

# Detecting and recognizing seven segment digits using a deep learning approach

*Loi Ming Low*<sup>1\*</sup>, *Faridah Hani Mohd Salleh*<sup>1</sup>, *Yi Feng Law*<sup>1</sup>, and *Nor Zaity Zakaria*<sup>1</sup>

<sup>1</sup>Department of Foundation and Diploma Studies, College of Computing and Informatics, Universiti Tenaga Nasional, 43000, Kajang, Selangor, Malaysia.

**Abstract.** Recognizing seven-segment digits is a specific task within the broader field of text detection and recognition. Seven-segment digits are commonly used for displaying numerical information in various applications. However, accurately detecting and recognizing these digits can be challenging due to factors like LED bleeding, glare, and the presence of printed text alongside the digits. The experiment described in this paper aims to identify the most effective models for detecting and recognizing texts and assess their accuracy and performance under different environmental conditions. The experiment reveals that DBNet from PaddleOCR is the best model for text detection, while PARSeq has the best accuracy for recognizing seven-segment digits on the 7Seg dataset. PARSeq also performs well on a custom dataset with lower LED ratios but struggles with glare conditions. Excluding special characters improves accuracy in all conditions.

## 1 Introduction

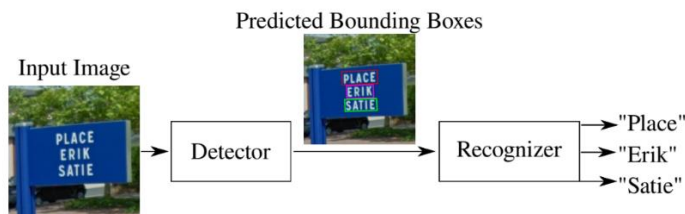
Seven-segment displays are widely used for showing numbers on screens, known for their energy efficiency and affordability in applications like industry and households. This has given rise to Seven-Segment Optical Character Recognition (SSOCR) as part of Optical Character Recognition (OCR), driven by deep learning progress. While SSOOCR has mainly focused on specific fields such as healthcare [1, 2] and industry [3, 4, 5, 6], its use in everyday settings with hardware limitations and environmental factors has been limited. In these scenarios, challenges arise when images contain both printed text and seven-segment displays. Assessing text detection in real-life situations, especially in devices with both text and seven-segment digits, can be complex. Pre-trained models may struggle to differentiate them, leading to unreliable outcomes. Furthermore, issues like LED bleeding, which blurs the decimal point with nearby digits, make precise seven-segment digit detection difficult. Fortunately, recent advancements in scene text detection and recognition offer more robust solutions for diverse scenes and font variations.

---

\* Corresponding author: [ming@uniten.edu.my](mailto:ming@uniten.edu.my)

## 2 OCR Pipeline

OCR, a computer vision subfield, involves converting text images into machine-readable format. It traditionally extracts text from scanned documents and modernly focuses on scene text recognition. Figure 1 depicts the core OCR pipeline processes: text detection and recognition.



**Fig.1.** Overview of the OCR pipeline.

Scene text detection in scenes is a critical step in OCR pipelines, as it pinpoints text areas using bounding boxes. Text detection methods can be classified into two main groups: regression-based and segmentation-based. Regression-based techniques adapt object detection for regular text but struggle with irregular text. Segmentation-based methods excel at diverse text shapes but may be slower and have difficulty with overlapping text. Scene text recognition predicts text from cropped scene images, often obtained through text detection. These algorithms can be grouped into two categories: regular text recognition, handling horizontal text like printed fonts, and irregular text recognition, managing curved, blurred, covered, or non-horizontal text. There are more recognition options available for irregular text. Several OCR toolboxes are available, including Paddle OCR, MMOCR, EasyOCR, and Tesseract. Refer to Table 1 for the comparison between open-source OCR toolboxes.

**Table 1.** Comparison between open-source OCR toolboxes. Adapted from [7].

Toolbox	Tesseract	EasyOCR	PaddleOCR	MMOCR
DL library	-	Pytorch	PaddlePaddle	Pytorch
Language #	100+	80+	80+	2
Detection	Convention LSTM	CRAFT, DB	DB, DB++, DRRG, FCENet, PSENet, EAST, SAST	DB, DB++. DRRG, FCENet, Mask R-CNN, PANet, PSENet, Textsnake
Recognition	Convention	CRNN	CRNN, Rosetta, STAR-Net, RARE, SRN, NRTR, SAR, SEED, SVTR, ViTSTR, ABINet, VisionLAN, SPIN, RobustScanner, RFL	ABINet, CRNN, MASTER, NRTR, RobustScanner, SAR, SATRN, SegOCR, CRNN-STN

### 3 Research methodology

Pretrained models have been chosen based on benchmark results from ICDAR 2013 and ICDAR 2015. These models are selected for transfer learning. The datasets used in this experiment were captured using smartphones and consist of horizontal seven-segment digits, often with slight rotation and blurriness. The similarity between these datasets and the pretrained models allows for the application of features learned from the models to improve seven-segment digit detection and recognition. This approach of transfer learning results in significantly better performance compared to training with a limited amount of data. The detailed information about the ICDAR 2013 and ICDAR 2015 datasets presented in Table 2.

**Table 2.** Details of ICDAR 2013 and ICDAR 2015 Adapted from [8].

Dataset	Orientation	Annotation	Additional Information
ICDAR2013	Horizontal	Character/word	Captured using a variety of hardware
ICDAR2015	Multi-oriented	Word	Some images are out of focus

### 4 Experiment of seven segment digits detection using scene text detection pre-trained models

The purpose of the experiment is to identify the appropriate scene text detection model, which can detect the presence of seven segment digits from images. The metric used for evaluation is the F-measure, specifically the balanced F-measure, which is defined as the weighted harmonic mean of precision and recall. By considering both precision and recall, it provides a single score for assessing the performance of classification machine learning models. The ICDAR 2015 Evaluation Protocol for text detection [9], based on IoU (Intersection over Union), is employed to automatically calculate the F-score for each model. The experiment will involve running each model on the same dataset and comparing the computed F1 score. The evaluation script, ground truth, predicted texts, and results are accessible at: <https://github.com/Milsk01/FYP>.

#### 4.1 Datasets collection and pre-processing

For simplicity, the testing dataset from the Seven Segment Display (SSDI) Dataset, which consists of 257 labelled images featuring only seven-segment digits, will be used for the evaluation. The SSDI dataset uses .pickle format for labels, containing bounding boxes for each seven-segment digit in test images. To align with the ICDAR 2015 evaluation script, a Python script has been created. It creates ground truth text files for each test image, consolidating multiple bounding boxes into one for streamlined evaluation. The watermark logo from the bottom right of test images have been removed to ensure accurate results.

#### 4.2 Evaluation metrics

Apart from precision and recall, Intersection Over Union (IoU) is also used in this experiment. IoU is a measure based on the Jaccard Index that quantifies the degree of overlapping between two bounding boxes. Figure 2 shows the equation used for the calculation of IoU.

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$

**Fig.2.** Equation of IoU.

### 4.3 Result and Discussion

Table 3 shows the result of the experiment. Due to the incompatibility issue, TextFuseNet and CharNet are run on Google Collab instead of the local environment. Thus, the Frame Per Second (FPS) results for both TextFuseNet and CharNet are not included. TextFuseNet, DB, and PANet are the best models from the ICDAR 2013 & 2015 benchmark, PaddleOCR and MMOCR, respectively, in terms of F-score and FPS on the SSDI dataset. In general, all models excluded CharNet have achieved an average F-score of  $79.88 \pm 0.0822$ . A low precision in all models except CharNet is observed, which reflects the prevalence problem of misdetection in text detection algorithms. Due to the limited size and variability of the dataset, the result showed no correlation between the F-score on the ICDAR 2013 & 2015 benchmark and the performance of seven segment digits detection. Overall, DBNet from PaddleOCR is the best model among all tested pretrained models in terms of F-score and FPS.

**Table 3.** Effectiveness of scene text detection algorithms in detecting seven segment digits.

Source	Method	Precision (%)	Recall (%)	F-Score (%)	FPS
ICDAR 2013 & 2015 Benchmark	<b>TextFuseNet</b>	<b>66.58</b>	<b>100</b>	<b>79.81</b>	-
	CharNet	95.58	21	34.45	-
PaddleOCR	<b>DB</b>	<b>66.67</b>	<b>100</b>	<b>80</b>	<b>8.7</b>
	DB++	66.58	100	79.94	6.7
	PSE	66.49	100	79.88	4.8
	SAST	66.45	100	79.78	3.7
MMOCR	DB_r18	66.49	100	79.88	4.9

	DB_r50	66.49	100	79.88	1.5
	DBPP	66.67	100	80	1.3
	MaskRCNN	65.98	100	79.51	1.2
	<b>PANet</b>	<b>66.67</b>	<b>100</b>	<b>80</b>	<b>5.2</b>
	PSNet	66.67	100	80	0.7

### 5 Experiment of seven segment digits recognition using scene text recognition pre-trained model

Seven-segment digits detection is the process of predicting seven-segment digits from input images, as shown in Figure 3. The purpose of the experiment is to identify the appropriate scene text recognition model, which can reliably recognise seven-segment digits from images under different environmental conditions. The datasets which will be used in this experiment are 7Seg [10] and a custom dataset collected from data contributors from Universiti Tenaga Nasional.



**Fig.3.** Example of text recognition result using PARSeq model.

The first stage of the experiment will be conducted by running each text recognition algorithm on the 7Seg dataset. Then, the list of predicted text is compared against the ground truth. The ICDAR 2015 evaluation protocol based on a standard edit distance metric [11] will be used to evaluate text recognition accuracy and character error rate. Edit distance, also known as Levenshtein distance, is a metric to measure the difference between two strings and is given by:

$$lev_{a,b}(i, j) = \begin{cases} \max(i, j) & \text{if } \min(i, j) = 0, \\ \min \begin{cases} lev_{a,b}(i - 1, j) + 1 \\ lev_{a,b}(i, j - 1) + 1 \\ lev_{a,b}(i - 1, j - 1) + 1_{(a_i \neq b_j)} \end{cases} & \text{otherwise.} \end{cases} \quad (1)$$

The more similar the two string sequences are, the lower the number of edits needed. Thus, it results in a lower Levenshtein distance. For comparison purposes, the Levenshtein distance will be used to evaluate the Character Error Rate (CER). It can be represented using this formula below:

$$CER = \frac{S + D + I}{N} = \frac{\text{Levenshtein distance}}{N} \quad (2)$$

where:

S = Number of substitution  
 D = Number of deletion  
 I = Number of insertion  
 N = Number of characters in text

The well-performing pretrained models will be used in the second stage of the experiment to identify the performance of the selected scene text recognition algorithm under various environmental conditions.

### 5.1 Data collection and pre-processing

The first dataset was obtained from Paddle AI studio called 7Seg. It contains 189 images of seven-segment digits scraped from the web. However, after manual filtering, only 165 images are seven-segment digits in the natural scene. Conforming to the requirement of the ICDAR 2015 evaluation script for text recognition, every image file is renamed with the format "word\_###.png". There is no ground truth provided by the author. A Python script was created to generate ground truth by extracting the actual text from the filename. The second dataset is a primary dataset collected from 88 data contributors from Universiti Tenaga Nasional for text recognition. The dataset includes images of a seven-segment display on common household appliances and electronic devices, using a smartphone camera. A random line of seven segment digits is cropped out of the image and manually labelled.

### 5.2 Result and Discussion

Table 4 shows the experimental results conducted using the 7Seg dataset. Two main evaluation metrics, accuracy and average character error rate are used to tabulate the result. The average character error recognition is obtained by totalling the character error rate for each image and dividing it by the total number of images. Thus, the pretrained models can be evaluated from word-level accuracy and character-level accuracy. Overall, the PARSeq model has the best accuracy of 56.97 % with the lowest average CER compared to other pretrained models from PaddleOCR and MMOCR. One of the possible reasons for the performance gap is the use of multiple datasets, including synthetic and real-world datasets for training PARSeq. Besides that, as all text recognition pretrained models from PaddleOCR using MJSynth and SynthText for training, it is safe to say the pretrained model with the best performance from PaddleOCR on the 7Seg dataset is SVTR, which only has a single vision mode. Finally, although NRTR has the best accuracy among all the pre-trained models from MMOCR, ABINet has the lowest average character error rate.

**Table 4.** Effectiveness of scene text recognition algorithms in recognizing seven segment digits.

Source	Method	Accuracy (%)	Average CER (%)
PARSeq	ABINET	49.09	29.41
	CRNN	49.09	26.23
	<b>PARSeq</b>	<b>56.97</b>	<b>22.35</b>
	Parseq-tiny	53.94	26.17




	TRBA	55.76	23.18
	ViSTR	46.06	25.6
PaddleOCR	ABINet	24.24	49.85
	CRNN	18.79	64.36
	RARE	20.61	57.2
	RFL	11.52	83.96
	Robust Scanner	13.33	64.71
	Rosetta	12.73	65.07
	SPIN	8.48	184.24
	SRN	10.91	82.64
	STAR-NET	21.21	49.88
	<b>SVTR</b>	<b>34.34</b>	<b>37.45</b>
	VisonLAN	22.89	49.99
	ViSTR	12.73	73.07
	<b>Source</b>	<b>Method</b>	<b>Accuracy (%)</b>
MMOCR	ABINet	20	<b>49.95</b>
	CRNN	6.67	81.56
	CRNN_TPS	3.03	100.08
	MASTER	12.12	75.11
	<b>NRTR</b>	<b>24.85</b>	52.96

	Robust Scanner	12.12	76.15
	SAR	18.79	66.49

\* Notes: CER values can exceed 100%, especially with many insertions

Table 5 shows the result of using PARSeq to detect seven-segment digits on the custom dataset. Compared to the performance on the 7Seg dataset, PARSeq has a higher accuracy on the custom dataset due to the lower ratio of LEDs in the dataset. A common problem that exists in recognizing seven-segment digits on LEDs is the noise from the turned-off segment and improper cropping.

**Table 5.** Results of using PARSeq.

		
<b>Fig.4. (a):</b> Example of misrecognition of digit 1 as 8.	<b>Fig.4. (b):</b> Example of bleeding effect.	<b>Fig.4. (c):</b> Example of occlusion.

After excluding all the special characters, PARSeq shows a significant improvement in accuracy in all 4 conditions. This is due to the lack of representation of these characters in the scene text. Besides that, the decimal point is easily fused into adjacent digits due to the ‘bleeding’ effect as shown in Figure 4(b). Out of 4 conditions, the PARSeq model has the worst performance under the glare condition. Parts of the digits might be occluded by the glare or reflection of the light, making it difficult to recognize the digit, as shown in Figure 4(c).

**Table 6.** Effectiveness of PARSeq in recognizing seven segment digits in different conditions.

Condition	Accuracy with special characters (%)	Accuracy without special characters (%)
Ideal	79.55	93.18
Blurry	71.43	91.43
Rotated	69.04	86.71
Glare	68.18	81.82

## 6 Conclusion

The objective of this paper is to evaluate the performance of various pretrained models for text detection and recognition on different datasets, specifically focusing on the detection and recognition of seven-segment digits. The experiment reveals the performance of various pretrained models for text detection and recognition on different datasets. Overall, DBNet



from PaddleOCR is the best model for text detection, while PARSeq has the best accuracy for recognizing seven-segment digits on the 7Seg dataset. PARSeq also performs well on a custom dataset with lower LED ratios but struggles with glare conditions. Excluding special characters improves accuracy in all conditions.

The successful completion of this paper and the associated research owes its realization to the outstanding support extended by the Yayasan Canselor Universiti Tenaga Nasional (YCU), under Grant No: 202210037YCU.

## References

1. S. Roy, "Application of Convolutional Neural Networks in Mobile Devices for Inferring Readings from Medical Apparatus," *International Journal of Research and Scientific Innovation (IJRSI)*, Jan. 2017, Accessed: Nov. 30, 2022. [Online]. Available: [https://www.academia.edu/36599646/Application\\_of\\_Convolutional\\_Neural\\_Networks\\_in\\_Mobile\\_Devices\\_for\\_Inferring\\_Readings\\_from\\_Medical\\_Apparatus](https://www.academia.edu/36599646/Application_of_Convolutional_Neural_Networks_in_Mobile_Devices_for_Inferring_Readings_from_Medical_Apparatus)
2. E. Finnegan, M. Villarroel, C. Velardo, and L. Tarassenko, "Automated method for detecting and reading seven-segment digits from images of blood glucose metres and blood pressure monitors," *J Med Eng Technol*, vol. 43, no. 6, pp. 341–355, Aug. 2019, doi: 10.1080/03091902.2019.1673844.
3. R. P. Ghugardare, S. P. Narote, P. Mukherji and P. M. Kulkarni, "Optical character recognition system for seven segment display images of measuring instruments," *TENCON 2009 - 2009 IEEE Region 10 Conference, Singapore, 2009*, pp. 1-6, doi: 10.1109/TENCON.2009.5395994.
4. Y. Li and H. Qian, "Automatic Recognition System for Numeric Characters on Ammeter Dial Plate," *2008 The 9th International Conference for Young Computer Scientists, Hunan, China, 2008*, pp. 913-918, doi: 10.1109/ICYCS.2008.498.
5. Y. Abdelrahman, M. El-Salamony, and M. Khalifa, "Visual Data Acquisition for Measuring Devices using Deep Learning," *NILES 2021 - 3rd Novel Intelligent and Leading Emerging Sciences Conference, Proceedings*, pp. 301–304, 2021, doi: 10.1109/NILES53778.2021.9600549.
6. M.B. Akanle, and V. Oguntosin, "A digital indicator system with 7-segment display." *Journal of Physics: Conference Series*. 1299. 012139. 10.1088/1742-6596/1299/1/012139 (2019).
7. Z. Kuang SenseTime Research China Hongbin Sun SenseTime Research China Zhizhong Li SenseTime Research China Xiaoyu Yue et al., "MMOCR: A Comprehensive Toolbox for Text Detection, Recognition and Understanding," *MM 2021 - Proceedings of the 29th ACM International Conference on Multimedia*, pp. 3791–3794, Aug. 2021, doi: 10.48550/arxiv.2108.06543
8. E. Rainarli, Suprpto, and Wahyono, "A decade: Review of scene text detection methods," *Comput Sci Rev*, vol. 42, p. 100434, Nov. 2021, doi: 10.1016/J.COSREV.2021.100434
9. "My Methods - Incidental Scene Text - Robust Reading Competition." <https://rrc.cvc.uab.es/?ch=4&com=mymethods&task=1> (accessed Dec. 25, 2022)
10. "中等规模的七段数码管数据集 - 飞桨AI Studio." <https://aistudio.baidu.com/aistudio/datasetdetail/124936> (accessed Dec. 26, 2022)
11. "My Methods - Incidental Scene Text - Robust Reading Competition." <https://rrc.cvc.uab.es/?ch=4&com=mymethods&task=3> (accessed Dec. 26, 2022)