). In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)* (Vol. 1, pp. 1444-1447). IEEE.

Yin, X. C., Zuo, Z. Y., Tian, S., & Liu, C. L. (2016). Text detection, tracking and recognition in video: a comprehensive survey. *IEEE Transactions on Image Processing*, *25*(6), 2752-2773.

Zitnick, C. L., & Dollár, P. (2014, September). Edge boxes: Locating object proposals from edges. In *European conference on computer vision* (pp. 391-405). Springer, Cham.

Zhu, Y., Yao, C., & Bai, X. (2016). Scene text detection and recognition: Recent advances and future trends. *Frontiers of Computer Science*, *10*(1), 19-36.

Zhou, X., Yao, C., Wen, H., Wang, Y., Zhou, S., He, W., & Liang, J. (2017). East: an efficient and accurate scene text detector. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 5551-5560).