symmetry

MDPI

Article

A Convolutional Recurrent Neural-Network-Based Machine Learning for Scene Text Recognition Application

Yiyi Liu, Yuxin Wang and Hongjian Shi *

Guangdong Provincial Key Laboratory of Interdisciplinary Research and Application for Data Science, Beijing Normal University—Hong Kong Baptist University United International College, Zhuhai 519087, China
* Correspondence: shihj@uic.edu.cn

Abstract: Optical character recognition (OCR) is the process of acquiring text and layout information through analysis and recognition of text data image files. It is also a process to identify the geometric location and orientation of the texts and their symmetrical behavior. It usually consists of two steps: text detection and text recognition. Scene text recognition is a subfield of OCR that focuses on processing text in natural scenes, such as streets, billboards, license plates, etc. Unlike traditional document category photographs, it is a challenging task to use computer technology to locate and read text information in natural scenes. Imaging sequence recognition is a longstanding subject of research in the field of computer vision. Great progress has been made in this field; however, most models struggled to recognize text in images of complex scenes with high accuracy. This paper proposes a new pattern of text recognition based on the convolutional recurrent neural network (CRNN) as a solution to address this issue. It combines real-time scene text detection with differentiable binarization (DBNet) for text detection and segmentation, text direction classifier, and the Retinex algorithm for image enhancement. To evaluate the effectiveness of the proposed method, we performed experimental analysis of the proposed algorithm, and carried out simulation on complex scene image data based on existing literature data and also on several real datasets designed for a variety of nonstationary environments. Experimental results demonstrated that our proposed model performed better than the baseline methods on three benchmark datasets and achieved on-par performance with other approaches on existing datasets. This model can solve the problem that CRNN cannot identify text in complex and multi-oriented text scenes. Furthermore, it outperforms the original CRNN model with higher accuracy across a wider variety of application scenarios.

Keywords: CRNN; DBNet; OCR; Retinex

check for updates

Citation: Liu, Y.; Wang, Y.; Shi, H. A Convolutional Recurrent Neural-Network-Based Machine Learning for Scene Text Recognition Application. *Symmetry* **2023**, *15*, 849. https://doi.org/10.3390/sym15040849

Academic Editors: João Ruivo Paulo, Cristina P. Santos, Gabriel Pires and Alexander Shelupanov

Received: 9 January 2023
Revised: 17 March 2023
Accepted: 30 March 2023
Published: 2 April 2023

1. Introduction

Optical character recognition (OCR) refers to the use of a machine to convert manuscript or printed text in an image into a format that a computer can directly process. OCR involves recognizing the text on geometric, orientation, location, and symmetric information such as horizontal, vertical, circular, or elliptic symmetry in detection. As an essential branch of computer vision, the typical application of OCR is to process information input using image text recognition. Meanwhile, given that text and symbols of an image contain rich semantic information, the extraction and analysis of OCR-based textual details can help the machine better understand the image.

1.1. Background

The development of OCR performance over the past few years, made possible by artificial intelligence technology, has solidly supported the more complicated OCR application scenarios brought on by industrial digitalization. While this is happening, more diversified service providers for mobile phones, electronics, and cloud services are helping OCR become more popular and spread into more social production and daily life spheres.

Text recognition has two parts: text detection and text recognition. Text detection is the process of locating text regions in an image. This can be done by methods such as edge detection, connected component analysis, sliding windows, etc. Text recognition is the process of identifying each character or word in each text region. This can be done by methods such as template matching, feature extraction, neural networks, etc. The development stages of text recognition can be roughly divided into the following.

- Early stage (the 1950s–1970s): mainly used for machine-printed text recognition, using hardware devices and simple algorithms, with limited effects, slow speed, and low accuracy.
- Middle stage (the 1980s–1990s): started to be used for handwritten and natural scene text recognition, using software systems and complex algorithms, with improved effects, fast speed, and high accuracy.
- Recent stage (2000s–present): widely used for various types and languages of text recognition, using deep learning and artificial intelligence technologies, with significant improvement in effects, fast speed, and high accuracy.

One of the critical developments for OCR technology will be creating an integrated end-to-end network and training in text detection and recognition. End-to-end network architecture can not only reduce double computation but also enhance feature quality and improve task performance. At the same time, many OCR applications must be operated on mobile terminal devices with limited resources and most of the current mobile terminal OCR algorithms compromise algorithm accuracy at running speed. The development of an efficiency-optimized lightweight OCR model for mobile devices will be an indispensable area of focus in the following years.

Writing is significant from the perspective of the entire culture because it is not a natural creation but a singular human creation and a carrier of high-level semantic information. The written word is inextricably linked to human civilization and serves as an essential medium for transmitting ideas, disseminating knowledge, and learning new things. Consequently, it is crucial to offer multi-scene, highly accurate text detection and identification services.

Word recognition can speed up text processing when used with massive data. There must be a comprehensive range of applications for text recognition. In recent years, using CNNs to locate and segment text regions in images, and split them into words or characters, is an important step in text detection. The purpose of text detection is to find regions containing text from complex backgrounds and divide them into smaller units, such as words or characters, for subsequent text recognition. There are several methods for using CNNs to locate and segment text regions, including the sliding-window-based method, fully convolutional network (FCN) based method, and regression network (RNN).

The limitations of the convolutional recurrent neural network (CRNN) [1] cause recognition with low accuracy for short texts with significant deformations, such as art terms or texts describing scenes in the natural world. To increase the precision of scene text recognition, this study developed a novel framework for text recognition based on CRNN. This design preprocesses the image before text recognition by combining many word processing methods. It can divide the text area more precisely, enhance the image, and ultimately achieve the accuracy needed for multi-application scene text recognition.

The primary application of CRNN is the solution of image-based sequence recognition issues, particularly those requiring scene text recognition. Its main benefits include the ability to recognize text sequences of any length and the ability to do end-to-end training on sample data without character segmentation. In addition, it uses less storage space and has fewer parameters than the usual DCNN model [2]. It exhibits exceptional performance in both thesaurus-free and thesaurus-based scene text recognition tests, and is not constrained to any particular thesaurus. CRNN has to scale vertically to a set length to recognize sequences of arbitrary length, but character segmentation and horizontal scaling processes are unnecessary. As a result, the recognition effect of CRNN for this type of text is low and it is not sensitive to multi-oriented text. Because CRNN does more than text recognition, it is ineffective in photos of complicated scenes.

1.2. Purpose

This paper proposes an innovative model based on the above problems by adding different schemes to optimize the results of the existing algorithms and models. To solve the problem of recognizing complex scenes, a pretext detection network—DBNet [3]—is added to detect and segment hidden text in complex scenes. In addition, an image enhancement algorithm—the Retinex [4] algorithm—is added between text detection and recognition to enhance feature points. To solve the problem of recognizing multi-oriented text, the text direction classification is used to classify the rotation angle of the image. The model flow used in this paper is shown in Figure 1.

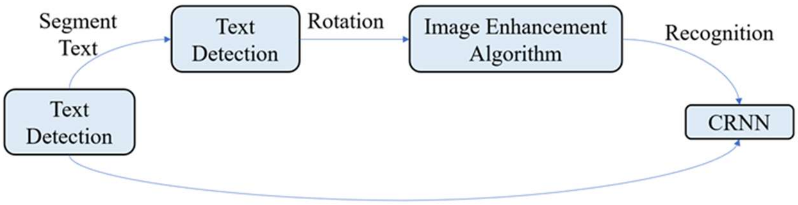


Figure 1. Flow chart of the proposed model.

In summary, this paper proposes a new model by fusing many models to improve the original CRNN. It uses an image enhancement technique, CRNN, a text direction classifier, and a relatively sophisticated text detection network called DBNet. The proposed model resolves the multi-oriented text recognition problem more effectively.

2. Related Work

Text recognition is divided into two specific steps: detection of text and recognition of text, especially text detection, which is a prerequisite for recognition. Several of today's popular text detection techniques are described below. Text detection is not a simple task; in particular, text detection in complex scenes is very challenging.

2.1. Context-Aware STR

Context-aware STR stands for context-aware scene text recognition, which is the task of recognizing text in natural scenes such as street signs, billboards, product labels, etc. Context-aware STR methods typically use semantics learned from data to aid in recognition. One paper that proposes a novel method for context-aware STR is scene text recognition with permuted autoregressive sequence models. This paper introduces a permuted autoregressive sequence model that can handle arbitrary orientations and languages of scene text without relying on external LMs. The main idea is to present a novel Urdu numeral dataset and a custom convolutional neural network (CNN) model for recognizing and classifying handwritten Urdu numerals. The paper also compares the performance of different CNN variants and classifiers on the dataset.

The experimental results show that the proposed custom CNN model with Softmax activation function achieves an accuracy of 99.6% on the test set, which is higher than other CNN variants such as LeNet-5, AlexNet, VGG-16, ResNet-50, and Inception-v3. The paper also shows that using a support vector machine (SVM) classifier instead of the Softmax activation function improves the accuracy by 0.2%. The paper claims that their proposed method outperforms existing methods for handwritten Urdu numeral recognition and classification.

2.2. Seglink

A CVPR2017 spotlight paper [5] introduces a detection algorithm that can detect text from any angle named SegLink, which incorporates the idea of CTPN small-scale candidate frames. This paper includes the concept of CTPN small-scale candidate frames and single shot multibox detector (SSD) [6] to achieve the effect of state-of-the-art text detection in natural scenes at that time.

SegLink offers the crucial idea of segmentation, which can be translated as a character or any other portion of a line of text. A whole line of text has many line segments, each joined and combined by links, as seen in Figure 2 below, where the yellow box symbolizes a line segment (green lines). Additionally, the concept of text detection for the segment is similar to CTPN, where the frame first detects a part of a text line and then links it with other portions to create a full-text line.

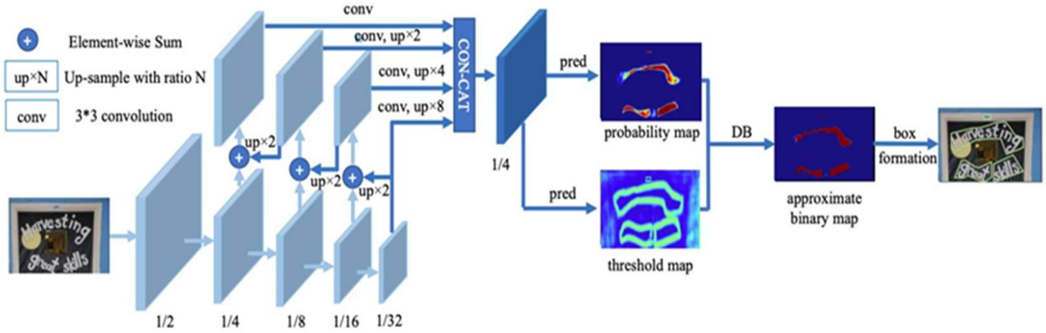


Figure 2. Structure of DBNet—DBNet is a novel network architecture for real-time scene text detection with differentiable binarization.

2.3. EAST

A CVPR2017 paper [7] proposes that the EAST model can elegantly and concisely complete multi-angle text detection.

The feature extraction layer, the feature fusion layer, and the output layer are the three main components that make up the EAST network. First, Backbone has given PVANet [8] permission to do feature extraction in the feature extraction layer. The collected features are then applied to the convolution layer. The number of convolution cores increases progressively or is double that of the preceding layer, and the size of the next convolution layer decreases incrementally or is cut in half. Feature maps of various levels are extracted to make feature maps of various scales to address the issue that the scale of text lines fluctuates noticeably. Large-sized layers cannot necessarily suggest small-sized text lines, but small-sized layers can anticipate large-sized text lines.

The extracted features are additionally combined at the feature merge layer. The U-net [9] approach is used in this merging rule. The top features in the feature extraction network are merged from top to bottom following the corresponding regulations. There are five primary components to the network output layer's final output.

3. Materials and Methods

The primary application of CRNN is the solution of image-based sequence recognition issues, particularly those requiring scene text recognition. Its main benefits include the ability to recognize text sequences of any length and the ability to do end-to-end training on sample data without character segmentation. In general, this model synthesizes an image enhancement technique, CRNN, a text direction classifier, and a relatively sophisticated text detection network called DBNet, which are described below. These methods resolve the multi-oriented text recognition problem more effectively.

3.1. DBNet

DBNet is a novel network architecture for real-time scene text detection with differentiable binarization. It aims to solve the problem of text localization and segmentation in natural images with complex backgrounds and various text shapes.

The network consists of three main components, including a segmentation network, a binarization module, and a threshold map learning module.

- The segmentation network is based on FPN and ResNet, which can output a probability map of text regions. The probability map indicates the likelihood of each pixel belonging to text or background.
- The binarization module is a differentiable step function that can convert the probability map into a binary map. The binary map has only two values: 0 for the background and 1 for the text.
- The threshold map learning module is a convolutional layer that can predict an adaptive threshold for each pixel in the probability map. The threshold map determines how to binarize the probability map by comparing it with the threshold value at each pixel location.

The main idea of DBNet is to insert the binarization operation into the segmentation network and jointly optimize them so that the network can learn to separate foreground and background pixels more effectively. The binarization threshold is learned by minimizing the IoU loss between the predicted binary map and the ground truth binary map. The structure of DBNet is shown in Figure 2.

DBNet is an innovative and effective method for scene text detection that leverages differentiable binarization to improve both accuracy and speed. It can handle various text shapes, orientations, scales, and languages in natural images.

3.1.1. Differentiable Binarization

The main innovation in the differentiable binarization module, which is essential to DBNet, is to transform the binarization process into one that is optimizable and introduces adaptive binarization. The DB module's formula is

$$B_{ij} = \frac{1}{1 + e^{-\alpha(P_{ij} - t_{ij})}}, \tag{1}$$

where α denotes the method coefficient and, in this study, α is set to 50. P_{ij} and t_{ij} denote the probability graph and threshold graph respectively. The function is closer to a sharp increase. The greater α is, the narrower the function is. The approximate binarization function operates similarly to the conventional binarization function but, because it is differentiable, it can be improved during training by the segmentation network. In addition to being able to separate text from the background, differentiable binarization with an adaptive threshold may isolate groups of closely related text instances.

3.1.2. Adaptive Threshold

Figure 3 illustrates three distinct threshold maps to highlight how employing a border-type threshold map benefits the outcomes. It clearly shows that the projected threshold graph can differentiate several areas without text, text boundary, and text core, even without the supervision of the threshold graph. The boundary obtained by the threshold graph is more precise than that in Figure 3d, which depicts the loss of the image with the addition of the word boundary.

3.1.3. Label Generation

The label for the approximate binary map and the probability map are the same. This label generation uses the Vatti pruning technique to condense the text with the PSENet kernel concept [10]. We reduce the size of the probability graph label, which is the original standard text box G_t to G_s. The shrink D's deviation is determined as follows:

$$D = \frac{A\left(1 - r^2\right)}{L} \tag{2}$$

where A is the polygon's surface area, L is its perimeter, and r is its contraction ratio. K is set to 0.4.

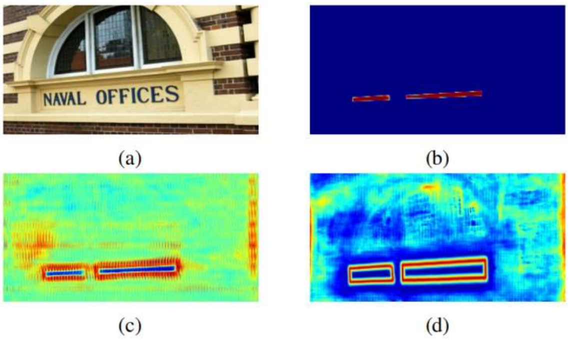


Figure 3. The threshold map with/without supervision. (**a**) Original image. (**b**) Probability map. (**c**) Unsupervised threshold map. (**d**) Supervised threshold map [3].

Labels can be constructed for the threshold graph using a similar procedure. First, with the same offset D, the text polygon G_t is extended to G_d. The text area's border is defined as the space between G_t and G_d, from which the label of the threshold graph can be constructed by figuring out how far G_t is from the nearest line segment. The outcome of the label generation is shown below in Figure 4.

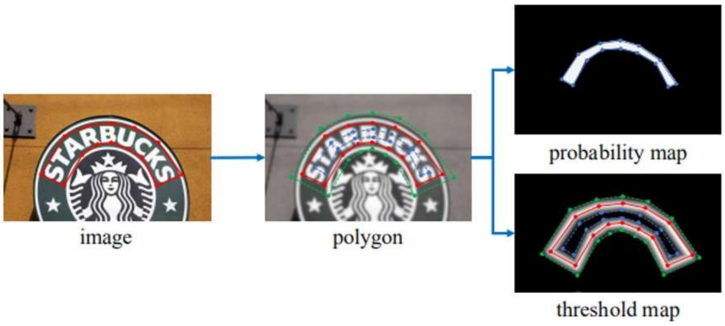


Figure 4. Example result of label generation for circular symmetric shape containing text. The annotation of the text polygon is visualized in red lines. The shrunk and dilated polygon are displayed in blue and green lines, respectively [3].

3.2. Text Direction Classification

Paddle text direction classifier is a module that is added between the text detection and recognition modules to deal with text in different directions. It uses a convolutional neural network (CNN) with four fully connected layers to extract features from the input image and classify them into four categories. It chooses the direction with the highest probability score as the final output. The text direction classifier network structure is as follows.

- A CNN backbone with 16 convolutional layers and 4 max-pooling layers;
- A global average pooling layer;
- Four fully connected layers with 256, 64, 16, and 4 neurons respectively;
- A SoftMax layer for outputting probability scores.

When the image is not 0 degrees, degree classification is utilized. In this case, the text lines found in the image need to be fixed. After text detection, a text line image is obtained. This image is then affine transformed and passed to the recognition model. This study requires training a two-class (0° and 180°) classification model because only 0° and 180° angle classification is necessary for the text. The flow chart and the definition of text direction classification are shown in Figures 5 and 6, respectively.

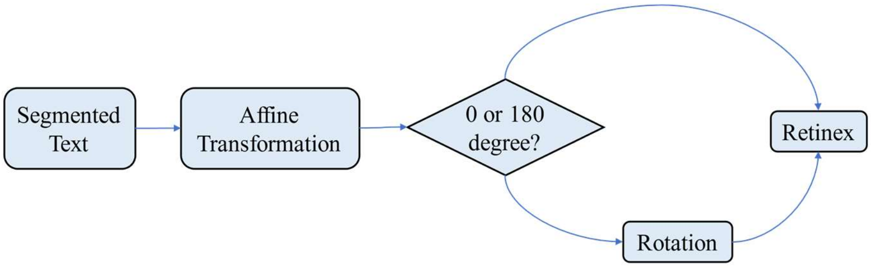


Figure 5. The flow chart of text direction classification.

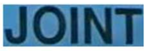
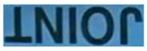
0 degree	180 degree
	

Figure 6. The definition of vertically symmetric text direction classification.

3.3. Retinex Algorithm

Retinex [4] is a widely used image enhancement technology developed through research and analysis. This theory's fundamental tenet is that, as opposed to the absolute value of reflected light intensity, an object's color is determined by its capacity to reflect long-wave (red), medium-wave (green), and short-wave (blue) light. Retinex is based on the consistency of color sensitivity since an object's color is unaffected by the non-uniformity of light and has consistency (color constancy). Retinex theory's fundamental premise is that the original image S is the result of the light image L and the reflectance image R, which may be written as the formula below.

$$S(x,y) = R(x,y) \times L(x,y), \tag{3}$$

3.3.1. Single Scale Retinex (SSR)

SSR is the most basic and most straightforward Retinex algorithm and it also gives the general framework of the Retinex algorithm in a broad sense.

To deconstruct R, remove the impact of uneven illumination, and enhance the visual effect of the image, image enhancement based on Retinex aims to estimate illumination L from the original image S. The image is typically moved to the logarithmic domain during processing, that is

$$s = \log S(x,y), \tag{4}$$

$$l = \log L(x,y), \tag{5}$$

$$r = \log R(x,y), \tag{6}$$

where $r(x,y)$ is the output image.

The fundamental SSR formula can be written as follows after conversion.

$$F(x,y) = \lambda e^{\frac{-(x^2+y^2)}{c^2}}, \tag{7}$$

where $F(x, y)$ is the center-surround function.

3.3.2. Multi-Scale Retinex (MSR)

The multi-scale Retinex algorithm (MSR) is a Retinex algorithm developed from SSR. It employs various sigma values before weighing the outcomes. Its fundamental syntax is as follows:

$$R_{MSR}(x,y,\sigma) = \sum_{k=1}^{n} w_k R_{MSR}(x,y,\sigma_k), \tag{8}$$

where n is the number of scales, σ is the vector of Gaussian fuzzy coefficients, and w_k is the weight associated with the kth scale, where $w_1 + w_2 + \ldots + w_n = 1$.

Figure 7 below depicts the steps involved in implementing MSR, which is used in this paper for image enhancement.

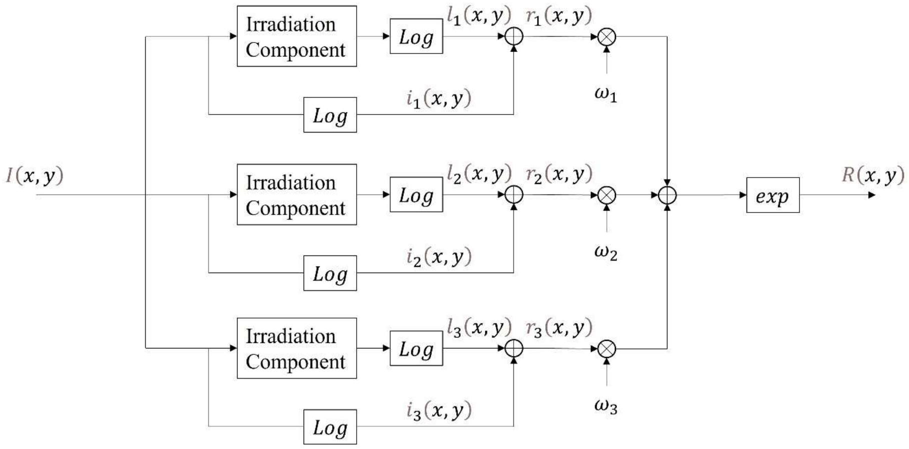


Figure 7. Steps of MSR.

Step 1: Input the original image $I(x,y)$, according to the gray value, set it into several scale levels, and separate the three-color components.
Step 2: Construct the Gaussian surround functions $G_k(x, y)$ with different scale parameters.
Step 3: The three channels, B, G, and R, are convolved by the Gaussian surround function. The illuminance component is obtained by weighted averaging.

$$L(x,y) = \sum_{k=1}^{N} w_k(I_k(x,y) \times G_k(x,y)) \; where \; \sum_{k=1}^{N} w_k = 1. \quad (9)$$

Step 4: Logarithm is taken and the original image is subtracted from the light component:

$$logR_i(x,y) = \sum_{k=1}^{N} w_k(log(I_i(x,y)) - log(I_i(x,y) \times G_k(x,y))) \quad (10)$$

Step 5: Convert the logarithm domain to the real domain $R(x,y)$.
Step 6: The output reflection component is used as the resulting image.

4. Experiments

We test our proposed algorithm on complex scene image data based on existing literature data to see how effective our method is. We use several real datasets that mimic different kinds of non-stationary environments. Our model outperforms the baseline methods on three benchmark datasets and matches other approaches on the existing dataset according to experimental results.

4.1. Datasets

Four datasets are used in the experiments as shown in Table 1. All three of the last come from Kaggle. The following summarizes the data set's basic details.

- ICDAR2015 [11] This consists of 1000 training and 500 test charts, and is the official dataset used in the Scene Text Detection Competition held by ICDAR in 2015.
- TextOCR [12]: TextVQA images offer roughly 1 million high-quality word annotations, enabling end-to-end reasoning for jobs down the line, including visual question answers or image captions.

- Total-Text [13]: Its primary goals are to complement the text's curvilinear orientation and to offer the scene text community a fresh line of inquiry.
- TOTAL-TEXT [14]: This is an English curvilinear text data set at the word level. There are 1555 images in total and the text is oriented in more than three different ways, including landscape, multi-oriented, and curved.

Table 1. Details of datasets.

Dataset	**Source**	**Images**	**Language**	**Shape**
TextOCR	TextVQA images	28,408	English	Arbitrary
ICDAR2015	ICDAR Robust Reading Competition Challenge 4	1500	English	Horizontal
Total-Text	Natural scene images	1855	English	Horizontal, multi-oriented, and curved
TOTAL-TEXT	Natural scene images	1555	English and Chinese	Curved and perspective

4.2. Testing Metrics

4.2.1. Confidence

OCR identification findings might not always be exact in real-world situations. Errors appear in the converted text that is readable when OCR recognizes results with less than 100% accuracy. As a result, a technique for assessing the accuracy of OCR identification results is required.

(1) After feeding the pre-trained CRNN model the images that need to be recognized, multiple output results are acquired. The results of the logistic regression matrix and character recognition are included in each batch of output results.

(2) Verification is performed to determine whether there are more valid character recognition results than the predetermined number in each set of output results. The same character recognition result appears in several character recognition results, making it valid:

   a. The confidence level of the OCR recognition results is set to zero if the number of valid character recognition results is less than the predetermined number.

   b. If there are more effective character recognition results than the predetermined number, the effective logistic regression matrix can be normalized to determine the probability value associated with each character in the effective character recognition results. The confidence in the OCR recognition results is shown to have the lowest probability value among the collected values. The one with the output result and the effective character recognition result in the same group is an effective logistic regression matrix.

Figure 8 illustrates the process for determination of how confident one can be in OCR identification results.

4.2.2. Accuracy

Detection accuracy is the ratio of correct detection to all detection as follows:

$$accuracy = \frac{correct}{all}, \tag{11}$$

where *correct* denotes the total number of predictions in which confidence is larger than the text threshold and *all* denotes the total number of labels.

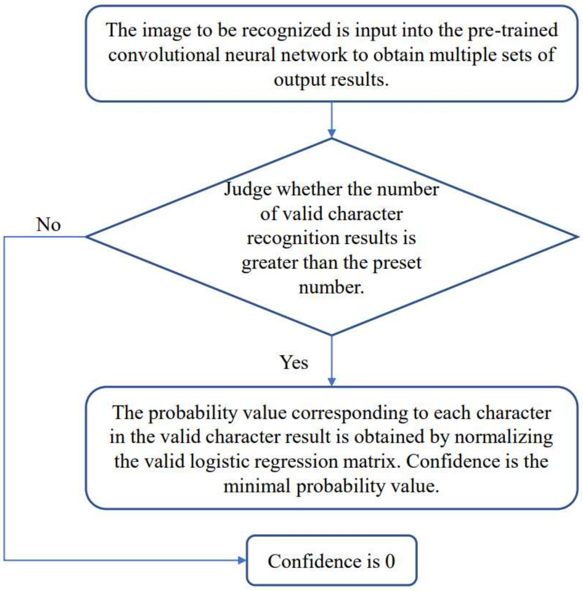

The image to be recognized is input into the pre-trained convolutional neural network to obtain multiple sets of output results.
Judge whether the number of valid character recognition results is greater than the preset number.
No
Yes
The probability value corresponding to each character in the valid character result is obtained by normalizing the valid logistic regression matrix. Confidence is the minimal probability value.
Confidence is 0

Figure 8. Steps of calculating confidence.

4.2.3. CTC Loss Function

The CTC loss function in CRNN is the transcription layer responsible for translating the RNN's predictions for each feature vector into a label sequence. Based on each frame prediction, transcription is defined as finding the tag sequence with the highest likelihood combination. The challenge with end-to-end OCR recognition is dealing with variable-length sequence alignment. OCR can be represented as a time-series-dependent text-image issue, and the CTC loss function can train the CNN and RNN end-to-end.

The sequence output by the RNN must now be translated into the final recognition result. There will be much redundant information when the RNN does time series classification. For example, de-redundancy procedures are required when a letter is recognized twice in a row, as shown in Figure 9.

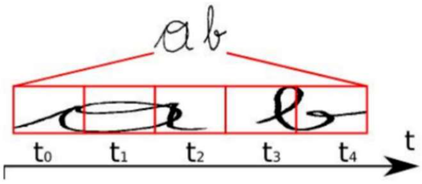


Figure 9. Example of sequence merge mechanism.

For example, RNN has five-time steps when recognizing the above text. Ideally, t_0, t_1, and t_2 should be mapped to "a". Then t_3, and t_4 should be mapped to "b", and then these characters should be mapped to a sequence to get "aaabb" and then combine the successive repeated letters into one to get "ab."

To indicate blank, this method uses the "-" sign. Insert one "-" between the repeated characters in the text labels when the RNN outputs the sequence. If the output sequence is "bbooo-ookk", for example, it will eventually be mapped to "book". If there are blank characters between identical characters, they will not be merged. Decoding is the process of removing consecutive repeated characters from a character sequence, followed by removing all "-" characters from the path; a neural network accomplishes encoding. It effectively solves the problem of repeated characters by adding the blank method.

For example, "aa-b", "aabb", and "-abb" all express the exact text ("ab") in a different alignment from the image. A text label can be found in one or more pathways in general.

Hence, for the RNN, given the input probability distribution matrix $y = \{y_1, y_2, \ldots, y_T\}$, T is the sequence length and, finally, the total probability of mapping to the label text l is

$$p(l|y) = \sum_{\pi:B(\pi)=1} P(\pi|y), \tag{12}$$

The product of the scores of the associated character at each time step determines the likelihood of each path. It is necessary to train the network to optimize this probability value. The CTC loss function is the same as the conventional classification's function: a negative maximum likelihood function of probability. The logarithm of the likelihood function is used for ease of computation.

The prior neural network can be back-propagated by computing the loss function and the neural network's parameters are changed according to the optimizer used to discover the character corresponding to the most likely pixel area. Because of this mapping transformation and the sum of all feasible path probabilities, CTC does not need the accurate segmentations of the original input character sequence.

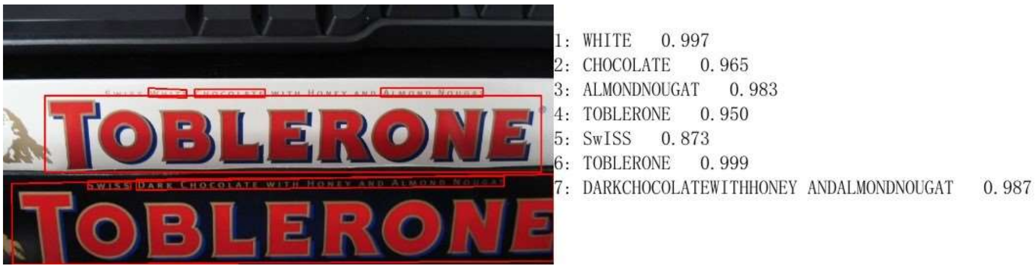
4.3. Implementation Platforms

- Training GPU: Nvidia RTX 2080Ti;
- Operating system: Windows 10;
- Programming platform: Python 3.8 + PaddleOCR.

5. Results and Discussion

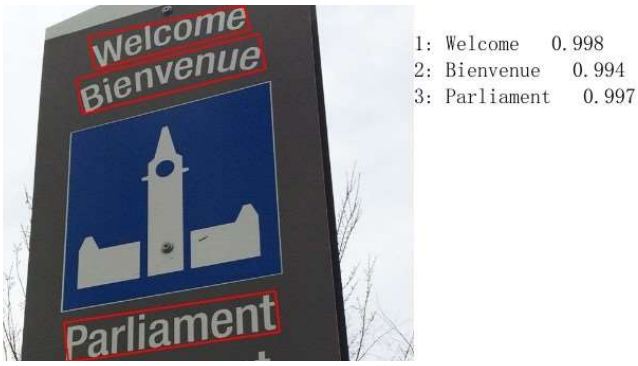
5.1. Results

There are several text recognition results of different complex scenes shown below in Figures 10–14.



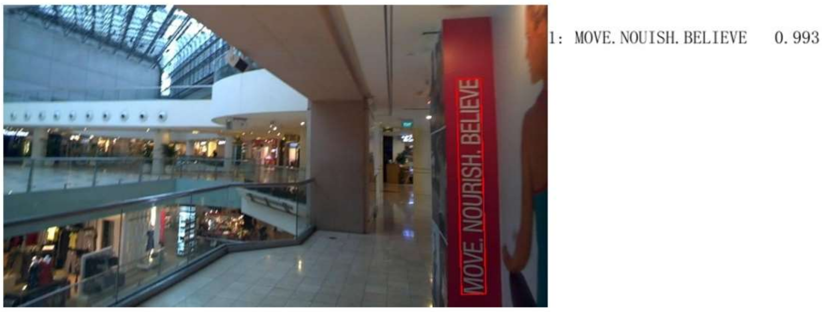
```
1: WHITE    0.997
2: CHOCOLATE    0.965
3: ALMONDNOUGAT    0.983
4: TOBLERONE    0.950
5: SwISS    0.873
6: TOBLERONE    0.999
7: DARKCHOCOLATEWITHHONEY ANDALMONDNOUGAT    0.987
```

Figure 10. Results of vertical symmetric text recognition.



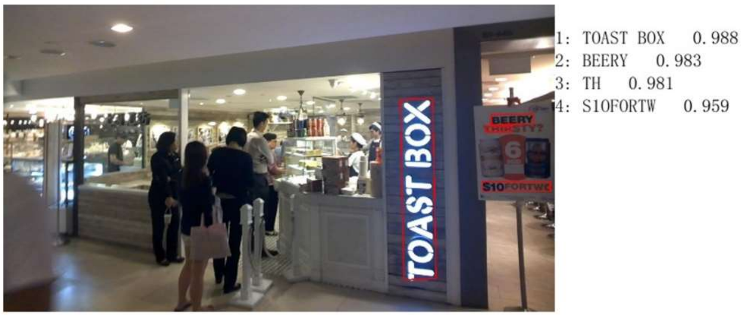
```
1: Welcome    0.998
2: Bienvenue    0.994
3: Parliament    0.997
```

Figure 11. Results of non-vertical text recognition.



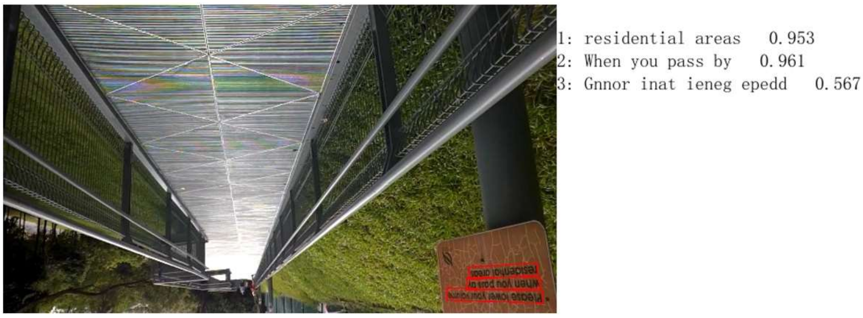
```
1: MOVE.NOUISH.BELIEVE    0.993
```

Figure 12. Results of multi-oriented text recognition.



```
1: TOAST BOX    0.988
2: BEERY    0.983
3: TH    0.981
4: S1OFORTW    0.959
```

Figure 13. Results of vertical text recognition.



```
1: residential areas    0.953
2: When you pass by    0.961
3: Gnnor inat ieneg epedd    0.567
```

Figure 14. Results of 180°-rotated text recognition.

5.2. Loss Values

During the training of 20 epochs, the purple line decreases faster and becomes closer to the blue line. The improved model is more sensitive to multi-orientation images, which is close to the loss value of horizontal orientation images. The CTC loss values are shown in Figure 15.

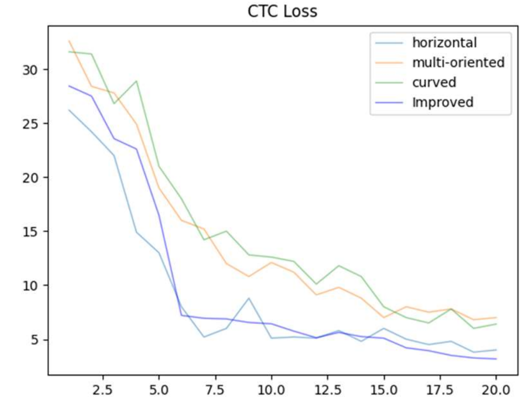


Figure 15. CTC loss values of the improved model.

5.3. Accuracy

Firstly, we compared experimental results using SegLink and EAST, which could recognize multi-oriented text. Table 2 shows how our method is better than other methods. Some results are from the online leaderboard. Our method beats the others by a lot. It has 10.7% higher precision than the second best.

Table 2. Comparison of the average accuracies using different models with the same datasets using SegLink [5] that is most accurate.

Method	Precision	Recall	F-measure
HUST_MCLAB	47.5	34.8	40.2
NJU_Text	72.7	35.8	48.0
StradVision-2	**77.5**	36.7	49.8
MCLAB_FCN [15]	70.8	43.0	53.6
CTPN [16]	51.6	74.2	60.9
Megvii-Image++	72.4	57.0	63.8
Yao et al. [17]	72.3	58.7	64.8
SegLink	**73.1**	**76.8**	**75.0**

EAST was evaluated on ICDAR 2015 Challenge 4, which is a benchmark dataset for text detection in natural scenes. We compared this method with other state-of-the-art methods using the F-score metric shown in Table 3. When we use the original scale of the images as input to our network, this method achieved an impressive F-score of 0.7820. However, when we tested our method at multiple scales using the same network architecture and parameters, we could further improve performance and reached an F-score of 0.8072. This showed that our proposed method is robust to different scales and can handle challenging text detection scenarios.

Table 3. Results on ICDAR 2015 Challenge 4 Incidental Scene Text Localization task. MS means multi-scale testing [7]. * The bold number indicates the best one related to others in different metrics.

Algorithm	Recall	Precision	F-score
Ours + PVANET2x RBOX MS *	**0.7833**	0.8327	**0.8072**
Ours + PVANET2x RBOX	0.7347	**0.8357**	**0.7820**
Ours + PVANET2x QUAD	0.7419	0.8018	0.7707
Ours + VGG16 RBOX	0.7275	0.8046	0.7641
Ours + PVANET RBOX	0.7135	0.8063	0.7571
Ours + PVANET QUAD	0.6856	0.8119	0.7401
Ours + VGG16 QUAD	0.6895	0.7987	0.7401
Yao et al. [17]	0.5869	0.7226	0.6477
Tian et al. [16]	0.5156	0.7422	0.6085
Zhang et al. [15]	0.4309	0.7081	0.5358
StradVison2 [18]	0.3674	0.7746	0.4984
StradVision1 [18]	0.4627	0.5339	0.4957
NJU [19]	0.3625	0.7044	0.4787
AJOU [20]	0.4694	0.4726	0.4710
Deep2Text-MO [21,22]	0.3211	0.4959	0.3898
CNN MSER [19]	0.3442	0.3471	0.3457

The original CRNN almost cannot recognize text in complex and multi-oriented scenes, and its accuracy under these circumstances is almost 0%. After the first step, we can add DBnet [3] to segment the text area, which improved the accuracy up to 75.4 and 60.7% respectively. This is because DBNet [3] can adaptively predict the threshold for each pixel, thus achieving a more accurate binarization operation, distinguishing foreground and background pixels. It introduces a differentiable binarization module, which enables the network to be trained and optimized end-to-end. Then, the text direction analysis model was put into our model. Great progress has been seen in terms of accuracy, exceeding 80%, because it can handle multi-oriented text. Eventually, our improved model, which is based on the above methods and the Retinex algorithm [4], can recognize complex scenes and has an accuracy up to 84.8%. In addition, it can recognize multi-oriented text scenes and the accuracy is up to 82.1%, which mimics the human vision system to improve the quality of images. It can achieve a balance among dynamic range compression, edge enhancement, and color constancy. Moreover, it can enhance the contrast and detail information of images. The accuracies tested in different datasets are shown in Figure 16.

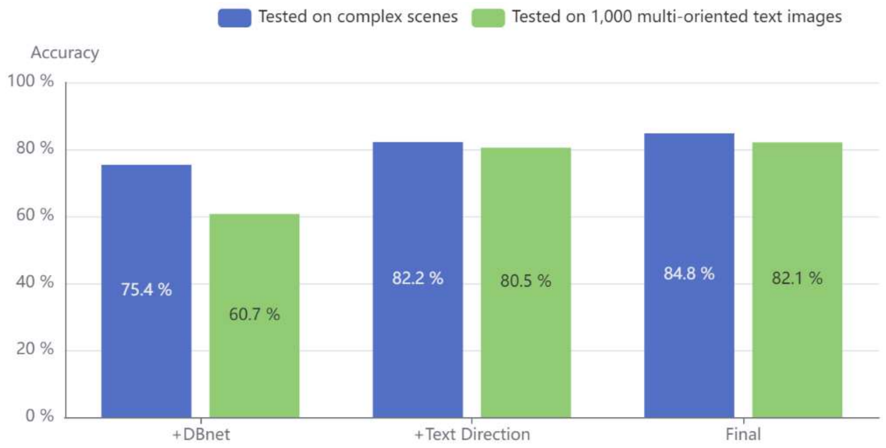


Figure 16. Accuracies after adding DBNet and text direction operation.

Table 4 shows the comparison of the average accuracies of recognizing text using different models with the same datasets. It can be found that our proposed model is more accurate for multi-oriented text recognition than other models.

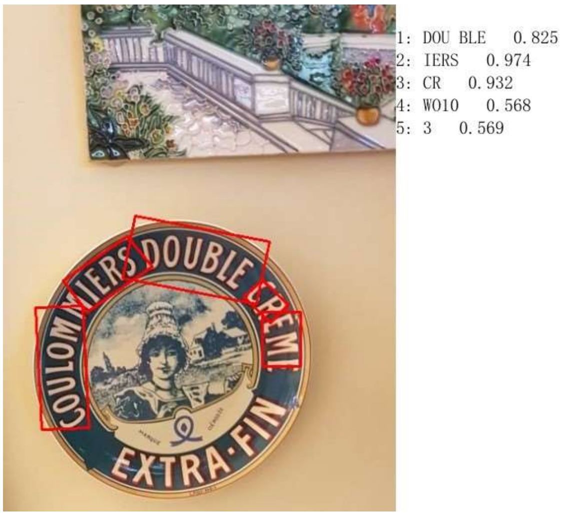
Table 4. Comparison of the average accuracies using different models with the same datasets.

	CRNN	SegLink	EAST	Proposed Model
Complex Scenes	0.03	74.69	78.33	84.80
Multi-oriented Text Scenes	0.01	73.10	73.47	82.10

5.4. Weakness

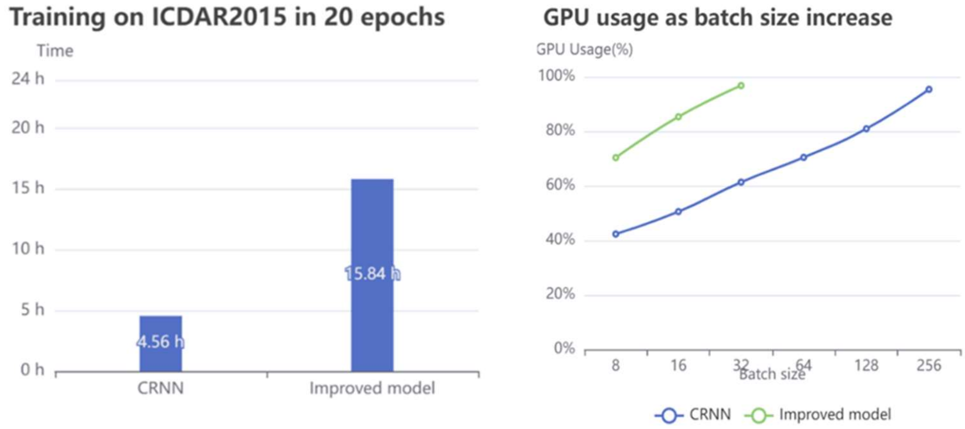
Although the model suggested in this work is more accurate than the original CRNN, it still has certain drawbacks. To connect this model to reality, we intend to strengthen these shortcomings in subsequent work. The recognition of curved text is poorly shown in Figure 17.

In addition, only two evaluation methods (loss and accuracy) were used to evaluate the model, which is not enough. Due to a large number of combined models, the training time is too long and, with the usage of memory increasing, deployment of the model has higher requirements for the deployment scenario. The training time and GPU usage are shown in Figure 18.



```
1: DOU BLE    0.825
2: IERS    0.974
3: CR    0.932
4: WO10    0.568
5: 3    0.569
```

Figure 17. Results of curved text recognition in sense of symmetry.



(**a**) (**b**)

Figure 18. (**a**) Results of training time. (**b**) Results of GPU usage.

6. Conclusions

This paper proposed a novel composite network model structure that combines the benefits of CRNN with other techniques such as text direction classifier, DBNet, Retinex algorithm, and CRNN. The model can effectively segment and recognize text in various backgrounds and orientations by applying the affine transformation, text direction classification, and clarity evaluation. The experiments on the training process and benchmark for scene text recognition demonstrated that the model can overcome the limitations of CRNN in complex and multi-oriented text scenes. It delivered higher accuracy and a wider application scope than the original CRNN model. This research contributes to the advancement of scene text recognition technology and provides new possibilities for future studies.

In the future, we will use knowledge distillation to compress the model by reducing the number of parameters to improve our model. Then, more evaluation methods should be added to evaluate the model. Furthermore, we will design a mini-app or website to deploy our model on mobile phones.

In addition, as an essential part of machine learning application, scene text recognition will be applied in more fields with the development of technology. When scene text recognition technology is combined with natural semantic recognition technology, the machine will have “comprehension”, that is, the ability to accurately understand the external world text content, providing the ability to structure the text. Future scene text recognition based on machine learning service providers will offer a wider range of cloud services in addition to a wider range of terminal carriers, such as smartphones and intelligent electronics, and lower the entry barrier and cost of use.

In recent years, there are more text recognition models which have brought people's attention. STAR-Net [23] emphasizes the importance of representative image-based feature extraction from text regions by the spatial attention mechanism and the residue learning strategy1. Combining the spatial attention mechanism with the residue convolutional blocks, STAR-Net is able to introduce a spatial attention mechanism by transforming a loosely bounded and distorted text region into a more tightly bounded and rectified text region. Also, ESIR [24] presents an innovative rectification network which employs a novel line-fitting transformation to estimate the pose of text lines in order to correct perspective and curvature distortions of scene texts iteratively1. The finally rectified scene text image is fed to a recognition network for further processing. FOTS [25] proposes RoIRotate to share convolutional features between detection and recognition. Benefiting from convolution sharing strategy, FOTS has little computation overhead compared to baseline text detection network, and the joint training method makes FOTS perform better than these two-stage method. In addition, the main advantage of SRN [26] is that it is a novel end-to-end trainable framework named semantic reasoning network (SRN) for accurate scene text recognition, where a global semantic reasoning module (GSRM) is introduced to capture global semantic context through multi-way parallel transmission1. The state-of-the-art results on 7 public benchmarks, including regular text, irregular text and non-Latin long text, verify the effectiveness and robustness of the proposed method1. In addition, the speed of SRN has significant advantages over the RNN based methods. Nowadays, there has been increasing interest in recognizing text in natural scenes in both academia and industry due to the rich text information in natural scenes which is very useful for vision-based applications such as industrial automation and image-based geo-location.

Author Contributions: Conceptualization, Y.L., Y.W. and H.S.; methodology, Y.L and Y.W.; software, Y.L. and Y.W.; validation, Y.L., Y.W. and H.S.; formal analysis, Y.L, Y.W. and H.S.; investigation, Y.L. and Y.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by BNU-HKBU United International College grant number R202114, UICR0400054-21CL, UICR0500005-21, Guangdong Higher Education Key Platform and Research Project grant number 2020ZDZX3039, Guangdong Provincial Key Laboratory of Interdisciplinary Research and Application for Data Science grant number 2022B1212010006. The APC was funded by [R202114].

Data Availability Statement: The used data are available as request.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Shi, B.; Bai, X.; Yao, C. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2298–2304. [CrossRef] [PubMed]
2. Kalchbrenner, N.; Grefenstette, E.; Blunsom, P. A convolutional neural network for modelling sentences. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Baltimore, MD, USA, 24–27 June 2014; pp. 655–665. [CrossRef]
3. Liu, Y.; Chen, H.; Shen, C.; He, T.; Jin, L.; Wang, L. Real-time scene text detection with differentiable binarization. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 11474–11481. [CrossRef]
4. Rahman, Z.; Jobson, D.J.; Woodell, G.A. Multi-scale retinex for color image enhancement. In Proceedings of the International Conference on Image Processing, Lausanne, Switzerland, 19 September 1996; IEEE: Piscataway, NJ, USA, 1996; Volume 3, pp. 1003–1006. [CrossRef]
5. Shi, B.; Bai, X.; Yao, C. Detecting Oriented Text in Natural Images by Linking Segments. In Proceedings of the Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2550–2559.
6. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C. SSD: Single shot multibox detector. In *Computer Vision—ECCV 2016*; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Springer International Publishing: Cham, Switzerland, 2016; pp. 21–37. [CrossRef]
7. Zhou, X.; Yao, C.; Wen, H.; Wang, Y.; Zhou, S.; He, W. EAST: An efficient and accurate scene text detector. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2017, Honolulu, HI, USA, 21–26 July 2017; pp. 5551–5560. [CrossRef]

8. Kim, K.H.; Hong, S.; Roh, B.; Cheon, Y.; Park, M. PVANet: Lightweight deep neural networks for real-time object detection. In Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016; pp. 1404–1408. [CrossRef]
9. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015*; Navab, N., Hornegger, J., Wells, W., Frangi, A., Eds.; Springer International Publishing: Cham, Switzerland, 2015; pp. 234–241. [CrossRef]
10. Wang, W.; Xie, E.; Li, X.; Hou, W.; Lu, T.; Yu, G.; Shao, S. Shape Robust Text Detection with Progressive Scale Expansion Network. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 9328–9337. [CrossRef]
11. Karatzas, D.; Gomez-Bigorda, L.; Nicolaou, A.; Ghosh, S.; Bagdanov, A.; Iwamura, M.; Matas, J.; Neumann, L.; Chandrasekharan Prasad, V.R.; Busta, M. ICDAR 2015 Competition on Robust Reading, Nancy, France, 23–26 August 2015. Available online: https://deepai.org/dataset/icdar-2015 (accessed on 30 November 2015).
12. Sidorov, O.; Hu, R.; Rohrbach, M.; Singh, A. TextCaps: A Dataset for Image Captioning with Reading Comprehension. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2020; pp. 725–741. Available online: https://textvqa.org/textocr/ (accessed on 9 January 2023).
13. Ch'ng, C.S.; Chan, C.S. Total-Text: A Comprehensive Dataset for Scene Text Detection and Recognition. In Proceedings of the 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), Kyoto, Japan, 9–15 November 2017; pp. 935–942. Available online: https://tc11.cvc.uab.es/datasets/Total-Text_1 (accessed on 28 October 2017).
14. Ipythonx. TotalTextStr. 2018. Available online: https://github.com/cs-chan/Total-Text-Dataset (accessed on 27 October 2017).
15. Zhang, Z.; Zhang, C.; Shen, W.; Yao, C.; Liu, W.; Bai, X. Multi-oriented text detection with fully convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 4159–4167.
16. Tian, Z.; Huang, W.; He, T.; He, P.; Qiao, Y. Detecting Text in Natural Image with Connectionist Text Proposal Network. In *Computer Vision—ECCV 2016*; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; ECCV 2016. Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2016; Volume 9912. [CrossRef]
17. Yao, C.; Bai, X.; Sang, N.; Zhou, X.; Zhou, S.; Cao, Z. Scene text detection via holistic, multi-channel prediction. *arXiv* **2016**, arXiv:1606.09002.
18. Poma, Y.; Poma, A. Adaptation of Number of Filters in the Convolution Layer of a Convolutional Neural Network Using the Fuzzy Gravitational Search Algorithm Method and Type-1 Fuzzy Logic. *J. Artif. Intell. Soft Comput. Res.* **2022**, *12*, 223–235. [CrossRef]
19. Karatzas, D.; Gomez-Bigorda, L.; Nicolaou, A.; Ghosh, S.; Bagdanov, A.; Iwamura, M.; Matas, J.; Neumann, L.; Chandrasekhar, V.R.; Lu, S.; et al. ICDAR 2015 competition on robust reading. In Proceedings of the 2015 13th International Conference on Document Analysis and Recognition (ICDAR), Tunis, Tunisia, 23–26 August 2015.
20. Koo, H.; Kim, D.H. Scene text detection via connected component clustering and nontext filtering. *IEEE Trans. Image Process.* **2013**, *22*, 2296–2305. [PubMed]
21. Yin, X.C.; Yin, X.; Huang, K.; Hao, H. Robust text detection in natural scene images. *IEEE Trans. PAMI* **2014**, *36*, 970–983.
22. Yin, X.C.; Pei, W.Y.; Zhang, J.; Hao, H.W. Multi-orientation scene text detection with adaptive clustering. *IEEE Trans. PAMI* **2015**, *37*, 1930–1937. [CrossRef] [PubMed]
23. Liu, W.; Chen, C.; Chen, C.; Wong, K.-Y.K.; Su, Z.; Han, J. STAR-net: A spaTial attention residue network for scene text recognition. In Proceedings of the British Machine Vision Conference, York, UK, 19–22 September 2016; pp. 1–13. Available online: http://www.bmva.org/bmvc/2016/papers/paper043/paper043.pdf (accessed on 3 January 2023).
24. Sun, Y.-F. ESIR: End-to-end Scene Text Recognition via Iterative Image Rectification. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 7264–7273. [CrossRef]
25. Wang, F.-L. FOTS: Fast Oriented Text Spotting with a Unified Network. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 5676–5685. [CrossRef]
26. Yu, J.-C. Towards Accurate Scene Text Recognition with Semantic Reasoning Networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 3352–3361. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.