Received: 4 March 2024 | Revised: 24 September 2024 | Accepted: 25 September 2024 | Published online: 8 October 2024

REVIEW

Artificial Intelligence and Applications
2024, Vol. 2(4) 257–277
DOI: 10.47852/bonviewAIA42022755

A Comprehensive Review on Text Detection and Recognition in Scene Images

Umapada Pal[1], Arnab Halder[1,2,*], Palaiahnakote Shivakumara[3] and Michael Blumenstein[2]

[1]*Computer Vision and Pattern Recognition Unit, Indian Statistical Institute, India*

[2]*University of Technology Sydney, Australia*

[3]*School of Science, Engineering and Environment, University of Salford, UK*

Abstract: Detecting and recognizing text in natural scene images and videos is vital for several real-world applications, such as in the analysis of Crime scene CCTV footage, sports videos, and autonomous driving, to name a few. Therefore, one can expect several challenges, namely arbitrarily oriented and shaped text detection and identification in movies and natural environments. Many methods have been developed in the past to address these challenges, including advanced deep-learning models and transformers. Due to several methods available in the literature, it is not so easy to understand the open challenges, applications, directions, scope, limitations, and weaknesses of the methods. Therefore, there is a need to write a survey/review to highlight and discuss the strengths and weaknesses of the developed methods. This survey/review presents different categories of work and discusses their importance, limitations, new challenges, applications, and, finally, directions such that readers can choose appropriate methods and directions to carry out research work in the field of text detection/recognition in the natural scene and videos.

Keywords: text detection, text recognition, text spotting, text classification, scene text, car number plate detection, optical character recognition

1. Introduction

The advent of text detection in scenes can be traced back to the 1990s and has been attributed to ground-breaking studies conducted by pioneers such as Greenhalgh and Mirmehdi [1], Yin et al. [2], Ham et al. [3], and Shilkrot et al. [4]. The surge in the implementation of Internet Technology, in conjunction with the ubiquity of portable mobile devices, has given rise to an escalating number of applications wherein the extraction of text from image data and videos has become imperative. Currently, the ability to detect and recognize text from images and video scenes has swiftly emerged as a vital research focus within Computer Vision and Pattern Recognition, distinct from but related to traditional Document Analysis and Recognition. Prominent international conferences, such as the International Conference on Document Analysis and Recognition (ICDAR), the International Conference on Computer Vision, Computer Vision and Pattern Recognition, the European Conference on Computer Vision, and the AAAI Conference on Artificial Intelligence, have recognized the paramount importance of scene text detection and recognition (STDR), dedicating it as an independent stream of academic research.

STDR is the technique of detecting and localizing textual components in photographs collected from real-world settings. STDR not only detects and locates pictures but also extracts critical high-level semantic data from them [5–7]. As a result, its value is felt across a wide range of industries, including, but not limited to, risk and knowledge management, cybercrime countermeasures, content augmentation, and fraud deterrence. Furthermore, STDR improves knowledge extraction from photos or video information at many granular levels, such as full pages, discrete text lines, specific phrases, and even single characters. Following text detection, recognition becomes an important component in a variety of computer vision applications, including automated sign interpretation, autonomous vehicle operation, language translation, and multimedia retrieval. The implementation of driver-assist systems or autonomous automobiles, which rely largely on such technology, is an appropriate demonstration of the aforementioned use cases, considerably boosting passenger safety and overall security [8, 9].

For example, sample scene images shown in Figure 1 represent scene images of industry applications where detecting moving objects, including humans, in day and night images is considered. In Figure 1, the bounding boxes of the text indicate the results of text detection and localization. For illustrating sample text detection, we use the method [10] to fix bounding boxes for the text lines in the images. Similarly, Figure 2 shows scene images with different complexities. These images are challenging for text detection and recognition. Therefore, choosing an appropriate method for detecting and recognizing text successfully is a hard task when several methods or models have been proposed in the literature [5–7].

2. Motivation

As illustrated by the performance of text detection methods in Figure 1, the method [10] works well, while the method [11] does not

*Corresponding author: Arnab Halder, Computer Vision and Pattern Recognition Unit, Indian Statistical Institute, India and University of Technology Sydney, Australia. Email: arnabhalder1997@gmail.com

Figure 1
Text detection from video frames

Night video frame **Day video frame**

Existing Halder Model [85]

Existing EAST Model [47]

Figure 2
Some challenging scenarios

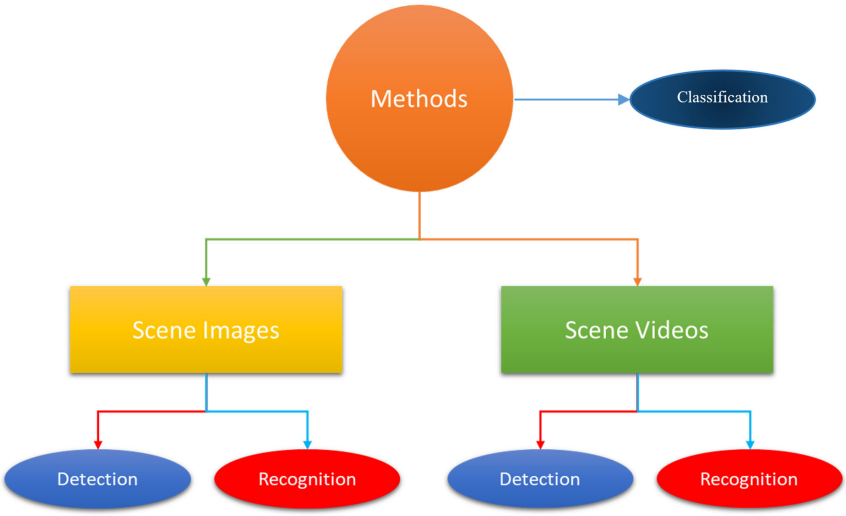
Foggy-Night **Night-Vision** **Smoky**

work well. For the images shown in Figure 2, both methods fail to detect text. This shows that although many methods/ models have been developed in the past, there is still confusion about choosing an appropriate method according to the complexity of the images and applications. This is because there are several models and methodologies that are offered with varying goals and bases and that employ various metrics, datasets, assessment schemes, etc. Therefore, in order to select a suitable technique based on user objectives and applications, it is required to analyze the current methods in order to validate the open challenges, limits, and strengths of the methods. Additionally, the evaluation may be utilized to identify unresolved issues and sources for developing novel models. To assist users in selecting the most appropriate method for their objectives and applications, we outline key criteria for evaluation. These include the characteristics of the text and scene (e.g., font style, background complexity, and motion), robustness to shakiness or motion, accuracy versus speed trade-offs, scalability and computational complexity, and specific constraints posed by unique applications (e.g., real-time processing in drones or underwater text detection). By considering these factors, users can make informed decisions and identify methods that best suit their requirements.

3. Proposed Plan

This work focuses on reviewing the methods of text detection and recognition in natural scene and videos, as well as classifying the text as shown in Figure 3. To capture a clear picture of the review, our plan is to divide the methods of text detection into regression [12], segmentation [13–15], and transform-based methods [16–18]. Similarly, recognition techniques are classified as transform and segmentation-based, encoder-decoder and attention-based, and connectionist temporary classification-based techniques. The sections that follow provide a thorough examination and analysis

Figure 3
Block diagram of the proposed plan



of each approach in relation to the various categories. However, those studies thoroughly address both machine learning and traditional techniques, since there are several surveys and reviews on text detection and identification accessible in the literature [5, 11, 19]. Consequently, the work here focuses on approaches for review based on deep learning.

4. Research Methodology

4.1. Benchmark datasets

In order to find and identify the text, selecting the right dataset is crucial. The well-known datasets for text detection and identification are introduced in this section. Furthermore, the attributes of various datasets are displayed in the table below.

ICDAR 2003 dataset [20]: For robust text localization, this dataset comprises 258 training images and 251 test images.

ICDAR 2011 dataset [21]: This dataset consists of 255 test pictures and 299 training photos.

ICDAR 2013 dataset [22]: In 462 photos, this dataset comprises 1189 words and 6393 letters. This package contains 233 test pictures and 229 training shots. The dataset contains several sophisticated examples, such as reflection and language generated in difficult situations, yet the text is always in English and is often horizontal. Low-contrast photos with strange typefaces and stark backgrounds are examples of flawed images.

ICDAR 2015 dataset [23]: For strong text localization, this dataset contains 1000 training pictures and 500 test images. This includes photos with text in various sizes. Pictures that occur in an image of words of varying sizes are examples of bad images.

ICDAR 2017 dataset [24]: There are 18,000 images in this collection that include texts in curved, multidirectional, horizontal, and multilingual orientations. Scene photographs are included in nine different languages. For the purpose of localizing robust texts, ICDAR 2017 includes 7200 training pictures, 1800 validation images, and 9000 test images. There aren't many vertical texts in this collection. This collection includes images in a variety of sizes with glittery or cluttered backgrounds.

ICDAR 2019 dataset: For the purpose of localizing robust texts, ICDAR 2017 includes 7200 training pictures, 1800 validation images, and 9000 test images. There aren't many vertical texts in this collection. This collection includes images in a variety of sizes with glittery or cluttered backgrounds. There are 1000 test and 1000 training photos in this dataset. This set of images features text in several sizes and orientations, such as directed, curved, and vertical. Bad photographs are those that include lettering arranged in different sizes, flashy, crowded backgrounds, and different orientations. Compared to the ICDAR 2017 collection, there are significantly more curved and vertical pictures in this dataset. Compared to ICDAR 2017, this collection's images are more intricate.

NEOCR: Multi-oriented words accompany photos of natural scenery in the NEOCR collection. It has 5238 marked bounding boxes and 659 real-world pictures. Given that the texts in it are written in a number of languages, including English, Hungarian, Russian, Turkish, and Czech, this dataset is multilingual.

KAIST [25, 26]: Three thousand photos taken in both indoor and outdoor settings with different lighting conditions make up the KAIST scene text dataset. This database, which contains texts in both Korean and English, also acts as a multilingual benchmark. Additionally, binary masks for every character in the pictures are included. This means that this dataset may be applied to both segmentation and text localization tasks.

SVHN [27]: The street view house numbers collection contains more than 730000 digits in natural landscapes. The numerals are chopped home numbers from Google Street View photographs. This benchmark dataset is primarily used for the development and testing of digit recognition algorithms.

MSRA-TD500 dataset: This dataset includes 500 natural photos of the interior (office and market) and outdoor (street) settings captured with a pocket camera. This dataset contains 300 training photos and 200 test images. Instead of words, text lines are employed as the primary unit in this dataset. This dataset comprises Chinese, English, or a mix of the two. This collection includes photos with

directed and curved text as well as text in various sizes. There are only a few vertical texts in this collection.

Char74K dataset: This collection consists of sixty-two courses. These 62 classes are made up of the numerals 0 through 9, the letters A through Z, and the letters A through Z. This dataset contains 75,776 characters that were extracted from various images. This comprises 62,992 characters made with different fonts and 12,784 handwritten characters produced using tablets, each of which has 1016 data. Its name comes from the fact that these data are a subset of a bigger dataset that contains 74,000 photographs.

IIIT 5 K-Words (IIIT5K) dataset [28]: There are 2000 training photos and 3000 testing images in this dataset. There is a 1000-word vocabulary and a 50-word vocabulary for each image. Images featuring a range of fonts, colors, sizes, noise, blurring, and really low resolutions are included in this collection.

MLT-2017 dataset [29]: The dataset is multilingual. It has nine languages that correspond to six distinct scripts. This dataset consists of 7,200 training photos, 1,800 validation images, and 9,000 testing images.

MLT-2019 dataset: This dataset is multilingual. This builds upon MLT-2017. It has ten languages that correspond to seven distinct scripts. Chinese, Japanese, Korean, English, French, Arabic, Italian, German, Bangla, and Hindi (Devanagari) are among the languages spoken there. This dataset contains 10,000 photos for training, 2,000 images for validation, and 10,000 images for testing.

Street View Text dataset [30]: This dataset includes 647 clipped text, 249 testing street view pictures, and 101 training shots. Pictures from Google Street View may be downloaded. This dataset includes photos with noise, blurring, and lighting variations. Each picture has its own 50-word glossary.

COCO-Text dataset [31]: Text identification and recognition in natural photography is done using this dataset. The collection consists of 63,686 photographs with 173,589 labeled text parts. The 43,686 training photos and the 20,000 validation images make up the two halves of the dataset. There are issues with natural images in the collection. This extensive dataset contains text with three different orientations: curved, random, and horizontal.

Total-Text dataset [32]: This dataset consists of 9330 annotated words in three different text orientations (curved, multi-oriented, and horizontal) and 1555 scene photos. Total-Text is split into two sets of 300 and 1255 images, respectively, for the training and test sets. There are several issues with this dataset, including different text fonts, text orientations, and picture backdrops.

SynthText [33]: There are 858750 images in all. This dataset comprises annotations at the word and character levels, as well as text recognition material, which may be used to train text detection and recognition models.

Synth90K [33]: The Synth90k dataset comprises 9 million synthetic images with text, with 7.2 million serving as training sets, 900000 serving as verification sets, and the remaining 900000 serving as testing sets.

The details of the standard datasets of text detection are presented in Table 1, where one can see different characteristics and nature of datasets, such as type of orientation, scripts, the number of training, testing samples, and ground truth at different levels and sizes of the datasets.

4.2. Methods for text detection and recognition in natural scene images

4.2.1. Detection methods

Text detection recognizes the position of text in a scene, locates the text, records the image, and feeds it into the text recognition model to provide anticipated text results. It consists of seven steps: input the picture, transform the image to a fixed size, normalize, extract the image's characteristics, determine the location of the text, and, lastly, run a series of processing steps to obtain the projected output. Text detection models may be classified into two types based on how they are implemented: regression text detection models [13, 34–36] and segmentation text detection models. Text detection using regression models first determines the text's position coordinates and then fits them to the actual box using regression. On the other hand, text detection based on segmentation models judges each pixel point using text detection as a segmentation problem.

Regression-based Text Detection Models: Connectionist Text Proposal Network (CTPN) [12] was created better to identify the placement of text in a scene, which employed a vertical anchor regression approach to find small-scale text candidate boxes in text identification. The CTPN model only forecasts the output text box's horizontal offset—not its horizontal direction—because the text's length is not fixed. In addition, the CTPN model uses a Recurrent Neural Network (RNN) loop network to get text lines, links small-scale text that has been identified, and applies end-to-end training. The CTPN model does not require post-processing and covers a wide range of languages and scales. The CTPN model, on the other hand, is poor at identifying non-horizontal text. In the year 2017, Shi et al. suggested the Link Segments (SegLink) model, which is similar to the CTPN model. Both models are identical in detecting a portion of a text line and then connecting all the pieces to produce the entire text line. To handle text in multiple directions, the SegLink model contains a rotation angle and then employs a fusion rule to fuse frame and line information at each stage to generate text lines. The SegLink model, on the other hand, cannot recognize extensive spacing lines or curved text.

Zhou et al. [11] presented the Efficient and Accurate Scene Text Detector (EAST) model. The core principle of the EAST model is to separate and label whole text lines before merging them. It is possible to separate text detection into steps before detecting them, which adds time and compromises the accuracy of text recognition as well as the use of intermediate processing. In addition to text boxes, the EAST model also predicts text box positions and angles. By incorporating local-aware non-maximum suppression (NMS) into the EAST model, NMS complexity is reduced while accuracy and speed are raised. Nevertheless, the EAST model's range of vision is constrained, and it is not very good at identifying long texts. There are two sections to the generic text detection network model. To begin, the text region (TR) proposal network is employed to extract text ideas. Second, a refinement network is employed to validate and enhance these recommendations. In 2019, Wang et al. introduced the Adaptive Text Region Representation (ATRR) model for detecting text in any shapes in scenes. The ATRR model uses an adaptive text area representation based on a recursive neural network to thin the text area. It does this by predicting two boundary points every time step until it detects the presence of an expected stop tag. The ATRR model is represented by an arbitrary number of suitable boundary points and is capable of identifying text sections of any shape.

Table 1
Details of various standard datasets for text detection

Dataset	Orientation	Annotation	Number of images in the training set	Scripts
Datasets for Text Detection				
Regular Text				
ICDAR2003 [20]	Horizontal	Character/Word	258 training images and 251 test images	English
ICDAR2013 [22]	Horizontal	Character/ Word	229 training images and 233 test images	English
KAIST [25, 26]	Horizontal	Character	Consists of 3000 images	English/Korean
MLT 2017 [29]	Multi-Oriented	Character/Word	7,200 training images, 1,800 validation images, and 9,000 testing images	Multi-lingual (9 languages representing 6 different scripts)
MLT 2019	Multi-Oriented	Character/Word	10,000 training images, 2,000 validation images, and 10,000 testing images	Multi-lingual (10 languages representing 7 different scripts)
Irregular Text				
ICDAR2011 [21]	Horizontal	Word	Developed by ICDAR2003 299 training and 255 test images	English
ICDAR2015 [23]	Multi-oriented	Word	1,000 training images and 500 test images	English
ICDAR2017 [24]	Multi-oriented	Word	7,200 training images, 1,800 validation images, and 9,000 test images	Multi-lingual
MSRA-TD500	Multi-oriented	Text line	300 training images and 200 test images	English/Chinese
Datasets for Text Recognition				
Regular Text				
Char74K	Horizontal	Character	Contains 75,776 characters	English
IIIT 5K-Words [28]	Horizontal	Character/Word	2,000 training images and 3,000 test images	English
SVHN [27]	Horizontal	Character	More than 730000 training and 260000 testing images	Digit
Synthetic Text				
SynthText [33]	Horizontal	Character/Word	Contains 858750 images	Multi-lingual
Datasets for Text Detection and Recognition				
Regular Text				
Street View Text [30]	Horizontal	Word	101 training images and 249 testing images	English
ICDAR2019	Multi-oriented	Word	1,000 training images and 1,000 test images	English
NEOCR	Multi-Oriented	Text Line	Contains 659 images with 5238 text fields	Multi-lingual
COCO-Text [31]	Multi-oriented	Word	43,686 training images for training, 10,000 validation images, and 1,000 test images	Multi-lingual
Total-Text [32]	Multi-oriented	Word	1,255 training images and 300 test images	English
Synthetic Text				
Synth90K [33]	Horizontal	Word	7200000 train images, 900000 test images, and more than 900000 for evaluation	English

Segmentation-based Text Detection models: The regression-based method fails miserably at segmenting text that is close to one another and at identifying curved text. Consequently, a pixel-based method called Progressive Scale Expansion Network (PSENet) [14] was introduced. Any type of text may be efficiently found using the PSENet model. The PSENet model makes use of an incremental expansion approach, which ensures that the positions of the text instances may be accurately determined even when they are close to one another. The selection of the hyperparameters is crucial as it directly affects the text recognition results. However, PSENet needs to re-select the hyperparameters (minimum scale and number of segmentation results) for each dataset.

Tian et al. [12] introduced the Learning Shape-Aware Embedding, or LSAE, model in 2019. An approach for text detection based on instance segmentation is the LSAE model. The pixels are mapped to the encoded feature space so that pixels from the same instance are as near together as possible and pixels from different instances are as far apart as possible in order to distinguish between separate text instances with similar locations. Moreover, in order to address the issue of text line length, the LSAE model generated a shape-aware loss example that could adapt to different shapes and used a novel post-processing technique to provide precise border prediction that distinguished the nearby text samples. When confronted with curved, irregular, or extremely lengthy text, the word-level text identification paradigm (identifying the full word) is challenging to recognize. Therefore, the Beak et al. 2019 model, Character Region Awareness For Text Detection (CRAFT), is suitable. A character-by-character labeled text detection method is the CRAFT model. There are no limitations on the shape of the text because CRAFT detects individual characters; just a limited field of view is needed. In order to identify a group of words, the CRAFT model first identifies individual characters and then determines which characters make up the text. Conversely, the CRAFT model is not able to identify glued characters.

He et al. saw the adoption of a two-stage pipeline that allowed for quicker detection rates while also improving accuracy and reliability. They used a proposed scale-based area to estimate the text positioning. Then, in order to get the best localization accuracy possible, they used a fully convolutional network (FCN). Significant variations in character size have been recognized by Zhu et al. as one of the most important problems with text recognition. Excessively small text makes it harder to read and decreases accuracy. They created a thorough text detector based on a combination of quicker R-CNN features and region

proposal network (RPN) features in order to solve this problem. 2017 saw the introduction of Text Boxes, a quick scene text detector with faster and more accurate performance than previous methods by Liao et al. A 28-layer deep convolutional network is used in this architecture to identify words. The inability of the approach to identify words with a large space between letters and words with less than three characters is one of its shortcomings. Before training the network, character-level label frames must be gathered. This calls for a lot of labeling data, challenging training, and subpar supervision training.

The rotation region proposal networks (RRPN) [37], a rotation-based identification framework for text detection in any direction, was proposed by Ma et al. in 2018. Nevertheless, this method was unable to identify vertical or curved text. Based on data from the higher convolutional layers of the grid on the orientation angle of the TR, the method recommends inclined rectangles. Consequently, multiple orientation text detection is produced. The efficiency of this technique has been increased by developing and fitting a revolutionary RRoI polishing layer to spinning RoI. Yang et al. pioneered multi-dimensional scene text identification in 2018. A novel technique for text line representation is proposed by TextSnake and Long et al. [38]. The TR breaks down into many disks that overlap in an organized manner. PixelLink [39] was influenced by SegLink. Deng et al. in the same year (2018). Links for a specific pixel are labeled as positive if they are in the same instance as nearby pixels; otherwise, they are labeled as negative. In order to depict the text, a linked zone is formed by connecting all pixels that are indicated as positive.

In 2018, Dai et al. introduced fused text segmentation networks (FTSN) for multi-oriented scene text detection. First, they used a RPN to identify and segment text occurrences simultaneously; then, they used NMS to suppress overlapping instances. Lastly, a quadrangle encloses the territory in each case. This method is unable to identify images with vertical text or text at different angles. They dubbed it IncepText [40], a new inception text module that improved PSROI polling to identify text in a variety of orientations and size scales. They used VGG, ResNet 50, and ResNet 101 networks, which lengthened calculation times. Short durations of time cannot be separated into two words with this method. Another disadvantage of the method is that some characters in words with dense backgrounds may not be identified. This approach does not recognize curved or vertical text. According to our findings, this strategy is ineffective for photos with varying text orientations.

A Pixel Aggregation Network (PAN) with a low-cost partition and learnable post-processing was proposed by Wang et al. [13] in 2019. A Feature Fusion Module (FFM) and a Feature Pyramid Enhancement Module (FPEM) make up the segment component of PAN. FPEM is a cascade module with a U-shaped design that provides multilayer information to aid with segmentation. To create final segmentation characteristics, FFM may combine features from FPEMs at different depths. Pixel Aggregation (PA), which properly collects text pixels using anticipated similarity vectors, is used to construct the learnable post-processing. The segment network predicts the text area, kernel, and similarity vector. To increase the accuracy of text recognition, the full-text instance is rebuilt from the anticipated kernel using FPEM + FFM. A segment-based text detection technique's post-processing phase is essential since it transforms the segmentation result into a text box or text area.

Liao et al. proposed a Differentiable Binarization Module Network (DBNet) model to simplify the post-processing stages. DBNet may create adaptive thresholds to optimize network performance. DBNet's differential binarization enhances text identification accuracy without requiring complex post-processing. An accurate multi-oriented scene text localization approach (MOSTL) was presented by Naiemi et al. [41]. The enhanced ReLU layer (i.ReLU) and the enhanced inception layer (i.inception) were included as part of the suggested methodology. Using the proposed framework, low-level visual information is first extracted. Then, another layer was applied to enhance feature extraction. Text detection has been enhanced by the i.ReLU and i.inception layers providing necessary information. Through feeding the output of the i.ReLU and i.inception layers into an additional layer, MOSTL was able to recognize multi-oriented texts, including curved and vertical ones.

Transformer-based Text Detection Models: In computer vision and natural language processing (NLP), text detection is a basic problem with applications ranging from picture interpretation to document analysis. Conventional text identification techniques may not be as flexible with a wide range of data as they frequently depend on manually created characteristics and pre-established criteria. Transformer-based models have become more potent text identification techniques in recent years because of their capacity to extract intricate patterns from massive datasets. These models have been used for a number of text detection tasks, such as document layout analysis [42, 43] and scene text identification. The capacity of transformer-based models to extract contextual information and long-range relationships from text-rich pictures is one of their main benefits when it comes to text identification.

These models were first created for NLP sequence-to-sequence tasks, but they have since been modified and improved for text detection applications. They are able to identify and locate text sections in a variety of situations by using self-attention mechanisms to assess the significance of various parts in a picture. LayoutLM [16], a transformer-based model presented Lee and Osindero [44] and published by Microsoft Research, expands BERT (Bidirectional Encoder Representations from Transformers) to capture layout and text information in documents. Its performance in information extraction and document layout analysis jobs has significantly improved. Rosetta [17], is meant to handle text in photos and movies in different languages.

It uses a vision-language pre-training model to comprehend text in a variety of settings, making it suited for text identification in a wide range of multimedia data. In 2022, Lu et al. [18] introduced a system that includes four essential components—a feature extraction module, boundary refinement module [45], boundary prediction module, and text recognition module—that increase the value of their model. The objective of the feature extraction module is to extract features from input photos for the text detection and identification tasks. Then, because of the different orientations and forms of the texts, the boundary prediction module densely identifies the boundary points but suffers from regressing their exact placements.

A lightweight boundary refinement module is suggested to reduce this impact and provide more accurate boundary points at minimal computational cost. Following the acquisition of the text's boundary points, the characteristics of text instances are immediately sampled and sent into the text recognition module that follows. ResNet series networks are suggested for feature extraction in a text detection model by Zhu and Wang [46], 2022. They concentrate on characteristics from stages C3, C4, and C5, leaving out C2 because of its low-level data. A feature redistribution module is introduced in order to maximize the utilization of multi-level features. The model runs in two concurrent branches: one that extracts text kernel characteristics and the other that generates precise boundary maps. They use a multi-level supervision technique to highlight text kernel

characteristics. Ultimately, a region formulation approach is employed to generate a binary map for text detection based on the text border and text kernel maps.

In 2023, Halder et al. [10] introduced a new transformer-based text detection module, setting a new standard in the field. This innovative technology not only tackles low-light scenarios but also scenes with unpredictable or arbitrary motion. Its activation frame selection framework allows only the selection of a few frames with dissimilarity to enter the detection module, enhancing performance. The transformer is a combination of two components: the similarity detection module and the detection module, working together to achieve the best possible score in record time. This transformative technology transcends time and leaves us in awe of what's possible in text detection.

Comparative Analysis of Different Text Detection Models: The most popular models for text detection are EAST [11] and CTPN [12] based models, and most of the text detection methods use Total-Text, CTW1500, and ICDAR 2015 datasets for experimentation and evaluation. The EAST-based approach leverages a FCN model to predict words or text lines based on pixel-level information swiftly. PVANet, a lightweight feature extraction network architecture, was introduced for real-time object detection. The model's functionality primarily relies on three crucial layers: a "stem" layer for feature extraction, a "feature merging" layer, and an "output" layer for feature processing. The network initiates by employing a sequence of convolutional layers to generate four feature maps from the input image, referred to as the feature extractor stem. Subsequently, a 1 × 1 convolutional operation is applied to these feature maps, followed by a 3 × 3 convolutional operation applied to the output of the previous 1 × 1 convolutional step. The resulting feature volume is utilized for scoring and box office forecasting. Specifically, a 1 × 1 filter with a depth of 1 generates the score map, a 1 × 1 filter with a depth of 5 produces the RBOX (rotated boxes) with four box offsets and a rotation angle, while a 1 × 1 filter with a depth of 8 yields the QUAD (quadrangle with eight offsets).

The CTPN is also introduced for comparative evaluation with the EAST approach. CTPN has three key components: a convolutional layer, a Bi Long Short-Term Memory (LSTM) layer, and a comprehensive connection layer. Notable outcomes from CTPN include its ability to reframe the text detection challenge by seeking fine-grained text proposals. By using an anchor method to forecast the horizontal location of every text proposal and its associated text/non-text score, CTPN achieves exceptional localization accuracy. Additionally, the network's convolutional feature maps include a recurrence mechanism that makes it possible to identify complicated text by considering contextual data from nearby lines. Moreover, all approaches may be combined into one completely trainable model that takes into account the text's sequential structure. There is no longer any need for substantial post-processing because of this coupled model's ability to analyze text at different sizes and in different languages. Using a pre-trained VGG16 backbone, CTPN's approach depends critically on the output of the most recent convolutional maps. The first two convolutional maps have fixed parameters, while the remaining four are trained with predetermined values within the CTPN framework.

Similarly, the differentiable binarization and adaptive scale fusion-based (ASF) methods are also popular for text detection and have achieved great success. These models introduce a novel architectural approach that relies on a stepwise structure. The foundation of this architecture is built upon a feature-pyramid backbone [47], where the initial input image undergoes a series of transformations. The resulting output features are then up-sampled to a consistent scale and channeled into the next module, referred to as ASF. The primary objective of ASF is to generate a contextual feature that serves a dual purpose: predicting both the probability map and the threshold map. The probability map and the threshold map, in turn, play pivotal roles in calculating an approximate binary map. During training, all three maps—probability, threshold, and approximate binary—undergo supervision. Remarkably, the probability map and the approximate binary map share the same supervision, fostering a cohesive learning process. In the inference phase, the architecture enables the extraction of bounding boxes from either the approximate binary map or the probability map. This adaptability empowers the model to detect text instances of varying scales effectively. It is essential to note that features derived from different scales exhibit distinct perceptions and receptive fields.

To harness the full potential of these scale-specific features, conventional semantic segmentation methods often employ feature-pyramid or U-Net structures to fuse them through simple cascading or summation. The novelty of the proposed ASF lies in its dynamic approach to feature fusion across different scales. ASF accomplishes this by concatenating scaled input features and subsequently applying a 3 × 3 convolutional layer [48] to obtain an intermediate feature representation. The next critical step involves calculating attention weights using a spatial attention module. Finally, these attention weights are partitioned along the channel dimension, and weighted multiplication is applied to the corresponding scaled features, yielding a fused feature that effectively leverages information from diverse scales. This dynamic and adaptive feature fusion mechanism enhances the architecture's ability to capture multiscale contextual information for improved text instance recognition and segmentation.

There are models that focus on text detection in video frames, and most of the methods explore the Yolov5 model for addressing the challenges of video text detection. The suggested technique, which is based on a CNN that is only utilized for object detection, employed the YOLOv5 model for text detection. Depending on preset grid sizes, the YOLO method splits the incoming photos into discrete grids. Next, it shows the probability of the desired text at each grid; it can accurately anticipate the text boundaries in a single run, making it useful for real-time text identification. Subsequently, it transfers the identified bounding box to TesseractOCR, a text recognizer, in order to extract the text from the picture. The assignment as a whole is composed of four key components. First, they acquired data. For their tests, they employed three standard datasets: ICDAR2013 [22], ICDAR2015 [23], and YVT. They also conducted a quick comparison with other SOTA approaches. Data annotation comes in second. The bounding box was manually added to each image or frame from the movie using an online annotation tool from Roboflow's official website. They then separated the resulting dataset into train sets, using 80% for training and 20% for testing.

Third, in developing a Deep Neural Network using YOLOv5, in every neural network architecture, features are extracted automatically, but they are done manually in machine learning. The most well-known deep-learning model for training using picture or video frame data is Convolutional Neural Networks (CNN). The main justification for utilizing YOLOv5 is that, in comparison to other state-of-the-art detection models, it is a quick and small model that uses a lot less processing power. YOLOv5 is composed of three main layers: the backbone, the neck, and the final layer. Its backbone is CSPNet (Cross Stage Partial Network), which it employs to extract features from pictures, including forms and edges, and to extract important properties. It also uses a Path Aggregation Network, or PANet, as its neck to enhance information flow. Its purpose is to make multiscale object prediction possible. As the last detection step, the 1 × 1

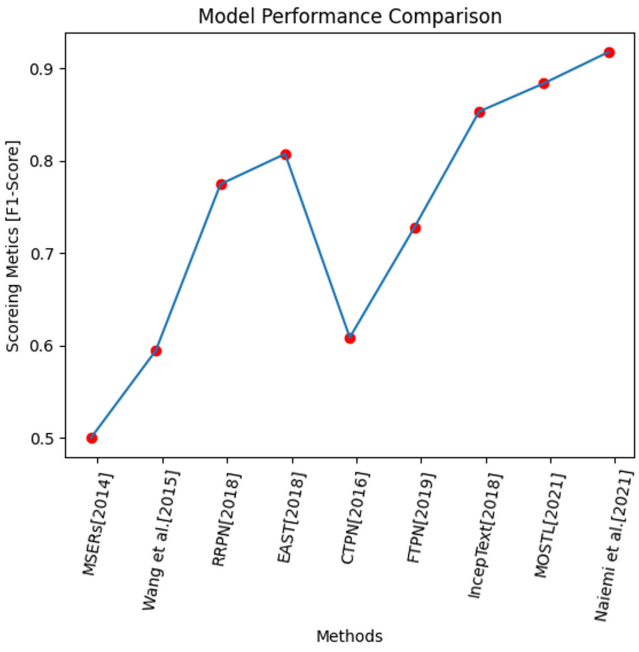
Table 2
Analyzing the text detection results of different methods on different standard datasets

Dataset	Method / Year	MSERs 2014	Wang et al. [49] 2015	RRPN [37] 2018	EAST [11] 2018	CTPN [12] 2016	FTPN [50] 2019	IncepText [40] 2018	MOSTL [51] 2021	Naiemi et al. [6] 2021
ICDAR2013	R	0.5178	0.6011	0.7189	0.7468	0.8336	0.9190	0.9234	0.9250	0.9283
	P	0.5248	0.7721	0.9022	0.9128	0.9311	0.9325	0.9402	0.9427	0.9463
	F	0.5213	0.6760	0.8002	0.8215	0.8797	0.9257	0.9317	0.9338	0.9372
ICDAR2003	R	0.7702	0.7563	0.7483	0.8351	0.8422	0.8664	0.8612	0.8819	0.9101
	P	0.7806	0.7712	0.7496	0.8730	0.8792	0.8920	0.9110	0.9281	0.9308
	F	0.7754	0.7637	0.7489	0.8536	0.8603	0.8790	0.8854	0.9044	0.9203
ICDAR2015	R	0.4940	0.5532	0.7323	0.7833	0.5156	0.7800	0.8060	0.8456	0.9100
	P	0.5063	0.6419	0.8217	0.8327	0.7422	0.6820	0.9050	0.9250	0.9250
	F	0.5001	0.5943	0.7744	0.8072	0.6085	0.7277	0.8530	0.8835	0.9174
MSRA-TD500	R	0.4740	0.5332	0.6831	0.6743	0.6512	0.6800	0.7904	0.8111	0.8640
	P	0.5063	0.6419	0.8219	0.8728	0.8842	0.8520	0.8726	0.8854	0.9034
	F	0.4896	0.5825	0.7461	0.7608	0.7500	0.7563	0.8295	0.8466	0.8833
ICDAR2017	R	0.6625	0.6929	0.7036	0.7296	0.7312	0.7431	0.7412	0.7986	0.8165
	P	0.6743	0.6991	0.7357	0.7812	0.7743	0.7911	0.7945	0.8371	0.8611
	F	0.6683	0.6960	0.7193	0.7545	0.7521	0.7663	0.7669	0.8174	0.8382
ICDAR2019	R	0.4571	0.4752	0.5619	0.5647	0.5914	0.6418	0.6521	0.6717	0.6908
	P	0.4912	0.4801	0.5849	0.5904	0.6311	0.6573	0.6611	0.6948	0.7119
	F	0.4735	0.4776	0.5732	0.5773	0.6106	0.6495	0.6566	0.6830	0.7012

convolutional layer known as the YOLO layer serves as the head of the final layer. Subsequently, the identified word will undergo Optical Character Recognition, or OCR, in order to get the identified text. They employed TesseractOCR, an OCR that consists of sequential pictures, adaptive binary thresholding, character recognition, character aggregation to create words, and evaluation of the linked components and their relationships.

The results of different methods on different standard datasets are listed in Table 2, whereas for the complex datasets, such as ICDAR 2017 and ICDAR 2019, almost all the methods report poor results compared to the results on other simple datasets. This shows that there is a scope for improving the results on complex datasets. At the same time, the models are advanced in addressing challenges of text detection, and the performance of the method improves, as shown in Figure 4, where it is noted that as the year passes, the performance of the methods improves. The same conclusions can be drawn from Tables 3–9, and Figures 5–11 on different datasets. With the above analysis, one can conclude that most of the challenges of simple datasets have been addressed while for the complex datasets, the methods have not achieved significant progress. Furthermore, when we look at the performance of all the methods on all the datasets, none of the methods achieve consistent results. For different datasets, different methods report the best results. This is the limitation of the present methods.

Figure 4
Detection comparison analysis over changing of time and method on ICDAR2015



4.2.2. Recognition methods

The popular deep-learning-based text recognition systems now in use will be categorized and summarized in this section. Depending on how they are implemented, six classes comprise text recognition techniques based on segmentation, Transformer, Encoder-Decoder and attention, Connectionist Temporary Classification (CTC), and other methods.

In order to address image-based sequence identification issues, particularly scene text recognition issues, Shi et al. introduced the Convolutional Recurrent Neural Network (CRNN) text recognition method. It integrates the loss functions of CTC, RNN, and Convolutional Neural Network (CNN). In order to improve context modeling, the CRNN model was extended to incorporate LSTM [36], and end-to-end indefinite sequence recognition was accomplished using the CTC loss function.

Table 3
On the CTW1500, a comparison of several text detection models

Type	Model	Year	Precision	Recall	*F*1-score
Regression	CTPN [12]	2016	60.4	53.8	56.9
Regression	EAST [11]	2017	78.7	49.1	60.4
Regression	SegLink	2017	42.3	40.0	40.8
Regression	TLOC [52]	2019	77.4	69.8	73.4
Regression	TextMountain [41]	2021	82.9	83.4	83.2
Segmentation	TextSnake [38]	2018	67.9	85.3	75.6
Segmentation	DBNet	2019	86.9	80.2	83.4
Segmentation	PAN [13]	2019	86.4	81.2	83.7
Segmentation	PSENet [14]	2019	82.5	79.9	81.2
Segmentation	SAE [53]	2019	82.7	77.8	80.1
Segmentation	FCENet [15]	2021	87.6	83.4	85.5

Table 4
On Total-Text, a comparison of several text detection models

Type	Model	Year	Precision	Recall	*F*1-score
Segmentation	Mask TextSpotter	2018	82.5	75.6	78.6
Segmentation	TextSnake [38]	2018	82.7	74.5	78.4
Segmentation	PAN [13]	2019	89.2	81.0	85.0
Segmentation	TextField	2019	81.2	79.9	80.6
Segmentation	LOMO [54]	2019	87.6	79.3	83.3
Segmentation	CRAFT	2019	87.6	79.9	83.6
Segmentation	PSENet [14]	2019	84.0	78.0	80.9
Segmentation	FCENet [15]	2021	89.3	82.5	85.8
Regression	ATRR	2019	80.9	76.2	78.5
Regression	CSE	2019	81.4	79.1	80.2
Regression	TextMountain [41]	2021	88.5	84.1	86.3

Table 5
Comparing the ICDAR 2015 dataset with a number of text detection models

Type	Model	Year	Precision	Recall	*F*1-score
Regression	CTPN [12]	2016	74.0	52.0	61.0
Regression	SegLink	2017	73.1	76.8	75.0
Regression	EAST [11]	2017	83.6	78.3	78.2
Regression	SSTD	2017	80.2	73.9	76.9
Regression	WordSup [55]	2017	79.3	77	78.2
Regression	TextBoxes++ [5]	2018	87.2	76.7	81.7
Regression	RRD [56]	2018	85.6	79	82.2
Regression	MCN	2018	72	80	76
Regression	SBD	2019	92.1	88.2	90.1
Regression	SPCNet [57]	2019	88.7	85.8	87.2
Segmentation	Lyu et al. [58]	2018	94.1	70.7	80.7
Segmentation	TextSnake [38]	2018	84.9	80.4	82.6
Segmentation	LOMO [54]	2019	91.3	83.5	87.2
Segmentation	SAE [53]	2019	85.1	84.5	84.8
Segmentation	CRAFT	2019	89.8	84.3	86.9
Segmentation	DBNet	2019	91.8	83.2	87.3
Segmentation	PAN [13]	2019	84.0	81.9	82.9
Segmentation	PSENet [14]	2019	88.7	85.5	87.1
Segmentation	FCENet [15]	2021	90.1	82.6	86.2

Table 6
Different text detection models comparing them on MSRA-TD500

Type	Model	Year	Precision	Recall	*F*1-score
Regression	EAST [11]	2017	75.3	87.3	76.1
Regression	SegLink	2017	86.0	70.0	77.0
Regression	DeepReg	2017	77	70	74
Regression	RRD [56]	2018	87	73	79
Regression	MCN	2018	88	79	83
Segmentation	He et al. [59]	2016	71	61	69
Segmentation	RRPN [37]	2018	82	68	74
Segmentation	PixelLink [39]	2018	83	73.2	77.8
Segmentation	Lyu et al. [58]	2018	87.6	76.2	81.5
Segmentation	TextSnake [38]	2018	83.2	73.9	78.3
Segmentation	Xue et al.	2018	83.0	77.4	80.1
Segmentation	MSR [59]	2019	87.4	76.7	81.7
Segmentation	SAE [53]	2019	84.2	81.7	82.9
Segmentation	CRAFT	2019	88.2	78.2	82.9
Segmentation	DBNet	2019	91.5	79.2	84.9
Segmentation	PAN [13]	2019	84.4	83.8	84.1

Table 7
A comparison of different models for text detection on MLT-2019

Type	Model	Year	Precision	Recall	*F*1-score
Segmentation	PSENet [14]	2019	73.5	59.6	65.8
Segmentation	CRAFT	2019	79.5	59.6	68.1
Segmentation	DBNet	2019	78.3	64.0	70.4

Table 8
Evaluating precision on a normative recognition dataset

Type	Model	Year	IIIT-5k	SVT	ICDAR2013
CTC	CRNN	2016	78.3	80.6	86.8
Encoder-Decoder & Attention	RARE	2017	79.8	81.7	87.1
Encoder-Decoder & Attention	ASTER	2018	90.1	93.3	91.4
Encoder-Decoder & Attention	DAN	2020	94.1	89.1	93.6
Transformer	NRTR [60]	2019	90.3	91.2	95.9
Transformer	MASTER [61]	2021	95.2	90.8	95.4
End-To-End	STN-OCR [57]	2017	86.2	79.9	90.7
Transformer	MGP-STR [62]	2021	96.4	94.7	96.4
Transformer	LevOCR [63]	2020	96.6	92.8	96.8

Table 9
Comparing detection accuracy on irregular dataset

Type	Model	Year	SVT-P	CUTE80	ICDAR2015
Encoder-Decoder & Attention	ASTER	2018	90.1	93.3	91.4
Encoder-Decoder & Attention	DAN	2020	80.2	84.5	74.5
Transformer	NRTR [60]	2019	94.9	80.7	79.5
Transformer	SRN [64]	2020	85.0	87.8	82.7
Transformer	MASTER [61]	2021	84.2	87.3	79.3
Transformer	MGP-STR [62]	2021	91.0	90.3	87.3
Transformer	LevOCR [63]	2020	88.1	91.7	86.4

Figure 5
Detection comparison analysis over changing of time and method on CTW1500

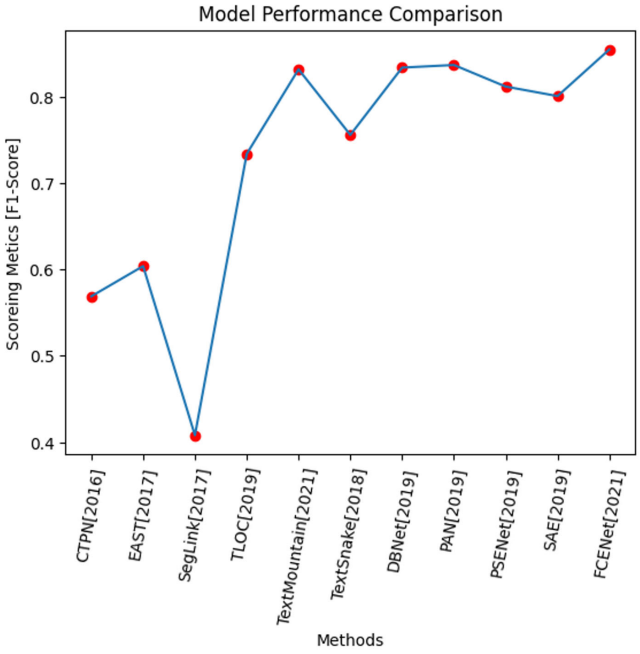


Figure 6
Detection comparison analysis over changing of time and method on Total-Text

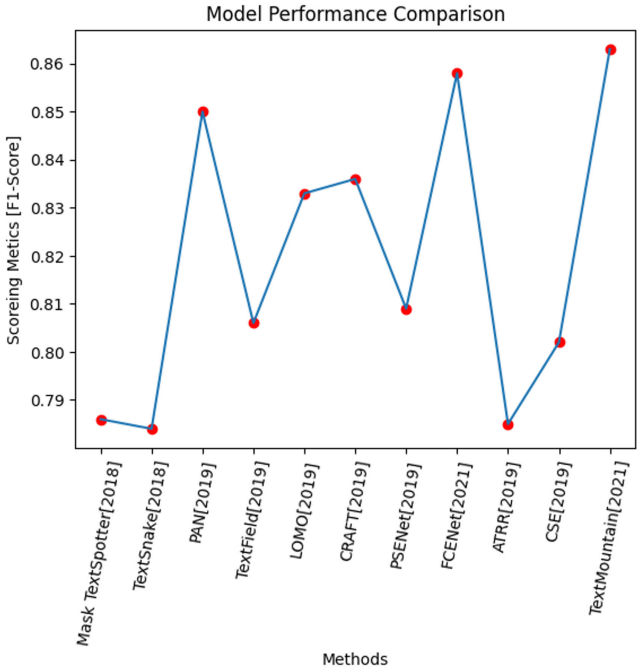


Figure 7
Detection comparison analysis over changing of time and method on ICDAR2015

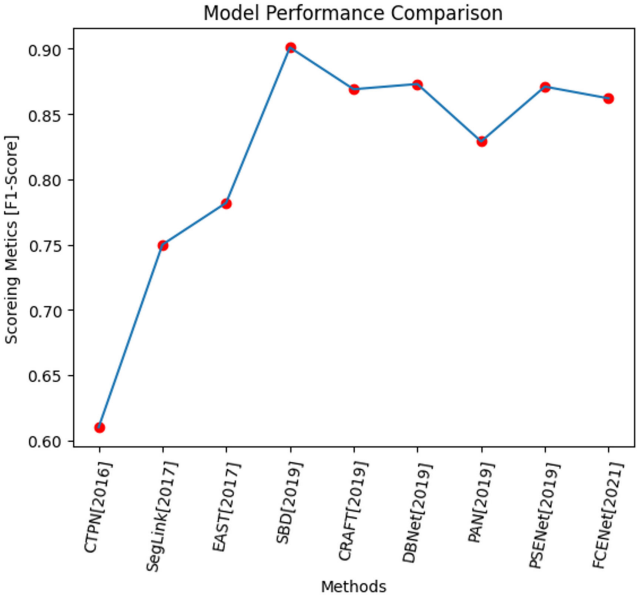
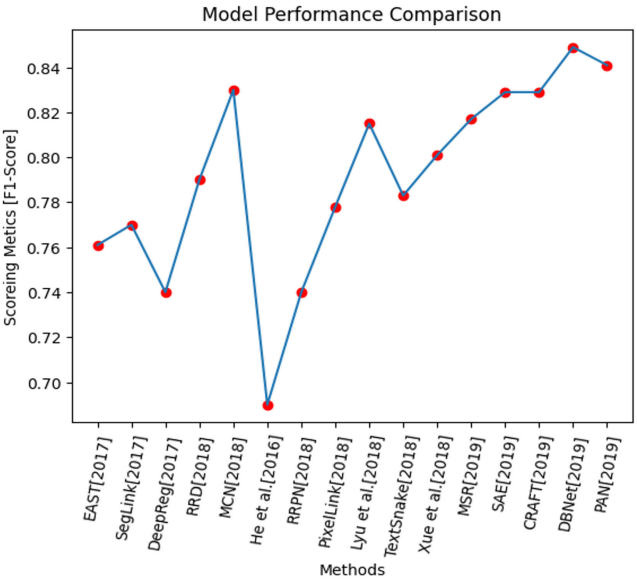


Figure 8
Detection comparison analysis over changing of time and method on MSRA-TD500



The CRNN technique is one of the most popular text recognition frameworks since it just needs basic word-level labels and input pictures to complete model training. Typically, the CTC process is utilized during the prediction step. CTC accumulates conditional probabilities to transform the output features of CNN [65] or RNN [66] into string sequences. By guaranteeing that the anticipated text sequence matches the real text sequence in both length and order, text recognition technology applications can solve the temporal text alignment challenge. Several researchers have looked at the improved method because the CRNN model has shown promising performance in text recognition. Gao et al. replaced LSTM with CNN convolution, which has fewer parameters and comparable accuracy performance. Additionally, Facebook unveiled Rosetta, an improved CTC-based text recognition technology. English datasets show that the Rosetta

Figure 9
Detection comparison analysis over changing of time and method on MLT-2019

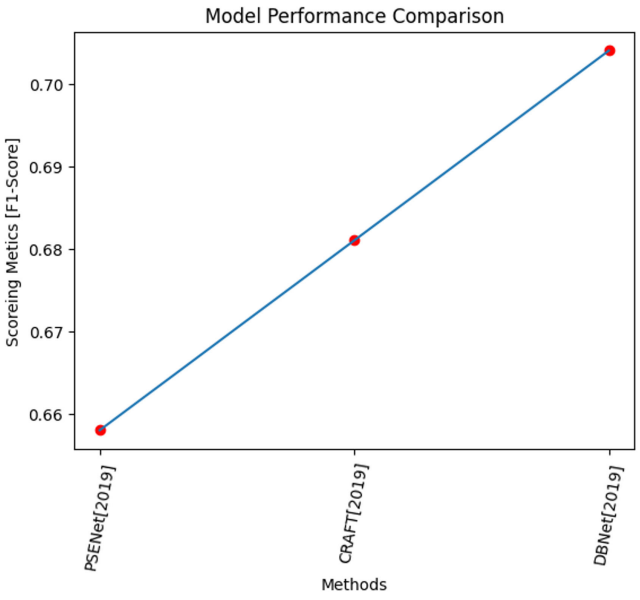


Figure 10
Detection comparison analysis over changing of time and method on regular text dataset

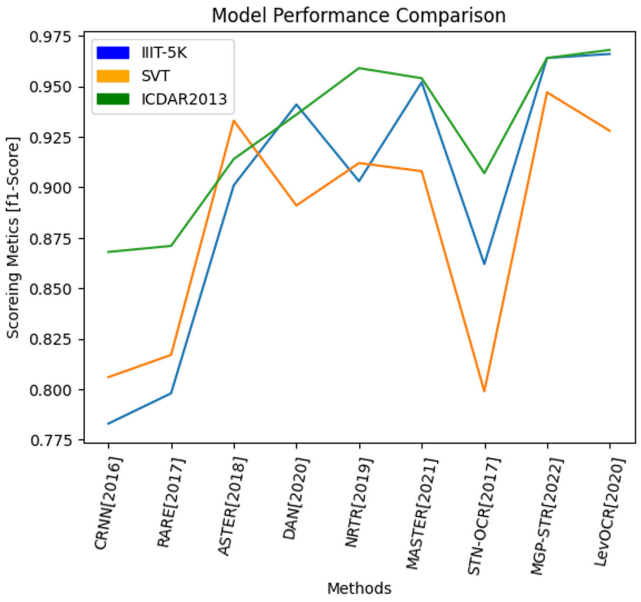
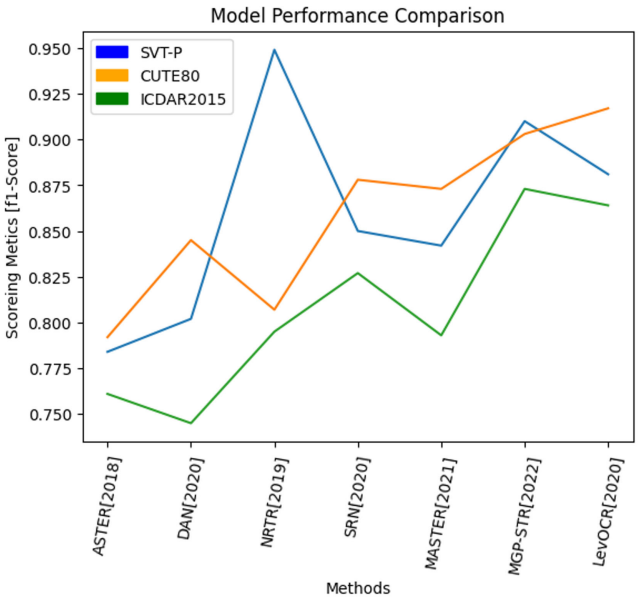


Figure 11
Detection comparison analysis over changing of time and method on irregular text dataset



model, which consists of a CTC and a full convolution network, works admirably. The preceding CTC-based algorithms perform well on normal text; however, due to network design restrictions, these systems struggle to tackle the recognition job of curved and rotated irregular text.

Encoder-Decoder and Attention-Based Text Recognition: The encoder, decoder, and attention methods serve as the foundation for text recognition systems [66]. Because text recognition algorithms based on CTC are unable to tackle text recognition problems before and after them due to the recognized text including special semantic information, researchers attempt to overcome the dependence problem using an encoder-decoder. A traditional Sequence2Sequence architecture is the encoder-decoder architecture. Machine translation was the first use of the encoder-decoder system. It takes a sequence as input and returns a sequence as output. The encoder-decoder structure was recently developed by vision researchers with promising results. The CRNN and other text recognition algorithms may only identify normal text files containing characters that are aligned horizontally. It cannot be accurately recognized in cases of inconsistent text recognition. Researchers began investigating irregular text recognition [67] by using an encoder-decoder for text recognition.

Uneven text collections sometimes include non-horizontal material with problems such as bending, occlusion, and blurring. Before recognizing irregular text, it is sometimes necessary to map it into horizontally ordered text using the rectification module. In 2017, Cheng et al. [68] suggested a focused attention network (FAN) for scene text detection in difficult and low-quality photos. The attention network (AN) and the focusing network (FN) comprise the FAN. Cheng et al. [68] described an arbitrary text recognition (AON) system in 2018 [69]. In order to create letter sequences, this method directly exploits the profound properties of irregular texts in an attention-based decoder. End-to-end networks may be trained with pictures and word-level annotations. For scene text recognition, Bai et al. developed edit probability, or EP. The objective of the EP is to ascertain the chance that absent or superfluous characters would surface in a series of output sequences derived from the conditional probability distribution corresponding to the input picture.

As a result, the misalignment issue is mitigated and the training process may concentrate on characters that are missing, duplicate, or unidentified. For scene text recognition, the authors Bai et al. [70] developed edit probability, or EP. The objective of the EP is to ascertain the chance that absent or superfluous characters would surface in a series of output sequences derived from the conditional probability distribution corresponding to the input picture. As a result, the misalignment issue is mitigated and the

training process may concentrate on characters that are missing, duplicate, or unidentified. A robust text recognizer that automatically corrects irregular text is called Robust Text Recognizer with Automatic Rectification (RARE).

The network is split into two sections: the Spatial Transformer Network (STN), a network for spatial transformation, and Sequence2Squence, an encoder-decoder network. The Thin-Plate-Spline (TPS) irregular text picture is corrected, and the correction module STN converts it into a horizontal image. A sequence recognition network is then created by decoding the image. Correction-based strategies improve migration. Methods based on corrections are very adaptable. Apart from RARE's text recognition approach, SpaTial Attention Residue Network (STAR-Net) adds the correction module to CTC-based algorithms and outperforms the conventional CRNN model.

Wang et al. [62], in their modified RNN [66] assigned greater weight to the target data and related data, enabling the decoder to focus its "attention" on the target data, gaining more details and generating a reasonable vector representation of a longer input sequence. The accuracy of irregular text recognition is significantly improved with the addition of an attention mechanism. The Recursive Recurrent Nets with Attention Modeling (R2AM) [16, 44] technique was the first to apply attention to the field of text recognition, coming after the advent of RARE-based correction algorithms. This model first obtains the encoded picture characteristics from the input image using the recursive convolutional layer.

The techniques get around the issues with traditional recognition methods [71–73] by using transformers. Transformer's rapid progress has led to strong results in both text categorization and visual task recognition. For example, in the rule text recognition section, the Transformer structure replaces additional LSTM context modeling modules and focuses on global information in the feature extractor to solve CNN's limitations in long-dependency modeling. Using a complete Transformer structure, the No-Recurrence Sequence-to-Sequence Text Recognizer (NRTR) [60] technique encoded and decoded the input photos. Using a foundation layer for text recognition, the NRTR approach gathered features and confirmed the Transformer structure. Yu et al. introduced Semantic Reasoning Networks (SRN), a trainable framework approach that operates from end to end in 2020. The SRN technique consists of a backbone network, a Parallel Visual Attention Module (PVA recommended parallel attention module), a Global Semantic Reasoning Module (GSRM), and a Visual Semantic Fusion Decoder (VSFD). By using the reading order as a query, the SRN technique may generate the aligned visual characteristics for all time steps in parallel, so making the computation time independent.

The SRN technique employs the Transformer encoder as the semantic module to integrate the picture's visual and semantic information, improving opaque, blur, and other irregular text detection. Even while the attention-based technique is capable of learning the internal representation of one-dimensional or one-dimensional features, it suffers from attention drift. A text recognition model initially has enough attention to concentrate on the TR when it scans the full image. But as the scan advances to the right, the focus progressively becomes skewed toward the background and other non-TRs, which degrades the model's accuracy and increases its failure rate. We call this phenomenon attention drift. Attention drift is a problem with the ASTER network structure diagram, for example, when letters in many words are identical or when there is no visible gap between consecutive characters.

The model is biased to spend attention on earlier characters while scanning pictures, especially in sequence learning, which causes the model to miss the subsequent letters or phrases. This is one reason for attention drift. The model may have trouble focusing on the middle letter or word when it detects a line of text if its initial focus is on the leftmost portion of the line. Several scholars have put forth several remedies to address the problem of attention-wandering. One way to identify drift phenomena is through the use of bidirectional RNNs in sequence learning.

Using a neural network, this technique keeps the attention focused on the TR by utilizing both forward and backward context information. Alternatively, the attention mechanism [74, 75] may be used with a text localization box to train the framework to identify text signals and more precisely predict text locations. The attention-based method is unable to perform parallel computing efficiently under the RNN architecture [76]. As a result, Lu et al. [61], In 2021, the Multi-Aspect non-local network for irregular Scene Text Recognition (MASTER) model was introduced, This has a global attention system included. The global contextual attention mechanism and multi-aspect-based encoder and the Transformer-based decoder constitute the foundation of the MASTER [61] model. Transformer-based text recognition models might improve text recognition accuracy by making full use of self-attention's advantages to better understand semantic information.

Segmentation-based techniques use an additional mechanism for text recognition in addition to the Transformer and attention-based recognition [61, 77] approaches. Characters inside a text line are recognized as separate entities using segmentation-based text recognition. Segmented characters are simpler to read than characters that have been fixed as a whole text line at a time. To achieve recognition results, the segmentation-based text recognition [78–80] approach locates each character in the input text picture using a character classifier. It reduces a big global problem to a local challenge, which works well in irregular text settings. However, this approach requires labeling each character individually, which is challenging to acquire data for. An instance segmentation model for word recognition was proposed by Lyu et al. [18]. It identified the text bounding box's corner and used an FCN-based approach to segment the TR in the relative location. This allowed the model to recognize text in the scene. Character Attention Fully Convolutional Network (CA-FCN) was developed by Wang et al. [49] with a new perspective on text recognition [81]. This approach produces superior localization results for both regular and irregular texts [82, 83] when the text is curved or significantly deformed. The summary of the results of different recognition models on different standard datasets is reported in Table 10 and Figure 12, where it is noted that when the complexity of the dataset increases, the performance of methods degrades. In the same way, as the methods advance, the results of recognition improve. Thus, it can be said that for all of the datasets shown in Table 9, none of the approaches produces the best and most consistent results. As a result, it is difficult to create a model that would work consistently over a range of datasets with varying levels of complexity.

4.3. Models for text detection and recognition in video

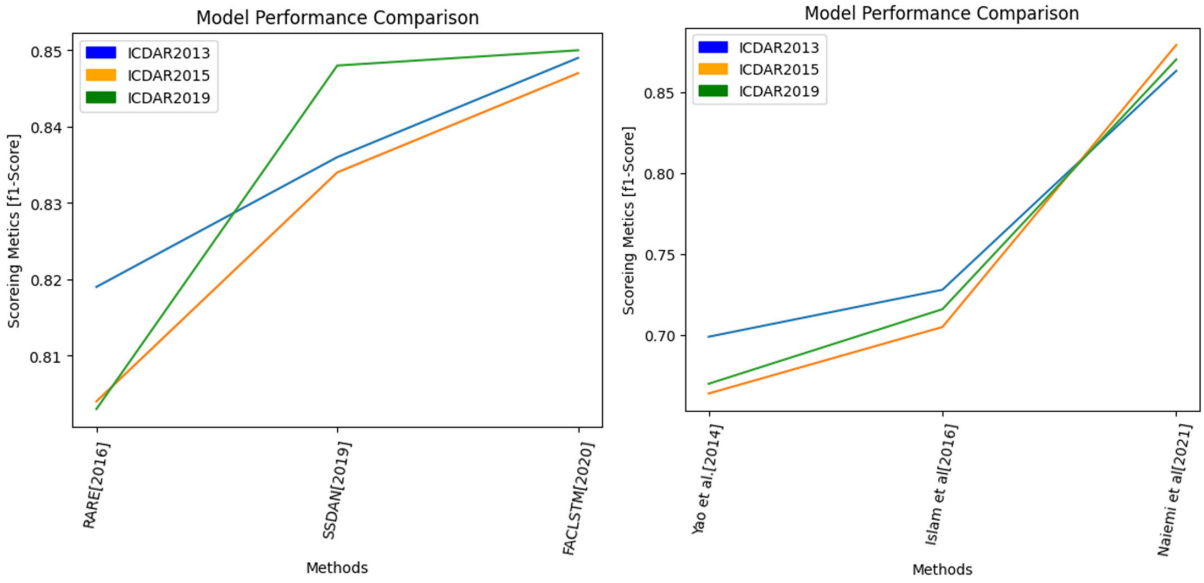
4.3.1. Detection models

An end-to-end technique for identifying text watermarks in videos was examined in 2022 by Banerjee et al. [85]. The model can recognize captions, scenes, and text watermarks in videos. To mitigate the effect of low contrast and complicated backgrounds, the method combines UNet3+ and Fourier contour embedding.

Table 10
Analyzing text recognition results

Method	Year	Task	ICDAR2013			ICDAR2015			ICDAR2019		
			R	P	F	R	P	F	R	P	F
Text Recognition											
SSDAN	2019	Reco.	0.779	0.901	0.836	0.783	0.892	0.834	0.783	0.923	0.848
FACLSTM [84]	2020	Reco.	0.795	0.912	0.849	0.799	0.901	0.847	0.790	0.921	0.850
RARE	2016	Reco.	0.761	0.887	0.819	0.754	0.862	0.804	0.742	0.875	0.803
End-to-End Text Spotting											
Islam et al. [7]	2016	End-to-End	0.658	0.813	0.728	0.641	0.783	0.705	0.638	0.817	0.716
Yao et al.	2014	End-to-End	0.637	0.775	0.699	0.598	0.746	0.664	0.603	0.753	0.670
Naiemi et al. [6]	2021	End-to-End	0.809	0.924	0.863	0.830	0.934	0.879	0.811	0.938	0.870

Figure 12
Recognition model performance over time



Anto Bennet et al. [86] created a deep-learning-based method for identifying Telugu text in videos that was released in the same year, 2022. The model uses a convolutional neural network to encode language-specific data in order to generate the results. The focus of Nandanwar et al. [86] was on the challenges associated with text recognition in 3D video. The model combines deep learning and the wavefront idea to tackle the problem. Furthermore, before combining wavefront and deep-learning models, dominating point identification was accomplished using generalized gradient vector flow. Chen et al. used parametric shape regression, propagation, and fusion for text detection in movies. The relationship between the intra-frame and inter-frames is used to refine the text candidate features. The Yolov5 and TesseractOCR combo was proposed by Chaitra et al. to identify and recognize text in video frames. Nevertheless, TesseractOCR's sensitivity to low contrast makes the method effective for high-quality photos. In conclusion, the methods leverage temporal information to enhance the performance of text recognition and detection. The techniques' applicability is restricted to daytime photos; nighttime videos are not supported. It follows that the models cannot be useful for daytime nighttime video taken by both shaky and non-shaky cameras when the approaches are unable to handle nighttime video. In 2023, Halder et al. [10] introduced a new transformer-based text detection module, setting a new standard in the field. This innovative technology not only tackles low-light scenarios [72, 85, 87] but also scenes with unpredictable or arbitrary motion. Its activation frame selection framework allows only to selection of a few frames with dissimilarity to enter the detection module, enhancing performance. The transformer is a combination of two components, the similarity detection module and the detection module, working together to achieve the best score possible score in record time. This transformative technology transcends time and leaves us in awe of what's possible in text detection.

Techniques like fusion, propagation, and parametric shape regression are popular and produce the best results for video text detection. The paper proposed a novel cross-frame text cues propagation and fusion procedure based on a parametric text shape representation and regression model. Unlike most previous approaches for text detection in videos, which fused text cues in multiple frames by tracking or aggregating features at the frame level, these approaches carefully propagate and fuse the features and shape parameters of individual text candidates across

Table 11
Accuracy comparison on ICDAR2015 video dataset advancements in text in video datasets

Type	Methods	Year	Precision	Recall	*F*1-score
Regression	East [11]	2017	55.4	40.0	46.4
Segmentation	PSENet [14]	2019	78.3	75.7	76.9
Regression	YOLOv5s	2020	61.0	46.0	52.44
Transformer	Halder et al. [10]	2023	80.4	77.8	79.0

neighboring frames, which enhances the text detection performance of the method on videos. Their suggested end-to-end trainable video text detection network featured an attentive text feature and parameter fusion module, an accurate TR regression module, and an efficient R-CNN-based cross-frame text area propagation module. The suggested detection network outperforms single-frame detection in video text detection by combining text signals from several frames.

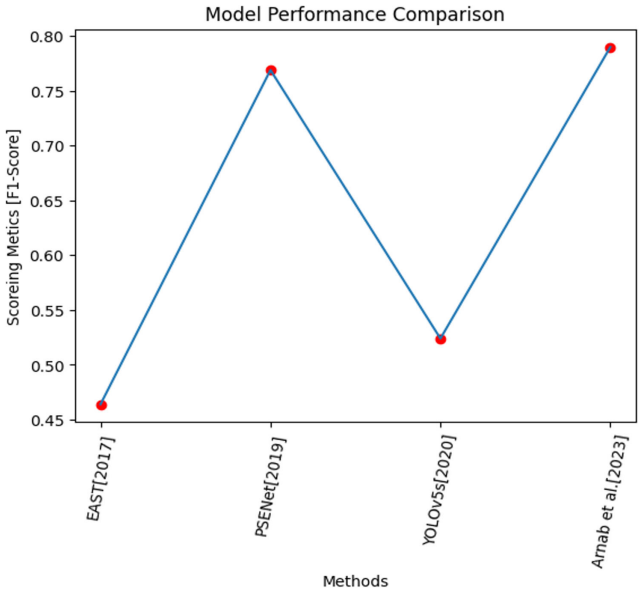
The suggested text detection model takes a video frame, extracts its text areas, and uses shape parameter regression to merge those regions with revised TR candidates that were transmitted from the previous frame. Careful fusing of the regression and previous parameter values further refines the form parameters of a text contender. The whole system, including text identification in video frames and localizing text in a single video frame, may be split into three primary modules using a polynomial-based parametric representation and regression approach for text areas. The global form properties of text are efficiently captured by these modules. To identify text candidates in a single video frame, they regress the shape parameters of a TR using an R-CNN-based network, such as PolyPRNet. The network combines ResNet50 and Feature Pyramid Network (FPN) for feature extraction; it combines RPN with RoIAlign operation for TR suggestions and associated feature maps.

After a Cascade R-CNN-based regression branch has been used to tighten its bounding box and compute its text/non-text score, a shape regression branch consisting of three convolutional layers and two full-connected layers is employed to predict the shape parameters of potential text candidates. An R-CNN model is used to simulate object tracking since the moving objects are tiny and move quickly between frames. They also present an easy-to-use method for text area candidates to spread over neighboring frames using bounding box regression [88] in the text recognition network. They propose to propagate the shape parameters and features of the text candidate in the previous frame to the current frame as supplementary text cues, which are used to refine the corresponding text candidate in the frame alternatively, since the appearance of a text instance typically varies quite slightly between two frames, and the TR propagation mechanism based on the R-CNN detection framework reveals the correlation between two corresponding text candidates in two adjacent frames. Using the ICDAR 2015 standard dataset, Table 11 and Figure 13 present a comparative analysis of the video text identification methods. In this investigation, the transformer-based model performs better than other deep-learning models. The findings indicate that there hasn't been much development and that different techniques function differently over time. This is the point where scene text detection datasets and video behave differently. Thus, it may be said that text detection in video is more challenging than word identification in scene images. Consequently, it is required to develop a single model that can be applied to both scene and video datasets.

4.3.2. Recognition models

Deep-learning-based text recognition techniques used to be straightforward, categorizing words directly as separate categories.

Figure 13
Detection model performance over time



Two popular frameworks for text recognition arose as deep learning progressed: the CTC-based approach (CRNN) and encoder-decoder techniques that combine attention processes. While decoders are different, both employ CNNs to extract features from text pictures. The most widely used model for scene text recognition is CRNN, which was the first to combine CNN, RNN, and CTC. CTC-based techniques are commonly employed in Chinese recognition applications because, while attention-based recognition performs well for English recognition, its influence on Chinese recognition is unreliable. In 2022, Feng et al. [89] Proposed a model where the text detection process involved a YOLOv3-based detector with a vertical anchor mechanism from CTPN, modifying the regression target from text lines to fixed-width text sub-lines. Bounding boxes were obtained using CTPN's text line construction methods. Recognition was performed using a CRNN model with two batch normalization layers. A post-processing method was proposed for correct recognition output. In the same year, Placidi et al. [90] proposed a system for text normalization, feature extraction, semantics, fusion, and encoder-decoder architecture. It uses a thin-plate spline spatial transformation network for image normalization, a ResNet module for feature extraction, an object detection network for semantics, a fusion of semantic vector and feature map, and an encoder-decoder transformer architecture for character predictions. The system incorporates inter-sequence contextual information and predicts characters based on previous predictions, ensuring accurate and efficient text processing. In 2023, Shuai et al. bring a new model, named CLIP4STR [91], it is

a dual encoder-decoder framework that utilizes CLIP and PSM for scene text recognition tasks. The text and image encoders utilize the architectures and pre-trained weights from CLIP, and the decoder framework adopts the design of the transformer decoder and PSM technique. The text encoder is partially frozen, and the visual branch is fully trainable with a cross-modal branch acting as a semantic-aware spell checker. The decoder aims to extract character information from visual or cross-modal features using the transformer decoder and PSM technique. In 2022, Weijia proposed a Transformer model. The TransDETR architecture [92] is a video text spotting method that uses a direct sequence prediction problem, with each text query predicting the entire text instance trajectory and corresponding text content. The pipeline consists of a backbone, transformer-based encoder and decoder network [88], and an attention-based recognition head with Rotated RoI. The backbone extracts the pixel-level feature sequence of the input video clip, while the transformer decoder decodes top pixel features representing text instances. The recognition head is designed for end-to-end training and consists of a starter and decoder.

The Single-Point Text Spotter v2 (SPTS v2), an improved text spotting model built on Transformer architecture, was suggested by Liu et al. [93] in 2023. The Parallel Recognition Decoder (PRD) and the Instance Assignment Decoder (IAD) are the two primary parts. For every text instance, the IAD predicts auto-regressively indicators, and the PRD permits simultaneous prediction for matching text recognition outcomes. The bounding box is simplified, allowing for simpler detection and recognition. SPTS v2 separates different text instances and uses information transmission to pass on recognition tokens' information. In the same year, Wu et al. [94] proposed the TRAN method, which is a text recognition model that rectifies irregular texts at both geometry and pixel levels using a two-level recognition network (TORN). To correct texts, it makes use of a Geometry-Level Rectification Network and a Pixel-Level Rectification Network. To enhance text-feature extraction, the model also includes a novel Channel-Wise and Kernel-Wise Attention Unit (CKUnit). It employs a Skip Training technique to bring subnetworks together during training and an Attention-Based Recognition Network (ABRN) for sequence recognition. Li et al. presented a scene text recognition model using a dual relation module in the feature extraction stage in the same year, 2023. Three phases make up the model: final decoding, feature extraction, and image rectification. The Local Visual Branch and the Long-Range Contextual Branch are combined into a single network module by the Dual Relation Block. The authors implement the Dual Relation Network (DRNet) for scene text recognition by replacing ResNet blocks in the feature extraction stage with the proposed DR block and removing the contextual stage. In the same year, Shivkumara et al. [95] proposed a language-independent text detection and style transfer system for social media images. It uses EffiUNet++, a text detection model based on EfficientNet and UNet++ architecture, and a Differential Binarization module. The system uses the TESP-Net generative model for text style transfer, which incorporates self-attention feature maps for multilingual ability. The system uses image inpainting to fit the generated target character without losing shape, color, structure, and visibility.

4.4. Models for text classification

In the year 2023, Raja et al. [96] introduced the first dataset for fake news detection in four Dravidian languages—Telugu, Tamil, Kannada, and Malayalam—consisting of approximately 26,000 news articles. They proposed a novel adaptive learning method to fine-tune a multilingual pre-trained transformer model, combining English and Dravidian fake news datasets for enhanced performance. The study employed transfer learning to assess the effectiveness of the fine-tuned models on the Dravidian languages dataset. Their approach demonstrated significant improvements in detecting fake news, providing a critical resource for combating misinformation in these underrepresented languages.

In order to detect false news in Dravidian languages, Raja et al. [97] presented a novel hybrid model in this study that combines a BiLSTM with a multiscale residual CNN network. They improved the model's capacity to identify subtle patterns in text by using a multiscale feature extraction technique intended to capture both local and long-range textual relationships. They used training optimization approaches and gradient clipping to reduce overfitting and enhance model performance in order to handle problems like bursting gradients. This method outperforms state-of-the-art models, as shown by their experimental research on the Dravidian_Fake dataset, demonstrating its resilience and efficacy.

In the same year, Raja et al. [98] made another noteworthy contribution when they created a hybrid deep-learning architecture to address the problem of detecting fake news in Dravidian languages. This architecture incorporates DTCN, BiLSTM, and CAM. The approach included a contextualized attention mechanism that increased the detectability of bogus news while simultaneously improving interpretability. To hasten model convergence, they also presented an adaptive-based cycle learning rate technique with an early halting mechanism. Based on studies conducted on the Dravidian_Fake dataset, their suggested hybrid model outperformed baseline and state-of-the-art methods, indicating a significant breakthrough in the field.

5. Future Challenges/Applications/Directions

5.1. Challenges

In the future, the detection and recognition of multi-oriented text in photos and videos will have countless real-world applications. Here are a few examples of possible applications:

1) Multi-oriented text detection enhances autonomous cars' capacity to perceive their surroundings more thoroughly. Detecting text on traffic signs from various angles, for example, can improve the safety and reliability of self-driving technology.
2) Augmented Reality (AR): In order to give valuable context to users, AR systems frequently require multi-oriented text detection and identification. For example, recognizing restaurant signage and delivering reviews and menu information.
3) Document Analysis: Extracting text data from scanned documents, even ones containing text in many orientations, can help businesses run more efficiently. It might lead to breakthroughs in fields such as digital archiving, data mining, and knowledge extraction from available data.
4) Robotics: This technology might help robots navigate and interact with their surroundings more efficiently. Inventory management robots, for example, can scan parcel labels in a variety of orientations.
5) Video Content Analysis: This technology can automatically extract and catalog text data from videos, which may then be utilized for a variety of applications like improved searchability, automated subtitling, and video summarizing.
6) Smart City Applications: Detection and recognition of multi-oriented text may be crucial in monitoring and maintaining urban infrastructure, such as recognizing and interpreting signs for repair, traffic control, or surveillance.

7) Assistive Technology: This type of technology may be extremely valuable to visually impaired persons. They may utilize it to engage with the world more readily, recognizing and understanding the text in real time and educating them about their surroundings.

However, it is critical to emphasize that as these applications get more advanced, greater emphasis must be placed on privacy and ethical issues to guarantee that the use of text detection and recognition technologies respects individuals' rights and freedoms.

Providing a system that can recognize texts in various orientations—especially curved and vertical texts—is essential in order to detect and recognize curved or vertically flipped letters in natural scene photographs. Although there are many languages spoken in the globe, scene text detection and identification have only advanced in a select few, namely English. So, multi-oriented and multi-language detection and recognition systems will play a crucial role in solving some major problems in the field of Computer Vision and pattern recognition in the future.

5.2. Applications

Text detection and identification in videos and photos of natural scenes can be extended to automatic driving vehicles, where detection and recognition methods should focus on number plates for tracing vehicles, similarly person identification and person re-identification. For example, the bib number in the marathon images can be used for person identification and re-identification. For sports videos, the text can be used for retrieving the exciting events and person tracing using a Multiview of the person. The work can also be used for surveillance and security applications such as watching exhibitions and processions and analyzing crowd behavior. There are other applications like image-to-text transformation, Visual Question Answering, and text image generation. In this review, we focus on STDR, a distinct subfield of Optical Character Recognition (OCR) that deals specifically with detecting and recognizing text in natural scenes such as images and videos. While STDR shares some techniques with traditional document analysis, it operates in more complex and dynamic environments. We clearly define the scope of this review by concentrating on STDR methods and outscoping other areas of OCR, such as printed document recognition, handwriting recognition, and CAPTCHA solving. This demarcation provides clarity on the focus of our review, helping readers understand where STDR fits within the broader OCR research landscape.

5.3. Directions

To address the new challenges of new applications, one should think of exploring robust, unified models and end-to-end models and, at the same time, exploring transformer-based models and language-based models. These models are capable of tackling challenges caused by distortion, different domains, quality, shapes, orientation, etc. For example, language models do not require recognition to predict the text. Furthermore, the language models can be used to restore the missing text and text with loss of characters.

6. Summary

This review presents a discussion on different text detection and recognition models on different datasets. When there are many methods available with the same objective, it is not so easy to draw inferences to understand the state-of-the-art. This makes it difficult to choose between an open challenge and a suitable approach for resolving difficulties in the future. The review assists readers in narrowing down the conclusion. This article offers a thorough examination of text recognition and detection in films and images of natural scenes. In conclusion, it is observed from the examination and analysis of several approaches that none of them produce consistent outcomes for various datasets and applications. Moreover, techniques designed for scene photos do not work well for videos. No techniques exist that are effective for both scene and video pictures. Furthermore, the approaches are not very effective when the picture domain changes. Therefore, there is room for future development of an approach that is independent of domain. This article provides not only a review of existing text recognition and detection methods but also offers a critical evaluation of these methods based on key criteria such as robustness to dynamic and shaky environments, computational efficiency, accuracy in complex scenes, and their applicability to real-world scenarios. Rather than merely summarizing the literature, we synthesize insights from various approaches and provide a framework that helps users make informed decisions when choosing methods for their specific applications. By focusing on these criteria, the review adds value beyond a simple literature survey, guiding researchers and practitioners in selecting techniques that align with their objectives and constraints.

Ethical Statement

This study does not contain any studies with human or animal subjects performed by any of the authors.

Conflicts of Interest

Palaiahnakote Shivakumara is the Editor-in-Chief and Umapada Pal is an Advisory Board Member for *Artificial Intelligence and Applications*, and were not involved in the editorial review or the decision to publish this article. The authors declare that they have no conflicts of interest to this work.

Data Availability Statement

Data sharing is not applicable to this article as no new data were created or analyzed in this study.

Author Contribution Statement

Arnab Halder: Conceptualization, Methodology, Software, Formal analysis, Investigation, Writing – original draft. **Palaiahnakote Shivakumara:** Methodology, Validation, Investigation, Writing – original draft, Supervision. **Umapada Pal:** Writing – review & editing, Supervision. **Michael Blumenstein:** Visualization.

References

[1] Greenhalgh, J., & Mirmehdi, M. (2015). Recognizing text-based traffic signs. *IEEE Transactions on Intelligent Transportation Systems*, *16*(3), 1360–1369. https://doi.org/10.1109/TITS.2014.2363167

[2] Yin, X. C., Zuo, Z. Y., Tian, S., & Liu, C. L. (2016). Text detection, tracking and recognition in video: A comprehensive survey. *IEEE Transactions on Image Processing*, *25*(6), 2752–2773. https://doi.org/10.1109/TIP.2016.2554321

[3] Ham, Y. K., Kang, M. S., Chung, H. K., Park, R. H., & Park, G. T. (1995). Recognition of raised characters for automatic classification of rubber tires. *Optical Engineering*, *34*(1), 102–109. https://doi.org/10.1117/12.184094

[4] Shilkrot, R., Huber, J., Liu, C., Maes, P., & Nanayakkara, S. C. (2014). FingerReader: A wearable device to support

text reading on the go. In *CHI'14 Extended Abstracts on Human Factors in Computing Systems*, 2359–2364. https://doi.org/10.1145/2559206.2581220

[5] Lyu, P., Yao, C., Wu, W., Yan, S., & Bai, X. (2018). Multi-oriented scene text detection via corner localization and region segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* 7553–7563.

[6] Liu, F., Chen, C., Gu, D., & Zheng, J. (2019). FTPN: Scene text detection with feature pyramid based text proposal network. *IEEE Access*, *7*, 44219–44228. https://doi.org/10.1109/ACCESS.2019.2908933

[7] Wang, Q., Huang, Y., Jia, W., He, X., Blumenstein, M., Lyu, S., & Lu, Y. (2020). FACLSTM: ConvLSTM with focused attention for scene text recognition. *Science China Information Sciences*, *63*, 120103. https://doi.org/10.1007/s11432-019-2713-1

[8] Bai, F., Cheng, Z., Niu, Y., Pu, S., & Zhou, S. (2018). Edit probability for scene text recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* 1508–1516.

[9] Bagi, R., Dutta, T., & Gupta, H. P. (2020). Cluttered TextSpotter: An end-to-end trainable light-weight scene text spotter for cluttered environment. *IEEE Access*, *8*, 111433–111447. https://doi.org/10.1109/ACCESS.2020.3002808

[10] Zhu, J., & Wang, G. (2022). TransText: Improving scene text detection via transformer. *Digital Signal Processing*, *130,* 103698. https://doi.org/10.1016/j.dsp.2022.103698

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

[12] Tian, Z., Huang, W., He, T., He, P., & Qiao, Y. (2016). Detecting text in natural image with connectionist text proposal network. In *Computer Vision – ECCV 2016: 14th European Conference*, 56–72. https://doi.org/10.1007/978-3-319-46484-8_4

[13] Wang, W., Xie, E., Song, X., Zang, Y., Wang, W., Lu, T., . . . , & Shen, C. (2019). Efficient and accurate arbitrary-shaped text detection with pixel aggregation network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 8440–8449.

[14] Li, X., Wang, W., Hou, W., Liu, R. Z., Lu, T., & Yang, J. (2018). Shape robust text detection with progressive scale expansion network. *arXiv Preprint:1806.02559*.

[15] Zhu, Y., & Du, J. (2021). TextMountain: Accurate scene text detection via instance segmentation. *Pattern Recognition*, *110*, 107336. https://doi.org/10.1016/j.patcog.2020.107336

[16] Kheng Chng, C., & Chan, C. S. (2017). Total-Text: A comprehensive dataset for scene text detection and recognition. *arXiv Preprint:1710.10400*. https://arxiv.org/abs/1710.10400v1

[17] Xu, Y., Li, M., Cui, L., Huang, S., Wei, F., & Zhou, M. (2019). LayoutLM: Pre-training of text and layout for document image understanding. *arXiv Preprint:1912.13318*. https://doi.org/10.48550/arXiv.1912.13318

[18] Liao, M., Zou, Z., Wan, Z., Yao, C., & Bai, X. (2022). Real-time scene text detection with differentiable binarization and adaptive scale fusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *45*(1), 919–931. https://doi.org/10.1109/TPAMI.2022.3155612

[19] Raja, E., Soni, B., & Borgohain, S. K. (2024). Fake news detection in Dravidian languages using multiscale residual CNN_BiLSTM hybrid model. *Expert Systems with Applications*, *250*, 123967. https://doi.org/10.1016/j.eswa.2024.123967

[20] Lucas, S. M., Panaretos, A., Sosa, L., Tang, A., Wong, S., Young, R., . . . , & Lin, X. (2005). ICDAR 2003 robust reading competitions: Entries, results, and future directions. *International Journal of Document Analysis and Recognition*, *7*, 105–122. https://doi.org/10.1007/s10032-004-0134-3

[21] Shahab, A., Shafait, F., & Dengel, A. (2011). ICDAR 2011 robust reading competition challenge 2: Reading text in scene images. In *2011 International Conference on Document Analysis and Recognition*, 1491–1496. https://doi.org/10.1109/ICDAR.2011.296

[22] Karatzas, D., Shafait, F., Uchida, S., Iwamura, M., i Bigorda, L. G., Mestre, S. R., . . . , & de las Heras, L. P. (2013). ICDAR 2013 robust reading competition. In *12th International Conference on Document Analysis and Recognition,* 1484–1493. https://doi.org/10.1109/ICDAR.2013.221

[23] Karatzas, D., Gomez-Bigorda, L., Nicolaou, A., Ghosh, S., Bagdanov, A., Iwamura, M., . . . , & Valveny, E. (2015). ICDAR 2015 competition on robust reading. In *13th International Conference on Document Analysis and Recognition*, 1156–1160. https://doi.org/10.1109/ICDAR.2015.7333942

[24] Nayef, N., Yin, F., Bizid, I., Choi, H., Feng, Y., Karatzas, D., . . . , & Ogier, J. M. (2017). ICDAR2017 robust reading challenge on multi-lingual scene text detection and script identification-RRC-MLT. In *14th IAPR International Conference on Document Analysis and Recognition, 1*, 1454–1459. https://doi.org/10.1109/ICDAR.2017.237

[25] Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., & Ng, A. Y. (2011). Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning, 2011*(2), 4.

[26] Jung, J., Lee, S., Cho, M. S., & Kim, J. H. (2011). Touch TT: Scene text extractor using touchscreen interface. *ETRI Journal*, *33*(1), 78–88. https://doi.org/10.4218/etrij.11.1510.0029

[27] Nagy, R., Dicker, A., & Meyer-Wegener, K. (2012). NEOCR: A configurable dataset for natural image text recognition. In *Camera-Based Document Analysis and Recognition: 4th International Workshop*, 150–163. https://doi.org/10.1007/978-3-642-29364-1_12

[28] de Campos, T. E., Babu, B. R., & Varma, M. (2009). Character recognition in natural images. In *Proceedings of the Fourth International Conference on Computer Vision Theory and Applications, 2*, 273–280.

[29] Gupta, A., Vedaldi, A., & Zisserman, A. (2016). Synthetic data for text localisation in natural images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* 2315–2324.

[30] Mishra, A., Alahari, K., & Jawahar, C. V. (2012). Top-down and bottom-up cues for scene text recognition. In *IEEE Conference on Computer Vision and Pattern Recognition,* 2687–2694. https://doi.org/10.1109/CVPR.2012.6247990

[31] Wang, K., Babenko, B., & Belongie, S. (2011). End-to-end scene text recognition. In *2011 International Conference on*

Computer Vision, 1457–1464. https://doi.org/10.1109/ICCV.2011.6126402

[32] Veit, A., Matera, T., Neumann, L., Matas, J., & Belongie, S. (2016). COCO-Text: Dataset and benchmark for text detection and recognition in natural images. *arXiv Preprint:1601.07140*. https://arxiv.org/abs/1601.07140v2

[33] Islam, M. R., Mondal, C., Azam, M. K., & Islam, A. S. M. J. (2016). Text detection and recognition using enhanced MSER detection and a novel OCR technique. In *5th International Conference on Informatics, Electronics and Vision*, 15–20. https://doi.org/10.1109/ICIEV.2016.7760054

[34] Tian, Z., Shu, M., Lyu, P., Li, R., Zhou, C., Shen, X., & Jia, J. (2019). Learning shape-aware embedding for scene text detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4234–4243.

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

[36] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780.

[37] Ma, J., Shao, W., Ye, H., Wang, L., Wang, H., Zheng, Y., & Xue, X. (2018). Arbitrary-oriented scene text detection via rotation proposals. *IEEE Transactions on Multimedia*, *20*(11), 3111–3122. https://doi.org/10.1109/TMM.2018.2818020

[38] Halder, A., Shivakumara, P., Pal, U., Lu, T., & Blumenstein, M. (2023). A new transformer-based approach for text detection in shaky and non-shaky day-night video. In *Asian Conference on Pattern Recognition*, 30–44. https://doi.org/10.1007/978-3-031-47637-2_3

[39] Deng, D., Liu, H., Li, X., & Cai, D. (2018). PixelLink: Detecting scene text via instance segmentation. *Proceedings of the AAAI Conference on Artificial Intelligence*, *32*(1), 6773–6780. https://doi.org/10.1609/aaai.v32i1.12269

[40] Yang, Q., Cheng, M., Zhou, W., Chen, Y., Qiu, M., & Lin, W. (2018). IncepText: A new inception-text module with deformable PSROI pooling for multi-oriented scene text detection. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence,* 1071–1077. https://dl.acm.org/doi/abs/10.5555/3304415.3304567

[41] Naiemi, F., Ghods, V., & Khalesi, H. (2021). A novel pipeline framework for multi oriented scene text image detection and recognition. *Expert Systems with Applications*, *170*, 114549. https://doi.org/10.1016/j.eswa.2020.114549

[42] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). SSD: Single shot multibox detector. In *Computer Vision – ECCV 2016: 14th European Conference*, 21–37. https://doi.org/10.1007/978-3-319-46448-0_2

[43] Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., & Wei, Y. (2017). Deformable convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision,* 764–773.

[44] Lee, C. Y., & Osindero, S. (2016). Recursive recurrent nets with attention modeling for OCR in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2231–2239.

[45] Liao, M., Zhang, J., Wan, Z., Xie, F., Liang, J., Lyu, P., . . . , & Bai, X. (2019). Scene text recognition from two-dimensional perspective. *Proceedings of the AAAI Conference on Artificial Intelligence, 33*(1), 8714–8721. https://doi.org/10.1609/aaai.v33i01.33018714

[46] Shikha, N., Pranav, R., Singh, N. R., Umadevi, V., & Hussain, M. (2023). Kannada word detection in heterogeneous scene images. In *10th International Conference on Signal Processing and Integrated Networks*, 379–383. https://doi.org/10.1109/SPIN57001.2023.10117096

[47] Zhong, Z., Jin, L., Zhang, S., & Feng, Z. (2016). DeepText: A unified framework for text proposal generation and text detection in natural images. *arXiv Preprint:1605.07314.* https://doi.org/10.48550/arXiv.1605.07314

[48] Long, S., He, X., & Yao, C. (2021). Scene text detection and recognition: The deep learning era. *International Journal of Computer Vision*, *129*(1), 161–184. https://doi.org/10.1007/s11263-020-01369-0

[49] Huang, W., Qiao, Y., & Tang, X. (2014). Robust scene text detection with convolution neural network induced MSER trees. In *Computer Vision – ECCV 2014: 13th European Conference*, 497–511. https://doi.org/10.1007/978-3-319-10593-2_33

[50] Wang, R., Sang, N., & Gao, C. (2015). Text detection approach based on confidence map and context information. *Neurocomputing*, *157*, 153–165. https://doi.org/10.1016/j.neucom.2015.01.023

[51] Naiemi, F., Ghods, V., & Khalesi, H. (2021). MOSTL: An accurate multi-oriented scene text localization. *Circuits, Systems, and Signal Processing*, *40*(9), 4452–4473. https://doi.org/10.1007/s00034-021-01674-0

[52] Liu, Z., Lin, G., Yang, S., Liu, F., Lin, W., & Goh, W. L. (2019). Towards robust curve text detection with conditional spatial expansion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,* 7269–7278.

[53] Liu, Y., Jin, L., Zhang, S., Luo, C., & Zhang, S. (2019). Curved scene text detection via transverse and longitudinal sequence connection. *Pattern Recognition*, *90,* 337–345. https://doi.org/10.1016/j.patcog.2019.02.002

[54] Zhang, C., Liang, B., Huang, Z., En, M., Han, J., Ding, E., & Ding, X. (2019). Look more than once: An accurate detector for text of arbitrary shapes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10552–10561.

[55] Lee, S., Cho, M. S., Jung, K., & Kim, J. H. (2010). Scene text extraction with edge constraint and text collinearity. In *20th International Conference on Pattern Recognition*, 3983–3986. https://doi.org/10.1109/ICPR.2010.969

[56] Fu, Y., Brown, N. M., Saeed, S. U., Casamitjana, A., Baum, Z., Delaunay, R., . . . , & Hu, Y. (2020). DeepReg: A deep learning toolkit for medical image registration. *arXiv Preprint:2011.02580.* https://doi.org/10.48550/arXiv.2011.02580

[57] Wang, P., Da, C., & Yao, C. (2022). Multi-granularity prediction for scene text recognition. In *European Conference on Computer Vision,* 339–355. https://doi.org/10.1007/978-3-031-19815-1_20

[58] Hu, H., Zhang, C., Luo, Y., Wang, Y., Han, J., & Ding, E. (2017). WordSup: Exploiting word annotations for character based text detection. In *Proceedings of the IEEE International Conference on Computer Vision*, 4940–4949.

[59] Xue, C., Lu, S., & Zhan, F. (2018). Accurate scene text detection through border semantics awareness and bootstrapping. In *Proceedings of the European Conference on Computer Vision,* 355–372.

[60] Sheng, F., Chen, Z., & Xu, B. (2018). NRTR: A no-recurrence sequence-to-sequence model for scene text recognition. *arXiv Preprint:1806.00926*. https://doi.org/10.48550/arXiv.1806.00926

[61] Lu, N., Yu, W., Qi, X., Chen, Y., Gong, P., Xiao, R., & Bai, X. (2021). MASTER: Multi-aspect non-local network for scene

text recognition. *Pattern Recognition*, *117*, 107980. https://doi.org/10.1016/j.patcog.2021.107980

[62] Yu, D., Li, X., Zhang, C., Liu, T., Han, J., Liu, J., & Ding, E. (2020). Towards accurate scene text recognition with semantic reasoning networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,* 12113–12122.

[63] Bartz, C., Yang, H., & Meinel, C. (2017). STN-OCR: A single neural network for text detection and text recognition. *arXiv Preprint:1707.08831*. https://doi.org/10.48550/arXiv.1707.08831

[64] Wang, T., Zhu, Y., Jin, L., Luo, C., Chen, X., Wu, Y., . . . , & Cai, M. (2020). Decoupled attention network for text recognition. *Proceedings of the AAAI Conference on Artificial Intelligence, 34*(7), 12216–12224. https://doi.org/10.1609/aaai.v34i07.6903

[65] Albawi, S., Mohammed T. A., & Al-Zawi, S. (2017). Understanding of a convolutional neural network. In *2017 International Conference on Engineering and Technology*, 1–6. https://doi.org/10.1109/ICEngTechnol.2017.8308186

[66] Medsker, L. R., & Jain, L. C. (1999). *Recurrent neural networks: Design and applications*. USA: CRC Press.

[67] Jiang, Y., Zhu, X., Wang, X., Yang, S., Li, W., Wang, H., . . . , & Luo, Z. (2017). R2CNN: Rotational region CNN for orientation robust scene text detection. *arXiv Preprint:1706.09579*. https://doi.org/10.48550/arXiv.1706.09579

[68] Cheng, Z., Bai, F., Xu, Y., Zheng, G., Pu, S., & Zhou, S. (2017). Focusing attention: Towards accurate text recognition in natural images. *arXiv Preprint:1709.02054*. https://doi.org/10.48550/arXiv.1709.02054

[69] Cheng, Z., Xu, Y., Bai, F., Niu, Y., Pu, S., & Zhou, S. (2017). AON: Towards abitrarily-oriented text recognition. *arXiv Preprint:1711.04226.* https://ui.adsabs.harvard.edu/link_gateway/2017arXiv171104226C/doi:10.48550/arXiv.1711.04226

[70] Bai, X., Shi, B., Zhang, C., Cai, X., & Qi, L. (2017). Text/non-text image classification in the wild with convolutional neural networks. *Pattern Recognition*, *66*, 437–446. https://doi.org/10.1016/j.patcog.2016.12.005

[71] Bagi, R., & Dutta, T. (2021). Cost-effective and smart text sensing and spotting in blurry scene images using deep networks. *IEEE Sensors Journal*, *21*(22), 25307–25314. https://doi.org/10.1109/JSEN.2020.3024257

[72] Li, H., Wang, P., & Shen, C. (2017). Towards end-to-end text spotting with convolutional recurrent neural networks. In *Proceedings of the IEEE International Conference on Computer Vision,* 5238–5246.

[73] Song, Q., Zhang, R., Zhou, Y., Jiang, Q., Liu, X., Wang, H., & Wang, D. (2019). Reading Chinese scene text with arbitrary arrangement based on character spotting. In *2019 International Conference on Document Analysis and Recognition Workshops, 5*, 91–96. https://doi.org/10.1109/ICDARW.2019.40087

[74] Zhong, Z., Sun, L., & Huo, Q. (2018). An anchor-free region proposal network for faster R-CNN based text detection approaches. *arXiv Preprint:1804.09003*. https://doi.org/10.48550/arXiv.1804.09003

[75] Cheng, G., Zhou, P., & Han, J. (2016). Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, *54*(12), 7405–7415. https://doi.org/10.1109/TGRS.2016.2601622

[76] He, T., Huang, W., Qiao, Y., & Yao, J. (2016). Text-attentional convolutional neural network for scene text detection. *IEEE Transactions on Image Processing*, *25*(6), 2529–2541. https://doi.org/10.1109/TIP.2016.2547588

[77] He, W., Zhang, X. Y., Yin, F., Luo, Z., Ogier, J. M., & Liu, C. L. (2020). Realtime multi-scale scene text detection with scale-based region proposal network. *Pattern Recognition*, *98,* 107026. https://doi.org/10.1016/j.patcog.2019.107026

[78] Chaitra, Y. L., Dinesh, R., Jeevan, M., Arpitha, M., Aishwarya, V., & Akshitha, K. (2022). An impact of YOLOv5 on text detection and recognition system using TesseractOCR in images/video frames. In *2022 IEEE International Conference on Data Science and Information System,* 1–6. https://doi.org/10.1109/ICDSIS55133.2022.9915927

[79] Chen, L., Shi, J., & Su, F. (2021). Robust video text detection through parametric shape regression, propagation and fusion. In *2021 IEEE International Conference on Multimedia and Expo*, 1–6. https://doi.org/10.1109/ICME51207.2021.9428195

[80] Liu, Z., Lin, G., Yang, S., Feng, J., Lin, W., & Goh, W. L. (2018). Learning Markov clustering networks for scene text detection. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition,* 6936–6944.

[81] Chhabra, P., Shrivastava, A., & Gupta, Z. (2023). Comparative analysis on text detection for scenic images using EAST and CTPN. In *7th International Conference on Trends in Electronics and Informatics*, 1303–1308. https://doi.org/10.1109/ICOEI56765.2023.10125894

[82] Xue, C., Lu, S., & Zhang, W. (2019). MSR: Multi-scale shape regression for scene text detection. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence,* 989–995.

[83] Nandanwar, L., Shivakumara, P., Ramachandra, R., Lu, T., Pal, U., Antonacopoulos, A., & Lu, Y. (2022). A new deep wavefront based model for text localization in 3D video. *IEEE Transactions on Circuits and Systems for Video Technology*, *32*(6), 3375–3389. https://doi.org/10.1109/TCSVT.2021.3110990

[84] Zhang, Y., Nie, S., Liu, W., Xu, X., Zhang, D., & Shen, H. T. (2019). Sequence-to-sequence domain adaptation network for robust text image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2740–2749.

[85] Banerjee, A., Shivakumara, P., Acharya, P., Pal, U., & Canet, J. L. (2022). TWD: A new deep E2E model for text watermark/caption and scene text detection in video. In *26th International Conference on Pattern Recognition*, 1492–1498. https://doi.org/10.1109/ICPR56361.2022.9956279

[86] Anto Bennet, M., Srividhya, R., Jayachandran, T., & Rajmohan, V. (2022). Deep learning based Telugu video text detection using video coding over digital transmission. In *2022 6th International Conference on Trends in Electronics and Informatics,* 1479–1483. https://doi.org/10.1109/ICOEI53556.2022.9776922

[87] Liu, Y., He, T., Chen, H., Wang, X., Luo, C., Zhang, S., . . . , & Jin, L. (2019). Exploring the capacity of sequential-free box discretization network for omnidirectional scene text detection. *arXiv Preprint:1912.09629.* https://doi.org/10.48550/arXiv.1912.09629

[88] Wu, W., Cai, Y., Shen, C., Zhang, D., Fu, Y., Zhou, H., & Luo, P. (2022). End-to-end video text spotting with transformer. *arXiv Preprint:2203.10539*. https://ui.adsabs.harvard.edu/link_gateway/2022arXiv220310539W/doi:10.48550/arXiv.2203.10539

[89] Song, Z., Zhang, H., & Cui, P. (2019). Towards end-to-end scene text spotting by sharing convolutional feature map. In *IEEE 5th International Conference on Computer and Communications,* 1814–1820. https://doi.org/10.1109/ICCC47050.2019.9064226

[90] Feng, C., Zhang, H., Li, X., Liao, K., & Lu, G. (2022). Character identifier spotting based on deep learning in video surveillance images. In *IEEE 10th International Conference on Information, Communication and Networks*, 530–536. https://doi.org/10.1109/ICICN56848.2022.10006626

[91] Placidi, J. C., Miao, Y., Wang, Z., & Specia, L. (2022) Scene text recognition with semantics. *arXiv Preprint:2210.10836.* https://doi.org/10.48550/arXiv.2210.10836

[92] Zhao, S., Quan, R., Zhu, L., & Yang, Y. (2023). CLIP4STR: A simple baseline for scene text recognition with pre-trained vision-language model. *arXiv Preprint:2305.14014*. https://doi.org/10.48550/arXiv.2305.14014

[93] Asadzadehkaljahi, M., Halder, A., Pal, U., & Palaiahnakote, S. (2024). Spatiotemporal edges for arbitrarily moving video classification in protected and sensitive scenes. *Artificial Intelligence and Applications, 2*(2), 92–99. https://doi.org/10.47852/bonviewAIA3202526

[94] Liu, Y., Zhang, J., Peng, D., Huang, M., Wang, X., Tang, J., . . . , & Jin, L. (2023). SPTS v2: Single-point scene text spotting. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 45*(12), 15665–15679. https://doi.org/10.1109/TPAMI.2023.3312285

[95] Li, M., Fu, B., Chen, H., He, J., & Qiao, Y. (2023). Dual relation network for scene text recognition. *IEEE Transactions on Multimedia*, *25*, 4094–4107. https://doi.org/10.1109/TMM.2022.3171108

[96] Shivakumara, P., Banerjee, A., Pal, U., Nandanwar, L., Lu, T., & Liu, C. L. (2023). A new language-independent deep CNN for scene text detection and style transfer in social media images. *IEEE Transactions on Image Processing*, *32*, 3552–3566. https://doi.org/10.1109/TIP.2023.3287038

[97] Raja, E., Soni, B., & Borgohain, S. K. (2023). Fake news detection in Dravidian languages using transfer learning with adaptive finetuning. *Engineering Applications of Artificial Intelligence*, *126,* 106877. https://doi.org/10.1016/j.engappai.2023.106877

[98] Raja, E., Soni, B., Lalrempuii, C., & Borgohain, S. K. (2024). An adaptive cyclical learning rate based hybrid model for Dravidian fake news detection. *Expert Systems with Applications*, *241,* 122768. https://doi.org/10.1016/j.eswa.2023.122768

How to Cite: Pal, U., Halder, A., Shivakumara, P., & Blumenstein, M. (2024). A Comprehensive Review on Text Detection and Recognition in Scene Images. *Artificial Intelligence and Applications*, *2*(4), 257–277. https://doi.org/10.47852/bonviewAIA42022755