A New Context-based Method for Restoring Occluded Text in Natural Scene Images

[1]Ayush Mittal, [2]Palaiahnakote Shivakumara, [1]Umapada Pal, [3]Tong Lu, [4]Michael Blumenstein, and [5]Daniel Lopresti
[1]Computer Vision and Pattern Recognition Unit, Indian Statistical Institute, Kolkata, India. Email: mittalayush939@gmail.com, umapada@isical.ac.in.
[2]Faculty of Computer Science and Information Technology, University of Malaya, Kuala Lumpur, Malaysia. Email: shiva@um.edu.my
[3]National Key Lab for Novel Software Technology, Nanjing University, Nanjing, China. Email: lutong@nju.edu.cn
[4]Faculty of Engineering and Information Technology, University of Technology Sydney, Australia. Email: Michael.Blumenstein@uts.edu.au
[5]Computer Science & Engineering, Lehigh University, Bethlehem, PA, USA. Email: lopresti@cse.lehigh.edu.

Abstract. Text recognition from natural scene images is an active research area because of its important real world applications, including multimedia search and retrieval, and scene understanding through computer vision. It is often the case that portions of text in images are missed due to occlusion with objects in the background. Therefore, this paper presents a method for restoring occluded text to improve text recognition performance. The proposed method uses the GOOGLE Vision API for obtaining labels for input images. We propose to use PixelLink-E2E methods for detecting text and obtaining recognition results. Using these results, the proposed method generates candidate words based on distance measures employing lexicons created through natural scene text recognition. We extract the semantic similarity between labels and recognition results, which results in a Global Context Score (GCS). Next, we use the Natural Language Processing (NLP) system known as BERT for extracting semantics between candidate words, which results in a Local Context Score (LCS). Global and local context scores are then fused for estimating the ranking for each candidate word. The word that gets the highest ranking is taken as the correction for text which is occluded in the image. Experimental results on a dataset assembled from standard natural scene datasets and our resources show that our approach helps to improve the text recognition performance significantly.

Keywords: Text detection, Occluded image, Annotating natural scene images, Natural language processing, Text recognition.

1 Introduction

Over the past decade, the scope and importance of text recognition in natural scene images have been expanding rapidly, driven by new applications, such as robotics, multimedia, as well as surveillance and monitoring. For example, Roy et al. [1] studied forensic applications, where it is necessary to recognize text from multiple views of the

same target captured by CCTV cameras. The method uses text detection and recognition for identifying crime location. Shivakumara et al. [2] addressed the issues of text detection and recognition in marathon images. This method detects text on torso images of marathon runners for tracing and studying a person's behavior. Xue et al. [3] proposed curved text detection in blurred and non-blurred video and natural scene images. The method focuses on addressing challenges caused by blur and arbitrary orientation of text in images. For all the above-mentioned methods, it is noted that the main aim is to achieve a better recognition rate irrespective of the applications. However, none of the methods proposed in the literature focuses on images where a part of the text is missing due to occlusion. In such cases, if we run existing methods on such images, the recognition performance degrades severely. Also, recognition results obtained for broken words may give an incorrect interpretation of the image content because the recognition results will miss the actual meaning of the text. Therefore, restoring the missing text is important to determine the semantics of the image.

(a) Input natural scene image with occluded text (b) The results of text detection

(c) Recognition results before prediction for the detected words: “oxfrd” and “ooksre”

(d) Labels given by GOOGLE Vision API: “Building, Advertisement, Outlet store, Display window”

(e) Candidate words for “oxfrd” and “ooksre” using distance measures with word dictionary: “Oxford” and “Books, Bookstore, Boostrap, Booking”, respectively.

(f) The proposed prediction results: “oxford book store”.

Fig.1. Illustrating the steps for predicting missing text in a natural scene image

It is evident from the illustration shown in Fig.1, where for the input image with occluded text shown in Fig.1(a) and text detection results shown in Fig.1(b), the recognition method does not recognize the text correctly as reported in Fig.1(c). The recognition results of broken words do not exhibit the desired meaning with respect to the content of the image. This is the limitation of state-of-the-art text recognition for the images where a part of the text is missing. Note that for text detection, we use the method called PixelLink [4] as it is robust to complex backgrounds, orientation, and the E2E method [5] for recognition, as it is robust to distortion and different fonts. These limitations of the state-of-the-art approaches motivated us to develop a method

for restoring the missing portions of broken words in natural scene images. The content of an image and the text in it indeed have a high degree of similarity at the semantic level. Therefore, the proposed method explores using the GOOGLE Vision API [6] to obtain labels for the input image as shown in Fig.1(d). This makes sense because the GOOGLE Vision API works based on the relationship between the objects in images. Through the use of a powerful dictionary and lexicon of candidates, we can expect labels which are often close to the broken words because of powerful language models. This shows that one can predict the possible words for incomplete and misspelled words. Based on these observations, the proposed method generates candidate words for each recognized broken word as shown in Fig.1(e). Then it extracts the context between labels and candidate words based on the distance measure, which results in the global context score. Similarly, the proposed approach uses natural language processing [7] for extracting context between candidate words, which results in a local context score. Furthermore, we combine the global and local context scores to estimate the ranking for each word. The word that gets the highest ranking is considered the most likely replacement for the broken word as shown in Fig.1(f).

2 Related Work

There are several methods proposed in the past several years for recognizing text in natural scene images. Most recent methods have explored deep learning models. Cheng et al. [8] proposed arbitrarily-oriented text recognition in natural scene images based on a deep learning approach. They use an Arbitrary Orientation Network (AON) to capture deep features of irregular text directly, which generates character sequences using an attention-based decoder. Tian et al. [9] proposed a framework for text recognition in web videos based on tracking text. Their approach combines information from text detection and tracking for recognizing text in images. Luo et al. [10] proposed a multi-object rectified attention network for scene text recognition. They explore a deep learning model, which is invariant to geometric transformation. The approach works well for images affected by rotation, scaling and to some extent – distortion. Raghunandan et al. [11] proposed multi-script-oriented text detection and recognition in video, natural scene and born-digital images. The work extracts features based on the wavelet transform for detecting characters with the help of an SVM classifier. Next, it applies a Hidden Markov Model (HMM) for recognizing characters and words in images. Qi et al. [12] proposed a novel joint character categorization and localization approach for character level scene text recognition. The idea of the method is to categorize characters by a joint learning strategy such that recognition performance improves. Shi et al. [13] proposed an attentional scene text recognizer with flexible rectification. The work uses a thin-plate spline transformation to handle a variety of text irregularities. The idea behind the method is to avoid pre-processing before recognition such that errors can be reduced to improve the recognition rate. Rong et al. [14] employs unambiguous scene text segmentation with referring expression comprehension. The study proposes a unified deep network to jointly model visual and linguistic information at both the region and pixel levels for understanding text in images. Villamizar et al. [15] proposed a multi-scale sequential network for semantic text segmentation and localization. The

work explores fully convolutional neural networks that apply to a particular case of slide analysis to understand the text in images. Feng et al. [16] proposed an end-to-end framework for arbitrarily shaped text spotting. The method proposes a new differentiable operator named RoISlide, which detects text and recognizes it in the images.

It is noted from the above discussion that existing methods have addressed challenges such as arbitrary orientation, distortion caused by geometrical transformation, irregularly shaped characters, complex backgrounds, low resolution and low contrast for text recognition. Most of the methods explored deep learning in different ways for achieving their results. However, none of these methods addressed the issue of text occlusion in natural scene images.

There are methods related to restoring missing information in natural scene images. For instance, Lee et al. [17] proposed automatic text detection and removal in video sequences. The method uses Spatio-temporal information for achieving its results. The method identifies locations where the text is missing due to occlusion and then removes that obstacle. However, the scope of the method is limted to text removal but not restoration of missing text. Ye et al. [18] proposed text detection and restoration in natural scene images. The method restores text, which is degraded due to perspective distortion, low resolution and other causes but not missing parts of text information due to occlusion. Tsai et al. [19] proposed text-video completion using structure repair and texture propagation. The method considers text as an obstacle, which occludes object information in images. Therefore, the method detects text information and uses an inpainting approach to restore the missing parts of the object information. Mosleh et al. [20] proposed an automatic inpainting scheme for video text detection and removal. The method is also the same as the text removal approach mentioned above, but it does not restore the missing part of text information in images. Zhang et al. and Wu et al. [21, 22] proposed methods for erasing text in natural scene images based on deep learning models. The scope of the methods is limited to the location of the text information, and to erase it such that it does not alter background information in the images. Hence, one can conclude that restoring the missing part of the text in natural scene images is not addressed for improving recognition performance. Thus, we propose a new method, which combines labels generated from the content of images and recognition results for predicting missing words to replace broken words with the help of a Natural Language Processing (NLP) approach. The main contribution of the proposed work is the way in which our approach combines the labels given by the GOOGLE Vision APIs with candidate words provided by Natural Language Processing to predict the likely words, which are to be replaced as broken words in naturals scene images.

3 Proposed Method

In this work, since the GOOGLE Vision API is available publicly for generating annotations for natural scene images, we use this for generating labels for our input images which contain text within natural scenes [6]. In this work, we propose to use the method called PixelLink [4] because the method is state-of-the-art and works well for images affected by the above-mentioned challenges. For recognizing detected text, we propose to use the E2E method [5] as it is robust to degradations, poor quality, orientation, and

irregularly-shaped characters. Also, the method requires fewer training iterations and less training data. The combination of text detection and recognition produces the recognition results for text in images.

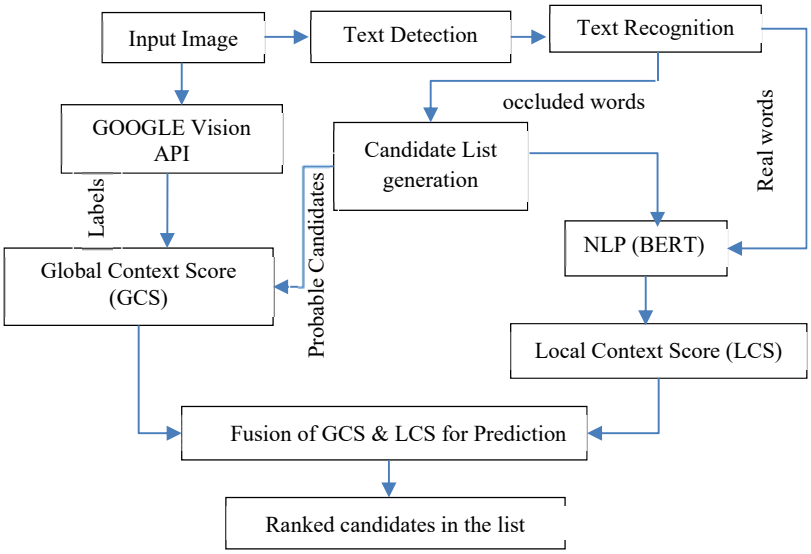

Fig.2. Block Diagram of the proposed method for predicting missing text.

For each word that is recognized, the proposed method generates the sequence of candidates by estimating the distance between the recognized word and the words in a predefined large-sized lexicon formed for scene text recognition. This results in a candidate list of the probable words for each of the non-real words. Then the annotated labels are compared with the list of probable candidates to estimate Global Context Score (GCS) using the distance measure. In the same way, the proposed method estimates the Local Context Score (LCS) for the candidates of each of the word recognition results using the NLP approach known as BERT [7]. The proposed method fuses GCS and LCS scores to generate a ranking for each word. The word that gets the highest rank is considered for a predicted word to replace broken words. The steps and flow of the proposed method can be seen in Fig.2.

3.1 Generating Labels using the Google Vision API

According to the website [6], the GOOGLE Vision API was developed based on a large number of features and deep learning. The system generates a list of labels for each input image with a ranking score. According to our experiments on our dataset, it is noted that if a rank score is greater than 85%, the generated keywords are relevant to the content of the images. Therefore, we set the same threshold empirically for all the experiments in this work. For the input image shown in Fig.3, the system generates labels as listed in the same figure, where it can be seen that all the labels are relevant to the content of the input image. However, sometimes, for images of an unknown place,

the system produces irrelevant keywords. In such cases, there are chances of predicting incorrect words by the proposed method. However, since the proposed method considers candidate words of all the words in images with the help of natural language processing, it may not have much of an effect on the overall performance of the proposed method. This step gives a list of labels for the input images.

Labels

- Building
- Advertisement
- Outlet Store
- Display window

Fig.3. Labels for the natural scene image using GOOGLE Vision API

3.2 Candidate List Generation

Though a portion of the text is missing due to occlusion, still the remaining text can provide some clues for restoring the missing text with the help of lexicons and language processing. Therefore, the proposed method generates possible candidate words for each recognized word including broken words in images. For this, the proposed method uses a deep learning model [4] for text detection in natural scene images to extract the text. Sample results of text detection are shown in Fig.4, where the method detects almost all the text that appeared in images. For the detected words, we use the recognition method proposed in [5] which also employs deep learning models to recognize the detected words as reported in Fig.4, where we can see recognition results for the two words present in Fig.4 as "oxfrd" and "ooksre". The proposed method uses the following steps for producing candidate words for each of the recognized words.

Let $S = \{s_1, s_2, \ldots, s_m\}$ be the sequence of strings separated by spaces obtained from the recognition method which includes good and broken words. It is noted that the broken word creates two new words due to a split in the word. For each word in the list, the proposed method obtains possible candidate words which can replace the broken words. To achieve this, we propose an iterative process to generate possible words using the Levenshtein distance [23] and the subsequence distance. The subsequence distance between two strings is the length of the longest common subsequence (LCS). This process involves the lexicon of 90k generic words available publicly[1]. The effectiveness of this process can be seen in Fig.4, where the list of candidate words is generated for the words "oxfrd" and "ooksre". It is observed from the list of candidate words that the list contains the most likely to be the correct word of the broken words.

[1] The vocabulary can be downloaded from https://rrc.cvc.uab.es/downloads/GenericVocabulary.txt

Candidate List for "oxfrd"
Oxford.
Candidate List for "ooksre"
Books, Bookstore, Bootstrap
Booking

Fig.4. Text detection by the PixelLink method (bounding boxes in the images), recognition results (oxfrd and ooksre) by the E2E method and list of candidate words for the broken words.

3.3 Global and Local Context Score Estimation

The step discussed in Section 3.1 outputs the list of labels for the input image, and the steps presented in Section 3.2 generates the list of candidate words for the words in the images. To extract the semantic similarity between the labels and candidate words, we calculate the distance between them and this gives a global score for all the words in the list. Furthermore, the proposed method considers the maximum word similarity value calculated against all the image labels as the Global Context Score (GCS). It is illustrated in Fig.5, where we can see the distance is estimated for all the combinations of the words between labels and the candidate words using the Euclidean distance measure. Therefore, this process results in a vector, which contains GCS in the range of [0, 1].

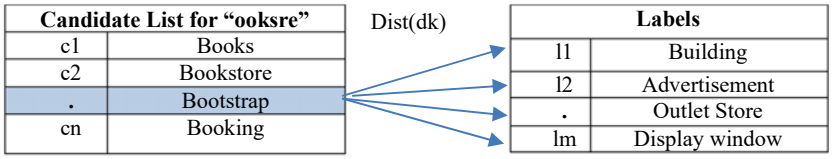

Fig.5. Estimating the Global context score between the label and the candidate words using the Euclidean distance measure. *n* indicates the list of candidate words, *m* indicates the list of labels and *k* indicates the distances.

To extract the local semantic similarity between candidate words, we propose to explore the natural language processing approach called BERT's Masked Language Model [7] for predicting missing words. BERT is bidirectional in nature and we use a pre-defined and trained model in this work. The advantage of this model is that it helps in extracting the context with a limited number of words. For *n* words in a sentence, the proposed method obtains *n* number of lists. Let $L = \{L_1, L_2, L_3, .., L_n\}$ be the sequence of all those n lists, which contains different possible sequences obtained by the combinations of words, say $\boldsymbol{m}$ number of sequences. For each candidate word in the list, logit

values are estimated as follows. For example, for list $\boldsymbol{L_1}$, the logit values for each candidate word are obtained with all the sequences formed using the remaining lists. These logit values are converted to the range [0, 1] using a sigmoid function over every list, which results in the Local Context Score (LCS). In this way, for each candidate word in every list, LCSs are calculated. The values of GCSs and LCSs are fused as defined in Equation (1), where the values of $\boldsymbol{a}$ and $\boldsymbol{b}$ are determined empirically based on predefined samples chosen randomly from the datasets. This process outputs fused values for each word in every sequence. The fused values are further multiplied with a scalar value, $\boldsymbol{z}$ for every sequence as defined in Equation (2) and Equation (3).

$$\boldsymbol{F_i = a(SC_i) + b(GC_i), \forall\, i \in l_k} \quad (\mathbf{1})$$

$$\boldsymbol{z_i = \prod_{j=1}^{n} F_j, \forall\, i \in [1, m]} \quad (\mathbf{2})$$

The best sequence $\boldsymbol{z^*}$ is determined based on the maximum z value as defined in equation

$$\boldsymbol{z^* = \underset{i}{\mathbf{argmax}}\, z_i, \forall\, i \in [1, m]} \quad (\mathbf{3})$$

Table 1. The fused z scores for predicting the correct word using GCS and LCS

Possible combinations with "oxford"	GCS score	LCS score	Fused (z) score
"books"	0.57	0.090	0.051
"bookstore"	**0.65**	**0.120**	**0.078**
"bootstrap"	0.00	0.001	0.000
"booking"	0.53	0.005	0.003

Sample values of GCS and LCS are reported for candidate words given the word "oxford" in Table 1, where we can see the word "bookstore" has received the highest fused values compared to other words. It is also noted that the next relevant word for "oxford" is "books", which scores the next highest value. For the word "bootstrap", the fused score is zero. This indicates that the word is not relevant to the word "oxford" In this way, the proposed method uses labels generated by the GOOGLE Vision API and NLP for predicting missing words in natural scene images.

4 Experimental Results

There is no standard dataset available for restoring missing words to replace broken ones in natural scene images, so we have created our own dataset, which include images captured by ourselves and images collected from FLICKER and SUN datasets, which gives 200 images with occluded text on a different background. As shown in the sample images in Fig.6, we can see that each image has a different background complexity, and includes occluded text. It is also noted from Fig.6 that the occlusions are due to trees, poles and other objects. The percentage of occluded words is around 25.9% out of 478 text instances in our dataset. Also, our dataset includes text in multiple scripts, text

affected by blur, and text in different resolutions as reported in Table 2. We also collected a few images from standard datasets of natural scene images, namely, ICDAR 2015. This dataset contains very few images with occluded text (2%) as they are created for text detection and recognition but not for the application we are targeting. The images of this dataset have almost the same resolution as reported in Table 1. For the GCS calculation, we use the ground truth of the ICDAR 2015 dataset where we found 3909 instances of text line information.

Fig.6. Sample images from our dataset with occluded scene text

For measuring the performance of the proposed method, we consider the standard measures, namely, Precision (P) which is defined as the number of words recognized correctly (real word) divided by the number of words recognized correctly (real words) + false positives, and Recall which is defined as the number of words recognized correctly (real words) divided by the total number of ground truth words. The F-measure is a harmonic mean of Recall and Precision.

Table 2. Statistical analysis of both ours and the benchmark dataset

Dataset	Dataset Size	Resolution	Blur	MLT Script	Uneven illumination	Percentage of occluded words	Number of text instances
Our Dataset	200 Test	Min: 428×476 Max: 4608×2304	✓	✗	✓	25.94%	478
ICDAR 2015	500 Test	720x1280	✗	✗	✗	2%	3909

In this work, we use the text detection method [4] for extracting text from natural scene images. As mentioned in the proposed methodology section, occluded text does not affect text detection performance. This is justified because text detection methods are capable of detecting a single character as text in natural scene images. Also, for text

detection, semantic or context of words in the text line is not necessary unlike recognition which requires a language model for accurate recognition. The method in [4] is considered as the state-of-the-art method for text detection in this work. To validate the above statements, we conducted text detection experiments using different existing methods on our dataset. We implemented the following existing methods including PixelLink and E2E methods. Zhou et al. [24] proposed an efficient and accurate scene text detector, which explores deep learning models for addressing arbitrary orientation of text detection in natural scene images. Shi et al. [25] proposed the detection of oriented text in natural scene images by linking segments that focus on the use of fewer training samples for text detection in natural scene images. Liu et al. [26] proposed a method for detecting curved text in natural scene images, which focuses on the challenge of curved text detection. Busta et al. [5] proposed an end-to-end text detection and recognition method, which focuses on images of multi-lingual scene text. Deng et al. [4] proposed a method for detecting scene text via instance segmentation which addresses the challenges of arbitrary orientation, multi-script and different types of texts. The results of the above methods are reported in Table 3 for our dataset, where it can be noted that the PixelLink method is the best for all three measures. This is the advantage of the PixelLink method compared to the other existing methods. It is also noted from Table 3 that almost all the methods score more than 80% accuracy for our dataset. This demonstrates that the presence of occlusion does not have much of an effect on text detection performance.

Table 3. Text detection performance of the different methods on our dataset.

Methods	P	R	F
Zhou et al. EAST[24]	0.81	0.76	0.78
Shi et al. SegLink[25]	0.83	0.79	0.81
Liu et al. CTD [26]	0.80	0.81	0.80
Busta et al. E2E MLT[5]	0.82	0.83	0.82
Deng et al. PixelLink[4]	**0.84**	**0.86**	**0.84**

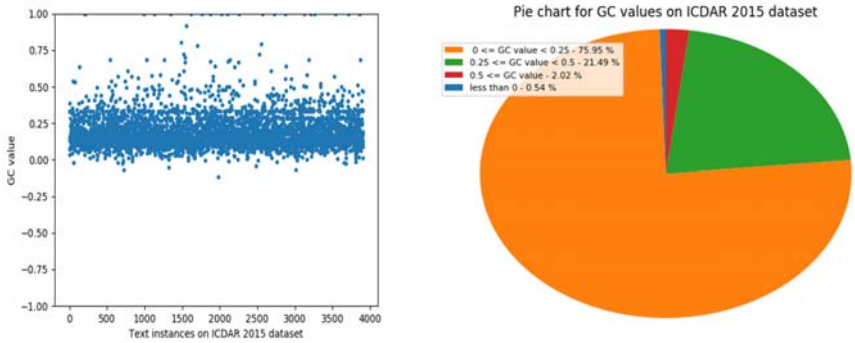

Fig.7. GCS for labels of the images with ground text instances of the ICDAR 2015 ground truth.

In the same way, we use labels given by the GOOGLE Vision API system for input images to restore missing parts. To show that the labels generated by the GOOGLE Vision API are relevant to the content of images, we calculate the Global Context Score (GCS) for the labels with the ground truth provided by the ICDAR 2015 dataset. The process involves 500 images and around 3909 text instances. The GCS values are plotted in Fig.7, where one can see that most of the GCS values are greater than zero as shown in the scattered graph and pie-chart. This shows that the GOOGLE Vision API generates relevant labels and that they share a high degree of semantic similarity with the corresponding scene text.

4.1 Evaluating the Proposed Prediction

To show that the proposed method is effective and useful, we calculate the measure before prediction, which provides the recognition rate for the output of text detection without the proposed global and local context scores. The results are compared with those of the proposed method after prediction. It is expected that the recognition rate of the proposed method after prediction should be higher due to the restoration of the missing information. To show the proposed combination (BERT and GOOGLE Vision API system) is better than the individual steps and other Natural Language Processing concepts, namely, edit distance, bigrams, we calculated the recognition rate for each step and the NLP steps, as reported in Table 4. For calculating the recognition rate for only BERT without the results of the GOOGLE Vision API, the proposed method only estimates the Local Context Score for predicting the missing words. Similarly, for calculating the recognition rate for the edit distance and bigram, the proposed method replaces the BERT approach with the edit distance and bigrams for missing word prediction. In Table 4, it is noted that the proposed method achieves the best precision, recall and F-measure compared to the individual steps before prediction. This shows that the GOOGLE Vision API system and the BERT contribute equally to achieve the best results. The reason is that the individual steps miss either local or global context information for accurately predicting the missing words.

However, when we compare the results of the proposed method before and after prediction, the results of after prediction are better than before, as reported in Table 4. The poor results of the proposed method with edit distance compared to the proposed combination is due to the failure in extracting semantic information between the candidate words. Similarly, the poor results of the proposed method with bigrams are because the bigram model extracts the context in one direction with the preceding word of the missing words. Since the occlusion can be at any position in the sentence, single direction information is not sufficient for complex situations. On the other hand, the proposed combination (BERT + GOOGLE Vision API) has the capability to extract context from both directions (left to right and right to left) in a sentence. Note that in the case of the bigram model, the ground truth of ICDAR 2015 is used for training the model.

Table 4. Analyzing the contribution of key steps of the proposed method for predicting missing text.

Recognition rate Before Prediction			After Prediction											
			Only BERT Without Google API			Edit distance			Bigram			Proposed Method (BERT+Google API)		
P	R	F	P	R	F	P	R	F	P	R	F	P	R	F
0.43	0.55	0.48	0.64	0.62	0.63	0.59	0.61	0.60	0.48	0.57	0.52	**0.67**	**0.64**	**0.65**

To illustrate that text detection does not have a substantial effect on recognizing broken words, we calculate the recognition rate for the output of each detection method listed in Table 5, including the PixelLink and E2E methods for before- and after-prediction. It is observed from Table 5 that the before-prediction results for all the text detection methods is are almost the same. In the same way, though there is a significant improvement in recognition results for after-prediction, the results of before prediction the respective text detection methods are almost the same. This shows that missing characters in the words do not affect text detection and recognigion performance of the different methods. It is also noted from Table 5 that the recognition rate of the PixelLink method is better than other existing methods for both before- and after-prediction. Therefore, one can conclude that the proposed combination of the GOOGLE Vision API labels, Text detection-recognition and NLP is the best for improving recognition performance for the images with occluded text.

Table 5. Recognition performance before- and after-prediction for the output of different text detection methods on our dataset

Different Text Detection Methods	Before Prediction (OCR)			After Prediction-Proposed Method		
	P	R	F	P	R	F
Zhou et al. EAST[22]	0.40	0.47	0.43	0.61	0.62	0.61
Shi et al. SegLink[23]	0.41	0.49	0.44	0.63	0.61	0.62
Liu et al. CTD[24]	0.39	0.44	0.41	0.62	0.60	0.61
Busta et la. E2E-Text Detection [5]	0.42	0.51	0.46	0.65	0.63	0.64
Deng et al. PixelLink [4]	0.43	0.55	0.48	**0.67**	**0.64**	**0.65**

The advantage of the proposed work is that if a part of the text in an image is missing due to other causes, but it is not necessarily due to occlusion, the proposed method can restore the missing part of the text as shown one example in Fig.8. In Fig.8, a few characters of the word “boundary” are missing. As mentioned earlier, for the proposed method it does not matter whether a part of the text is missing due to occlusion or some other causes. As a result, the proposed method predicts the missing part of the word “boundary” as shown in Fig.8. However, it is hard to generalize that the proposed method works for all the situations and restores the missing text correctly because the success depends on the lexicon size and labels given by the GOOGLE Vision API.

- The recognition results: “bo”, “ary and “Road”
- Labels using GOOGLE Vision API: “Nature”, “Signage”, “Tree”, “Grass”, “recreation”
- Candidate List for “bo” + “ary”: 'bigotry', 'boringly', 'boundary', 'bovary'
- The proposed method results: “boundary road”

Fig.8. Restoring the missing text without occlusion

- Recognition results before prediction “Raymo”+“vanheusen”
- The proposed method results: “ - ”

Fig.9. Limitation of the proposed method

In this work, the labels generated by the GOOGLE Vision API and generating candidate words are critical to the results. If the GOOGLE Vision API mislabels the input image, there are chances of predicting nothing as shown in Fig.9. For this image, the word is a noun which is not present in the lexicon and hence the proposed method predicts nothing “-“. Similarly, if the candidate generation step lists irrelevant choices for a recognized word, the proposed method fails as well. To overcome these limitations, a new method should be developed for labeling images and generating candidates. One possible solution is after getting labels from the GOOGLE Vision API, we can use geographical information about the location to verify or modify the label according to the situational context. Similarly, the same information can be used for verifying generated candidates. However, this is beyond the scope of the proposed work.

5 Conclusions and Future Work

In this work, we have proposed a new method for restoring missing text for replacing broken words due to occluded text in natural scene images. We explore the GOOGLE Vision API system for obtaining labels of the input images. For text in images including broken words, the proposed approach obtains possible candidate words with the help of the lexicons provided by the ICDAR 2015 ground truth. The proposed method finds

semantic similarity between labels and candidate words, which results in a global context score. Our method also employs the natural language processing technique called BERT for finding semantic similarity between the candidates, which results in a local context score. Furthermore, the proposed approach fuses global and local context scores to estimate ranking for each word in the list. The word that gets the highest rank is considered as a correct word or a predicted word for replacing broken words. Experimental results on our dataset show that the proposed method is effective and predicts missing parts well for different situations. As shown, the performance of the proposed approach depends on the results of the GOOGLE Vision API and the list of candidate words. Sometimes, for the images with large variations, the GOOGLE Vision API system may not generate correct labels. In this case, the performance of the proposed method degrades. Therefore, there is scope for future work and we can also extend the proposed work to other languages.

References

1. S. Roy, P. Shivakumara, U. Pal, T. Lu and G. H. Kumar, “Delaunay triangulation based text detection from multi-view images of natural scene”, Pattern Recognition Letters, 129, 2020, pp 92-100.
2. P. Shivakumara, R. Raghavendra, L. Qin, K. B. Raja, T. Lu and U. Pal, “A new multi-modal approach to bib number/text detection and recognition in Marathon images”, Pattern Recognition, Vol. 61, 2017, pp 479-491
3. M. Xue, P. Shivakumara, C Zhang, T. Lu, and U. Pal, “Curved text detection in blurred/non-blurred video/scene images”. Multimedia Tools and Applications, 2019, pp. 1-25.
4. D. Deng, H. Liu, X. Li and D. Cai, “PixelLink: Detecting scene text via instance segmentation”, In Proc. AAAI, 2018.
5. Y. Patel, M. Bušta, J. Matas, “E2E-MLT-an Unconstrained End-to-End Method for Multi-Language Scene Text”, arXiv preprint arXiv:1801.09919. 2018.
6. Google Cloud Vision API
7. J. Devlin, M. Wei. Chang, K. Lee and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. arXiv preprint arXiv:1810.04805, 2018
8. Z. Cheng, Y. Xu, F. Bai and Y. Niu, “AON: Towards arbitrarily-oriented text recognition”, In Proc. CVPR, pp 5571-5579, 2018.
9. S. Tian, X. C. Yin, Y. Su and H. W. Hao, “A unified framework for tracking based text detection and recognition from web videos”, IEEE Trans. PAMI, 40, pp 542-554, 2018.
10. C. Luo, L. Jin and Z. Sun, “MORAN: A multi-object rectified attention network for scene text recognition”, Pattern Recognition, 90, pp 109-118, 2019.
11. K. S. Raghunandan, P. Shivakumara, S. Roy, G. H. Kumar, U. Pal and T. Lu, “Multi-script-oriented text detection and recognition in video/scene/born digital images”, IEEE Trans. CSVT, 29, pp 1145-1162, 2019.
12. X. Qi, Y. Chen, R. Xiao, C. G. Li, Q. Zou and S. Cui, “A novel joint character categorization and localization approach for character level scene text recognition”, In Proc. ICDARW, pp 83-90, 2019.
13. B. Shi, M. Yang, X. Wang, P. Luy, C. Yao and X. Bai, “ASTER: An attentional scene text recognizer with flexible rectification”, IEEE Trans. PAMI, 41, pp 2035-2048, 2019.
14. X. Rong, C. Yi, and Y. Tian, “Unambiguous scene text segmentation with referring expression comprehension”, IEEE Trans. IP, 29, pp 591-601, 2020.

15. M. Villamizar, O. Canevert and J. M. Odobez, “Multi-scale sequential network for semantic text segmentation and localization”, Pattern Recognition Letters, 129, pp 63-69, 2020.
16. W. Feng, W. He, F. Yin, X. Y. Zhang and C. L. Liu, “TextDragon: An end-to-end framework for arbitrary shaped text spotting”, In Proc. ICCV, pp 9076-9085, 2019.
17. C. W. Lee, K. Jung and H. J. Kim, “Automatic text detection and removal in video sequences”, Pattern Recognition Letters, 24, pp 2607-2623, 2003.
18. Q. Ye, J. Jiao, J. Huang and H. Yu, “Text detection and restoration in natural scene image”, J. Vis. Commun, Image. R, 18, pp 504-513, 2007.
19. T. H. Tsai and C. L. Fang, “Text-video completion using structure repair and texture propagation”, IEEE Trans. MM, 13, pp 29-39, 2011.
20. A. Mosleh, N. Bouguila and A. B. Hamaza, “Automatic inpainting scheme for video text detection and removal”, IEEE Trans. IP, 22, pp 4460-4472, 2013.
21. S. Zhang, Y. Liu, L. Jin, Y. Huang and S. Lai, “EnsNet: Ensconce Text in the Wild”, In Proc. AAAI, 2019.
22. L. Wu, C. Zhang, J. Liu, J. Han, J. Liu, E. Ding and X. Bai, “Editing Text in the Wild”, In Proc. ACM MM, pp 1500-1508, 2019.
23. X. Tong and D. A. Evans, “A Statistical Approach to Automatic OCR Error Correction in Context”. In Proc. WVLC, pp 88-100, 1996.
24. X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, and J. Liang, “East: an efficient and accurate scene text detector,” in Proc. CVPR, pp. 2642–2651, 2017.
25. B. Shi, X. Bai, and S. Belongie, “Detecting oriented text in natural images by linking segments”. In Proc. CVPR, pp. 3482–3490, 2017.
26. Y. Liu, L. Jin, S. Zhang and S. Zhang, “Detecting curve text in the wild: New dataset and new solution”, arXiv:1712.02170, 2017.