International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
Article Received: *16 August 2023* ***Revised:*** *08 October 2023* ***Accepted:*** *24 October 2023*

Text Detection Using Transformation Scaling Extension Algorithm in Natural Scene Images

A.S.Venkata Praneel[1], Dr.T.Srinivasa Rao[2]
[1]Department of Computer Science and Engineering
GST, GITAM(Deemed - to -be University)
Visakhapatnam, Andhra Pradesh, India
praneelsri@gmail.com
[2]Department of Computer Science and Engineering
GST, GITAM(Deemed - to -be University)
Visakhapatnam, Andhra Pradesh, India
sthamada@gitam.edu

Abstract— In recent study efforts, the importance of text identification and recognition in images of natural scenes has been stressed more and more. Natural scene text contains an enormous amount of useful semantic data that can be applied in a variety of vision-related applications. The detection of shape-robust text confronts two major challenges: 1. A large number of traditional quadrangular bounding box-based detectors failed to identify text with irregular forms, making it difficult to include such text within perfect rectangles.2. Pixel-wise segmentation-based detectors sometimes struggle to identify closely positioned text examples from one another. Understanding the surroundings and extracting information from images of natural scenes depends heavily on the ability to detect and recognise text. Scene text can be aligned in a variety of ways, including vertical, curved, random, and horizontal alignments. This paper has created a novel method, the Transformation Scaling Extention Algorithm (TSEA), for text detection using a mask-scoring R-ConvNN (Region Convolutional Neural Network). This method works exceptionally well at accurately identifying text that is curved and text that has multiple orientations inside real-world input images. This study incorporates a mask-scoring R-ConvNN network framework to enhance the model's ability to score masks correctly for the observed occurrences. By providing more weight to accurate mask predictions, our scoring system eliminates inconsistencies between mask quality and score and enhances the effectiveness of instance segmentation. This paper also incorporates a Pyramid-based Text Proposal Network (PBTPN) and a Transformation Component Network (TCN) to enhance the feature extraction capabilities of the mask-scoring R-ConvNN for text identification and segmentation with the TSEA. Studies show that Pyramid Networks are especially effective in reducing false alarms caused by images with backgrounds that mimic text. On benchmark datasets ICDAR 2015, SCUT-CTW1500 containing multi-oriented and curved text, this method outperforms existing methods by conducting extensive testing across several scales and utilizing a single model. This study expands the field of vision-oriented applications by highlighting the growing significance of effectively locating and detecting text in natural situations.

Keywords- Text Detection, mask scoring R-ConvNN, Pyramid-based Text Detection Network, Instance Segmentation, Transformation Component Network.

I. INTRODUCTION

Text detection, the process of locating text in an image and enclosing it within rectangular boxes, employs two primary approaches: image-based and frequency-based methods. In image-based techniques, the image is divided into segments composed of pixels with similar features. Statistical properties of these components are utilized to organize and structure text. At the same time, machine learning methods like support vector machines (SVM) and convolutional neural networks (ConvNN) are employed to classify these components as either text or non-text.

While using methods like frequency-based methods, discrete wavelet transform (DWT) or discrete Fourier transform (DFT) concentrate on retrieving high-frequency components. It is thought that text has high-frequency components and separates text from non-text regions by isolating these high-frequency coefficients. With the help of several ConvNN-based object detection and segmentation frameworks, including Fully Convolutional Networks (FCNs) [1], Faster R-CNN (FRCNN) [2] and SSD [3], recent developments in deep learning have substantially improved text detection. These deep learning methodologies outperform time-honoured bottom-up text detection techniques like MSER [4] and SWT [5].

Some techniques [6,7] treat text detection as a semantic segmentation problem, employing FCNs to generate text/non-text predictions and construct a text saliency map. These methods can identify coarse text regions, but post-processing is often needed to obtain precise bounding boxes. Another category of methods treats text as a separate object and uses DenseBox [8], YOLO [9], SSD [3], and R-CNN [10] frameworks to automatically find words or text in images.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
Article Received: *16 August 2023* ***Revised:*** *08 October 2023* ***Accepted:*** *24 October 2023*

Although simpler in terms of processing, some algorithms might need assistance in detecting curved text.

Text detection is a challenge that more recent methods like PixelLink [11], IncepText [12], and FTSN [13] seek to solve as an instance segmentation issue. They have a single foundation for recognizing both straight and curved text. For instance, to successfully recognize text, PixelLink joins pixels within the same text instance. This study uses an R-ConvNN-based text detection system with mask scoring to accurately detect curved and multi-oriented text in real-scene images. In order to enhance the capabilities of the mask-scoring R-ConvNN [14], the PBTPN, which was inspired by the Attention Pyramid Network [15], is used in tandem with a transformation component network as a novel backbone network. On benchmark datasets, experiments show that this approach limits false alarms brought on by backdrops that seem like text while outperforming other approaches. The field of scene text detection is continually evolving and holds great promise for applications in data analysis, information retrieval, OCR translation, robot navigation, and augmented reality, driving the growing interest of the Computer Vision community.

II. RELATED WORK

This section examines recent developments in instance segmentation issues as well as recently proposed text identification methods employing CNN from authors who have previously published significant work in this area.

A. Text Detection

In prior years, ConvNN-based object detection has gained significant traction in addressing the challenge of text detection. Some algorithms, such as those proposed in references [6,7], draw inspiration from semantic segmentation techniques. To find out whether a pixel contains text or not, they use (FCNs), which produce a text saliency map for text detection. Still, these methods are limited because they primarily identify coarse text blocks, necessitating more advanced post-processing techniques to accurately extract bounding boxes around the text.

Alternatively, several researchers approach text as a distinct object and leverage state-of-the-art object detection techniques [16,17,18,19,20,21,22] to identify text lines or words in images directly. Jaderberg [20] developed an approach for text detection using RCNN, but the traditional region proposal generation techniques-imposed limitations on its performance. To address horizontal text at the word level, approaches like Faster R-CNN (FRCNN) and Single Shot MultiBox Detector (SSD) were used [17,21]. The YOLO framework was used by Gupta [22] to achieve Bounding Box Regression (BBR) and text identification at different scales and places within an image. Quadrilateral anchors were also studied in order to better suit multi-oriented text instances with inclined text recommendations, which improved the performance of the FRCNN [2] and SSD [3]. Other approaches, such as those in [3,18], used the DenseBox concept and a one-stage FCN to create pixel-wise text results and Quadrilateral bounding boxes at various scales and orientations, thus addressing the drawbacks of the anchor-based method. Despite these advancements, they still struggle to recognize curved text.

Furthermore, specific techniques, as described in references [23,24], utilize object detection methods to first identify text segments, which, instead of directly identifying whole words or text lines, are then sorted into words or lines utilizing fundamental text-line clustering techniques or connectivity-based approaches. While these methods can automatically detect curved text, they introduce increased complexity into the overall process.

To address text detection as an instance segmentation task, [23,24] applied object identification algorithms to identify text segments, which were subsequently categorized into words or lines. In this paper, a state-of-the-art instance segmentation method called Mask Scoring R-ConvNN was used to enhance text detection performance. To efficiently detect text, Deng [11] suggested linking pixels contained within identical text instances.

B. Instance Segmentation

The Detection and segmentation-based methods are the two important types of instance segmentation methods now in use. Detection-based approaches focus on using detectors to identify the regions of each instance and then predict masks for each of these regions. Notable examples of detection-based techniques include Faster R-CNN (FRCNN) [2], Region-based Fully Convolutional Networks (R-FCN) [25], and DeepMask [26]. Additionally, instance-sensitive Fully Convolutional Networks (FCNs) [27] have been proposed to create position-sensitive maps for merging into final masks. The instance segmentation results are achieved through methods like FCISF [28], Mask R-CNN (MRCNN) [29], and MaskLab [30], which build upon the foundation of detection-based frameworks by incorporating instance-level semantic segmentation branches. However, a drawback of these methods is that mask quality is exclusively based on categorization scores, which presents particular difficulties.

However, segmentation-based algorithms work by predicting the category labels of every pixel before combining them to produce instance segmentation results. These methods often employ techniques like spectral clustering [31], border detection data [32], watershed algorithms [33], and metric learning to cluster pixels into instances. While they can efficiently organize pixels into models, they frequently rely on average pixel-level classification scores to evaluate the quality of an instance mask.

Both categories of approaches tend to overlook the correlation between mask quality and score. This can lead to

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
Article Received: *16 August 2023* ***Revised:*** *08 October 2023* ***Accepted:*** *24 October 2023*

instances with high Intersection over Union (IoU) values with ground truth being incorrectly deprioritized if they have low mask scores. Consequently, this can degrade the final average precision of the instance segmentation.

Instance segmentation is a challenging problem as it involves incorporating accurate image features and precise segmentation for each instance. Some multi-stage processes have been proposed, such as segment proposals from bounding-box suggestions followed by classification. However, despite the speed of methods like FCISF [28], they may need to improve on overlapping instances and generate incorrect edges.

Mask R-CNN (MRCNN) [29], which expands FRCNN [2] by including a bounding box identification branch in addition to an object mask prediction branch, is one example of recent developments in this area. It also utilizes advanced network architectures like ResneXt and incorporates techniques like RoIAlign to address pixel-level mismatches. Additionally, methods like Pyramid-based Text Detection Networks are introduced to improve feature representation capabilities, effectively suppressing false alarms caused by text-like backgrounds.

Overall, the field of instance segmentation is evolving with a focus on addressing the limitations in existing techniques and enhancing the quality of instance masks and detections.

C. Mask - Scoring RCNN

Each instance's classification is normally considered as a score in the context of the majority of instance segmentation frameworks, reflecting the assurance in the caliber of the associated mask for that instance. However, a significant issue arises because there often needs to be a clear correlation between the quality of the mask and the classification score. Where the quality of the mask is measured by MaskIoU (Intersection over Union of the ground truth and the predicted mask).

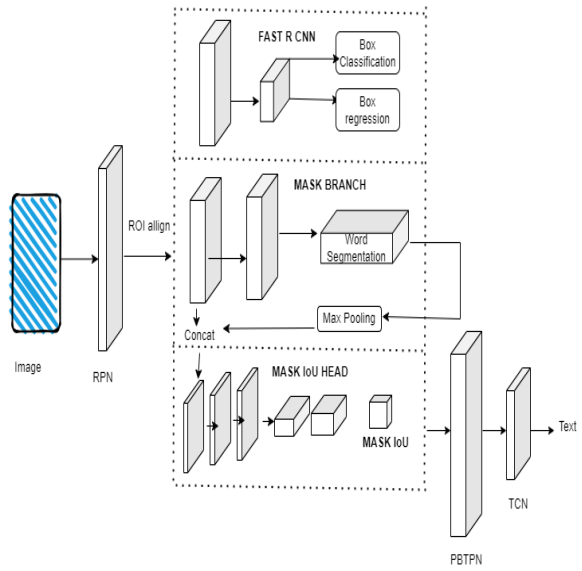

Figure 1: Structure of the Method

The mask scoring R-ConvNN presents a new direction to solve this problem by utilizing the instance's features as well as the projected mask to calculate the MaskIoU. With this method, the categorization score and the mask's real quality are reconciled. By doing so, it rectifies any discrepancies between these two measures. In essence, this Mask Scoring system ensures that more accurate mask predictions are given higher priority during the instance segmentation process.

This invention has improved instance segmentation's overall efficacy in the end. By awarding greater marks to cases with more accurate mask predictions it enables the algorithm to make more educated decisions. This correction aligns the model's confidence in instance masks with the actual quality of those masks, leading to improved performance in tasks involving data analysis, object detection, and segmentation.

III. PROPOSED WORK

A. Framework

The entire framework of the suggested strategy is shown in Figure 1. The main building blocks of this method are a PBTPN, a Region Proposal Network (RPN), a framework for Bounding Box Regression (BBR), a MaskIoU head branch, and a Mask branch in charge of text instance segmentation.

During the learning phase, the proposed method initiates by generating a multitude of text candidate boxes utilizing the RPN. These candidate boxes are then linked to Region of Interest (ROI) features, which serve as inputs to the MaskIoU head. The MaskIoU head's function is to forecast the MaskIoU, which expresses the caliber of the masks connected to these candidate boxes.

The ROI features from the candidate boxes are simultaneously sent to the Fast R-CNN branch for additional processing. The Mask branch also enters the picture, producing a precise Text candidate box and a Text Instance Segmentation map. The task of text detection and segmentation depends on these highly precise outputs.

This structure demonstrates the suggested method's systematic approach, combining several elements to produce effective text identification and segmentation, making it a useful tool in situations where precise localisation and recognition of text in images are essential. This method looks like an ensemble of multimodel representations [34] that produces a notable improvement in detection.

B. Backbone

The size of text in natural images varies widely, making text identification difficult. A ResNet50 backbone along with a PBTPN backbone are used to solve this problem. This mixture successfully creates a feature map with a rich of semantic data at different scales.

The PBTPN backbone is essential to this strategy. It uses top-down processing to make use of contextual information and

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
Article Received: *16 August 2023* ***Revised:*** *08 October 2023* ***Accepted:*** *24 October 2023*

combine data from several resolutions into a single input scale. When dealing with text of various sizes, the PBTPN makes sure that the feature map maintains a constant and unified scale.

This method works especially well for finding small targets, like text. The PBTPN operates on larger feature maps, improving the resolution of the feature map and supplying more pertinent data for precisely recognizing and localizing smaller text items in the natural image. Essentially, this method makes use of the PBTPN and the ResNet50 backbone to effectively solve the issues posed by text size variance in natural images, leading to more reliable and accurate text identification capabilities.

C. RPN

In this method, text proposals are generated for both the Fast R-CNN branch and the mask branch via the Region Proposal Network (RPN). The anchors employed by the RPN are carefully assigned to various levels or phases based on their sizes in order to handle the variety of text region sizes. These sizes, which correspond to five phases denoted P2, P3, P4, P5, and P6, are primarily described as 32 x 32, 64 x 64, 128 x 128, 256 x 256, and 512 x 512 pixels.

It's vital to remember that not all text regions have square corners; instead, they can have different aspect ratios. As a result, the aspect ratios are changed to 1:2, 1:1, and 2:1 at each step. As a result, 15 anchors are produced, enabling the RPN to effectively handle regions of aspect ratio or any size.

The strategy uses RoI Align rather than RoI Pooling to obtain regional proposals. RoI Align is chosen because it can maintain more precise positional information, which is essential for the segmentation operation of the mask branch. It makes sure that the fine spatial details of text sections are preserved, improving the segmentation process quality.

In traditional cases of object detection, feature extraction often takes place on a 3x3 Convolutional layer. Nonetheless, plenty of the text elements that are used to describe nature scenes are rectangular in design and have a sizable length and width. A typical 3x3 convolution layer cannot adequately capture these distinctive qualities. Convolution kernels made to extract features from anchors with different aspect ratios are incorporated into the RPN regression process as a novel technique to handle this. In addition to the usual 3x3 convolutions, this entails adding a 1x5 convolution layer, followed by a 5x1 convolution layer. This innovative design enhances the accuracy and robustness of text detection inside natural scene images by enabling the RPN to efficiently process and extract characteristics from text regions of various forms.

D. Fast R-CNN

The main objectives of the Fast R-CNN (FRCNN) branch are classification and regression. In order to build the most compact bounding rectangle boxes and increase the precision of bounding box detection, several activities are essential.

The RoI Align operation, which is derived from the region proposals produced by the RPN, is used to make these duties easier. A feature map with a 7x7 pixel resolution is provided by RoI Align. The Fast R-CNN branch then receives this feature map for additional processing and analysis.

It's important to note that a 7x7 feature map is thought to be adequate for efficient Text detection. The exact localisation and classification of items inside the image are made possible by this resolution. The Fast R-CNN branch is able to accomplish its primary objective of creating precise and compact bounding boxes while ensuring the overall accuracy of object detection by working at this resolution.

E. Mask Branch

The main responsibility of the Mask branch is to split the entire image's text per instance. This branch accepts a Region of Interest (RoI) with a size requirement of 16x64 pixels as input. This method's mask branch forecasts two distinct map types: a backdrop and a global text instance map.

The mask branch employs a number of layers, including one Deconvolutional (deConv) layer and four Convolutional (Conv) layers, to accomplish this. These layers play a key role in creating the necessary maps. Without explicitly recording the shape of the text instance, the Text instance map created by this branch is intended to pinpoint the exact placement of text areas.

It's significant to notice that the character background map does not include the character regions, which are essentially tiny text pieces or components. This distinction simplifies the instance segmentation procedure and increases the method's overall accuracy by enabling the successful segmentation of the main text sections without capturing the smaller components or characters.

F. MaskIoU Head

A significant part of the model, the MaskIoU head, is in charge of calculating the regression of the Intersection over Union (IoU) between the ground truth mask provided in the training data and the anticipated mask created by the model.

The RoI Align layer, which offers accurate location details about the region of interest (RoI) and the predicted mask produced by the model, is used as input by the MaskIoU head to complete this IoU regression. By quantifying the IoU, a statistic used to measure the overlap or intersection between two masks, the MaskIoU head's goal is to determine how well the predicted mask matches with the ground truth mask.

The MaskIoU head plays a crucial role in improving the precision of mask predictions and, as a result, the overall effectiveness of the instance segmentation task by performing this regression. This part makes sure that the ground truth masks

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
Article Received: *16 August 2023* ***Revised:*** *08 October 2023* ***Accepted:*** *24 October 2023*

and forecasted masks closely match, producing more accurate and dependable outcomes.

G. TCN

When used to refine and fix an input image using a predicted 2D transformation, the transformation component is critical. Such changes can be helpful when working with images, particularly when it comes to text identification and instance segmentation. since they can enhance the image's quality and make it more conducive to further analysis.

The transformation component receives as input a predicted 2D transformation, which frequently includes translation, rotation, scaling, and shearing parameters. These settings specify how the image should be corrected for distortions, alignment problems, or other flaws.

This component's main objective is to successfully apply the predicted transformation to the input image. It can then fix problems like misalignment, perspective distortions, or scale variances. After correction, the image is prepared for additional processing, such as word recognition, instance segmentation, or object detection tasks.

The significance of transformation component can be particularly significant when dealing with text in natural scene images in the context of text identification and instance segmentation. Text in these photos frequently has different orientations, viewpoints, and scales. The text is normalized and aligned as a result of the expected 2D transformation, making it easier to analyse it later and eventually improving the accuracy of the analysis and identification tasks.

In conclusion, the transformation component is an important part that corrects and enhances input images by applying expected 2D transformations, ensuring that they are ready for tasks like text identification and instance segmentation that come later in the analysis process.

IV. PBTPN

The PBTPN, which is made up of the Up-sample Unit (USU) and Pyramid Network (PN), is an essential part of the text detection paradigm. Together, these segments strengthen feature representation and increase the precision of text segmentation. In the PN segment, pooling and a spatial pyramid algorithm are combined to learn high-level image characteristics based on significant features in the input data. It creates a more thorough grasp of the image's content and successfully gathers context information. The PN module uses the Res-4-layer output characteristics from ResNet50 or ResNeXt50 as its input to do this. To obtain context information, it applies 3x3 Dilated Convolutions with various dilation rates (3, 6, and 12). The three feature maps were concatenated to reduce their size, and a 1x1 convolution layer was then applied. The input Res-4 features are further subjected to another 1x1 convolution using PN, and the pixel-wise result is then multiplied with the previously recorded context characteristics. As a final step, the output features from the global pooling branch are combined with the extracted features to create a feature representation that spans many scales.

On the other hand, the USU segment plays a complementary role by being placed on every decoder layer. It provides a broad context and aids in choosing basic attributes for category localization information. The USU module performs 3x3 Convolution on low-level features, reducing the number of channels in the CNN feature maps. Following a 1x1 Convolution operation with Instance Normalization and ReLU Nonlinearity, the low-level features are mixed with the context information produced from the high-level features. The high-level features are upsampled, weighted, and combined with the low-level features to produce the final USU features.

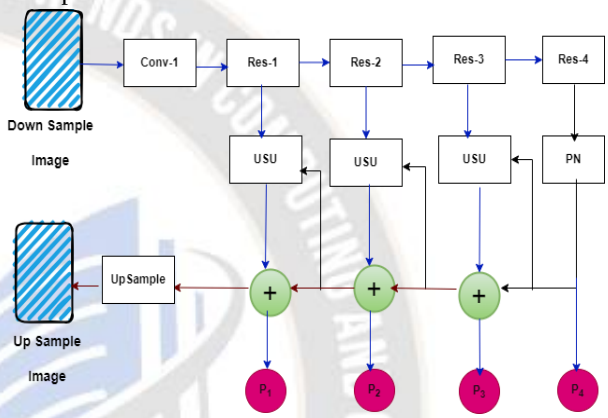

Figure 2: Structure of PBTPN

These two segments, PN and USU, work together to create a robust feature pyramid with three tiers: P1, P2, P3, and P4, each having different strides (2, 4, 8, and 16, respectively) as shown in Figure 2. This feature pyramid encompasses a range of scales and levels of detail, making it highly effective for the task of text detection and segmentation within natural scene images.

The innovations introduced by PBTPN, including the combination of PN and USU modules based on ResNet50 and ResNeXt50, contribute significantly to the model's success in achieving precise segmentation and feature representation learning. These enhancements result in improved accuracy in the detection and localization of text in complex image environments.

V. TRANSFORMATION COMPONENT NETWORK

By using a predicted 2D transformation, the transformation component is crucial in rectifying an input image. The Thin-Plate-Spline (TPS) [35] transformation, which is renowned for its adaptability in image transformation and matching, is the specific transformation technique used in this approach.

TPS has a lot of benefits over other 2D transformations, such as projective and affine transformations. Its capacity to alter images in a non-rigid way is one of its main advantages. Because

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
Article Received: *16 August 2023* ***Revised:*** *08 October 2023* ***Accepted:*** *24 October 2023*

of its adaptability, it can manage a variety of distortions, which makes it especially ideal for solving a range of issues in image repair and enhancement.

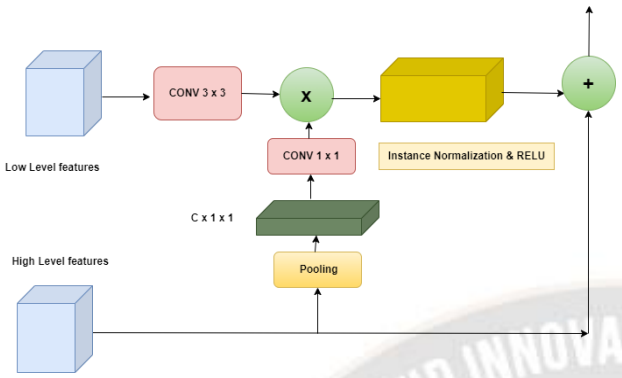

Figure 3: Structure of the USU

Some frequent rectifications made with TPS are shown in Figure 3. Perspective text and curved text are two frequent forms of irregular text that TPS is particularly good at fixing. TPS is well adapted to handle these challenging irregular text patterns that frequently arise in image processing and recognition.

Using a Spatial Transformer Network (STN) [36], the TPS transformation is implemented. The STN is a layer of a learnable network created to efficiently model spatial transformations. Figure 4 illustrates numerous crucial components that go into the design of the transformation component.

The localization network, which starts the process, initially anticipates a collection of control points. The TPS transformation is computed using these control points as a starting point. The corrected image, designated as "I_t," is then produced by passing the TPS transformation along to the sampler and grid component.

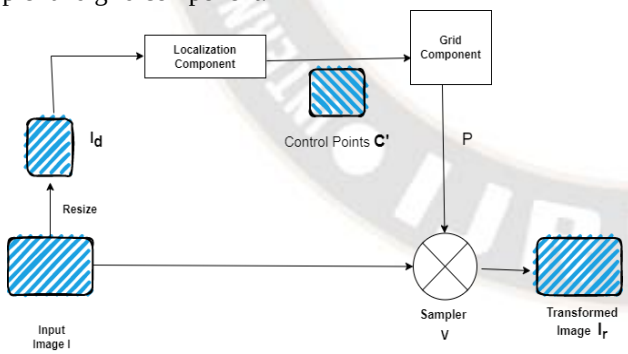

Figure 4: Structure of the TCN

It's important to note that the transformation component only needs the input image itself as an input. The reason for this is that the control points required for the transformation are immediately predicted from the input picture "I." As a result, this method effectively enhances and corrects input photos, making them better suited for the next processing steps, especially when there is erroneous or distorted text in the images.

A. Localization Component

a) Figure 4 demonstrates how text in a picture can be corrected using the Thin-Plate-Spline (TPS) transformation. Control points, designated as 'C,' are important in figuring out the TPS transformation. Two sets of the same size make up these control points.

b) The control points in this particular situation are uniformly spaced and fixed; predicting the control points along the top and bottom text borders edges of the input image is the objective. After precisely anticipating these control points on the input image, the TPS transformation may be used to correct the text, producing an image with both regular and corrected text.

c) The prediction of control point placements on the input image is a crucial step in the text rectification problem. Convolutional Neural Network (CNN) predictions are used to do this. The control points, denoted by the letters "C_p" for the input image "I" and "C_p" for the converted image "I_t," are intended to be predicted by the network. The letter "C" stands for the number of control points.

d) This methodology's key feature is the differentiability of each module in the transformation component. This characteristic is important because it allows the model to develop and change while being trained. Specifically, during training, back-propagated gradients are used to train the localization component. This streamlines training by doing away with the requirement for manual control point annotation and making it more efficient.

e) In conclusion, this method uses a CNN-based localization component and TPS transformation to forecast and use control points for correcting text in images. The model's differentiable components' adaptability and flexibility help it to be effective in text correction, and the removal of manual control point annotations makes training easier.

B. Grid component

On the modified image "I_t," the grid component generates a sample grid named "G" that is made up of specific grid points denoted as "G_i." To achieve this, a change is calculated and applied to each pixel point in "I_t."

The TPS solution and transformation method can be compared to a neural network module, as shown in Figure 4. This module receives as inputs 'C_p' (the anticipated control points for the transformed image) and produces 'g' for each pixel position 'p' on the corrected image.

It's significant to remember that a certain resolution's image has set pixel locations. As a result, the mapping between pixel coordinates "p" and "p" for images of the same resolution can be pre-calculated and stored. When working with photos of a similar resolution, the procedure is more effective and computationally cost-effective because this precomputed information can be used several times.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
Article Received: *16 August 2023* ***Revised:*** *08 October 2023* ***Accepted:*** *24 October 2023*

In actuality, the grid component and the calculations that go along with it are crucial in facilitating the transformation of the image, allowing precise and accurate alterations to be made to the positions of pixels in the corrected image 'I_t.' This procedure is essential for text rectification as well as for enhancing image quality for later processing and analysis.

C. Sampler

At the output of the transformation component, the sampler is a critical component that creates the corrected image. This is done by using the grid 'G' to apply a transformation on the input image 'I' in order to create the corrected image 'I_t.' This method can be pictured as:

$$S = It\ (G; I) \quad (1)$$

In this equation, "S" stands for the transformed image "I_t," while "I" is transformed using the grid "G."

The sampler employs interpolation based on the nearby pixels of the corresponding transformed location "p" in "I" to determine the value of each pixel "p" in the corrected image "I_t." It's vital to remember that 'p" could be situated outside of the image's bounds. Value clipping is used prior to sampling to remedy this and make sure that the sampling points are within the image area.

The sampler's differentiability is one outstanding quality. To backpropagate gradients from the corrected image (I_t) to the original pixel coordinates (p) in the input Image (I), the sampler has to be configured in a certain way. This approach to differentiable image sampling is essential for neural network training and transformation process optimization.

In conclusion, by applying the TPS transformation to the input image, the sampler is a component that is crucial to producing the corrected image. The final corrected image is generated precisely by using interpolation techniques to address situations where transformed locations are outside the image bounds. The differentiability of the sampler is a key feature for training and optimizing the transformation component within neural networks.

VI. TRANSFORMATION SCALING EXTENSION ALGORITHM [TSEA]

```
Function Transformation Scaling Extension Algorithm
(proposed regions)
detected_text_regions = []
for scale in [p1, p2, p3, p4]:
  expanded_text_regions=scale_expansion_algorithm(scale, S)
  transformed_text_regions=apply_TCN_transformation
  (expanded_text_regions, TCN_parameters)

  Final_text_regions=apply_spatial_transformation(transforme
  d_text_regions)
  detected_text_regions.extend(final_text_regions)
return detected_text_regions
```

```
def scale_expansion_algorithm(scale, S):
T = [] , P = [] ,Q = [] # Queue for BFS
expanded_text_regions = []
          for pixel in scale
           T.append((pixel, label))
           P.append(pixel)
           Q.append(pixel)
   while Q:
     p, label = Q.pop(0)
     neighbors = Neighbor(p)
     for q in neighbors:
       if q not in P and S[q] == True:
         T.append((q, label)
         P.append(q)
         Q.append(q)
   E = GroupByLabel(T)
expanded_text_regions.extend(E)
return expanded_text_regions
```

VII. DATASETS AND RESULTS

ICDAR2015 [37] and SCUT-CTW1500 [38], two well-known standard datasets, were used to conduct extensive testing on the method's performance and capabilities. With regard to text detection and, in the case of SCUT-CTW1500, curved text detection, these datasets serve as comparisons to evaluate the method's efficacy.

A. ICDAR 2015

The ICDAR2015 dataset for multi-oriented text detection is well known. ICDAR2015 emphasizes text detection in a more significant way than its predecessor, ICDAR2013, which puts more of a focus on text in particular settings.

1) There are 1,500 total images in the dataset. One thousand of them are set aside for training, giving the model access to a wide range of samples.

2) The remaining 500 images will be used for testing, allowing an impartial assessment of the method's effectiveness on unobserved data.

3) The dataset's ability to recognize text in a variety of angles and settings makes it a great candidate for testing the model's text detection capabilities.

B. SCUT-CTW1500

On the other hand, SCUT-CTW1500 is a specific dataset created for the recognition of curved text. Curved text's non-linear and frequently uneven shape makes it a special challenge for text detection.

1) The dataset consists of 1,000 photos in total, 500 of which are used for training and the remaining 500 for testing.

2) The dataset's emphasis on curved text makes it especially useful for assessing how well the algorithm can handle this difficult feature of text detection.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
Article Received: *16 August 2023* ***Revised:*** *08 October 2023* ***Accepted:*** *24 October 2023*

C. RESULTS and COMPARISONS

Precision, Recall, and F1 Score are fundamental performance metrics used in the field of machine learning and data science to assess the effectiveness of classification models, particularly in binary classification tasks.

1) Precision: also known as Positive Predictive Value, measures the proportion of true positive predictions (correctly predicted positive instances) out of all positive predictions made by a model. - It quantifies the model's accuracy in predicting the positive class and its ability to avoid making incorrect positive predictions. High precision is essential when minimizing false positives is crucial, such as in medical testing or fraud detection

$$\text{Precision (P)} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Positives (FP)}}$$

2) Recall: also known as Sensitivity or True Positive Rate, assesses the model's ability to correctly identify all relevant instances of the positive class. It measures the model's capability to capture and correctly classify all positive instances, thereby minimizing false negatives. High recall is crucial when missing positive instances has significant consequences, such as in medical diagnoses or security applications.

$$\text{Recall (R)} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Negatives (FN)}}$$

3) F1-Score: The F1 Score is a single metric that balances both precision and recall. It is the harmonic mean of precision and recall, providing a comprehensive assessment of a model's performance. It takes into account both false positives and false negatives and is particularly useful in situations where there is an imbalance between the classes or where you want to balance the trade-off between precision and recall. The F1 Score is valuable for evaluating classification models when there is a need to strike a balance between minimizing false positives and false negatives.

$$\text{F1 Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

4) Comparison with existing methods on ICDAR 2015:

On the ICDAR-2015 dataset, Table 1 compares the performance of a suggested technique with many current methods in terms of precision (P), recall (R), and F1-score (F). The following is an overview of the comparison:

Among all techniques, the "Proposed" method has the greatest F1-score of 0.867. It also has high accuracy (0.912) and recall (0.828), placing it first in this evaluation. With an F1-score of 0.859, "Mask R CNN + PAN" comes close to the suggested technique. However, the proposed method exceeds it in terms of precision and recall.

TABLE I. COMPARISON WITH EXISTING METHODS ON ICDAR-2015. P, R, AND F STAND FOR PRECISION, RECALL, AND F1-SCORE, RESPECTIVELY

Method for STD	**Comparison with existing Methods on ICDAR-2015.**		
	P	**R**	**F**
Proposed	**0.912**	**0.828**	**0.867**
Mask R CNN + PAN[39]	0.908	0.815	0.859
InceptText[12]	0.905	0.806	0.853
FTSN [13]	0.886	0.800	0.841
R2CNN[40]	0.856	0.749	0.825
DDR[16]	0.820	0.800	0.810
EAST [18]	0.832	0.783	0.774
RRPN[41]	0.822	0.732	0.774
Seglink[23]	0.786	0.731	0.749

The techniques "InceptText," "FTSN," "R2CNN," "DDR," "EAST," "RRPN," and "Seglink" had decreasing F1-scores, with "Seglink" having the lowest F1-score of 0.749. Among the known approaches, "DDR" has the best accuracy (0.820), but its recall is rather poor, resulting in an F1-score of 0.810"EAST" has an F1-score of 0.774 based on an accuracy of 0.832 and recall of 0.783.Among the known approaches, "RRPN" has the lowest accuracy (0.822), but a slightly greater recall (0.732), resulting in an F1-score of 0.774. The precision (0.786) of "Seglink" is the lowest of all techniques, with a recall of 0.731 and an F1-score of 0.749.

In the evaluation of text detection methods based on F1 scores, the "Proposed" method emerges as a strong performer, consistently leading the pack in percentage terms. It outpaces "Mask R CNN + PAN" by approximately 1.87%, showcasing its superior performance. "IncepText" closely follows, but the "Proposed" method maintains an approximately 1.17% advantage. "FTSN" lags behind by about 3.30%, and "R2CNN" is surpassed by a margin of approximately 4.48%. "DDR" exhibits a difference of around 6.30%, highlighting the "Proposed" method's substantial lead. Furthermore, the "Proposed" method showcases an impressive 11.40% edge over "EAST" and an approximately 11.20% advantage over "RRPN." Finally, it significantly outperforms "Seglink" by about 15.49%, affirming its status as the top-performing method across these comparisons. These results underline the strength and effectiveness of the "Proposed" method in the realm of text detection, where precision and recall are of utmost importance. In this comparison, the "Proposed" technique performs best, with the greatest F1-score, precision, and recall among the methods tested on the ICDAR-2015 dataset. This implies that the suggested technique performs the best balanced and accurate text detection in the current context.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
Article Received: *16 August 2023* ***Revised:*** *08 October 2023* ***Accepted:*** *24 October 2023*

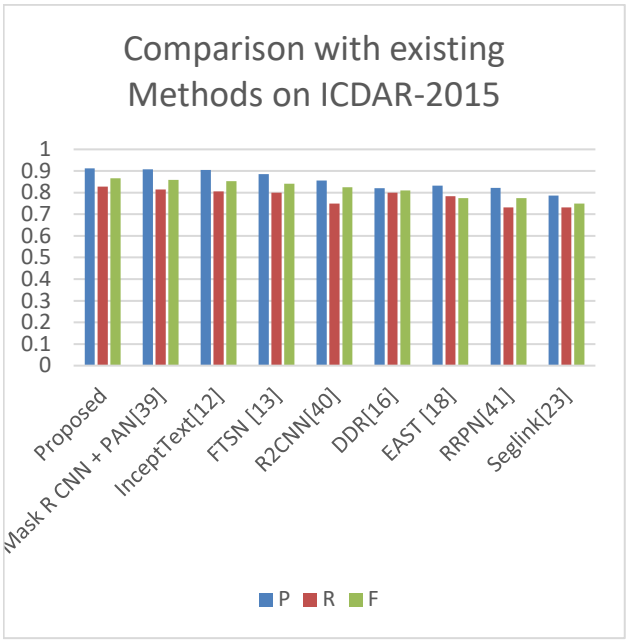

Figure 5: Comparison of Existing Methods on ICDAR 2015 Dataset

Figure 6: Examples of ICDAR 2015 produced by our Transformation Scaling Extension Algorithm

5) Comparision with existing methods on SCUT - CTW 1500:

The "Proposed" technique performs admirably in the evaluation of text detection algorithms on the SCUT - CTW 1500 dataset, with a precision (P) of 0.876, recall (R) of 0.846, and F1-Score (F) of 0.860. It excels at accuracy and recall, achieving a balance that yields a strong F1 score. It is still the best option when compared to other available approaches. With an accuracy of 0.868, recall of 0.832, and an F1-score of 0.850, "Mask R CNN + PAN" is behind the "Proposed" approach by a little margin. "CDT+TLOC" and "DMPNet" had lower F1 scores of 0.734 and 0.622, respectively. The "EAST" approach has an F1-score of 0.604, whereas the "CTPN" method has an F1-score of 0.569.

TABLE II. COMPARISON WITH EXISTING METHODS ON SCUT - CTW 1500. P, R, AND F STAND FOR PRECISION, RECALL, AND F1-SCORE, RESPECTIVELY

Method for Text Detection	**Comparison with existing Methods on SCUT - CTW 1500.**		
	P	**R**	**F**
Proposed	**0.876**	**0.846**	**0.860**
Mask R CNN + PAN[39]	0.868	0.832	0.850
CDT+TLOC[38]	0.774	0.698	0.734
DMPNet[19]	0.699	0.560	0.622
EAST [18]	0.787	0.491	0.604
CTPN[24]	0.604	0.538	0.569

When comparing percentage differences in performance, the "Proposed" approach outperforms "Mask R CNN + PAN" by around 1.17%, confirming its outstanding performance. It outperforms "CDT+TLOC" by roughly 17.81% and "DMPNet" by approximately 23.86%. When compared to "EAST," it has a significant 42.05% advantage, and when compared to "CTPN," the "Proposed" technique has a significant 51.75% advantage. These findings highlight the "Proposed" method's efficacy and resilience in text detection tasks, notably on the SCUT - CTW 1500 dataset.

In this comparison, the "Proposed" technique performs best, with the greatest F1-score, precision, and recall among the methods tested on the SCUT - CTW 1500 dataset. This implies that the suggested technique performs the best balanced and accurate text detection *in the current context.*

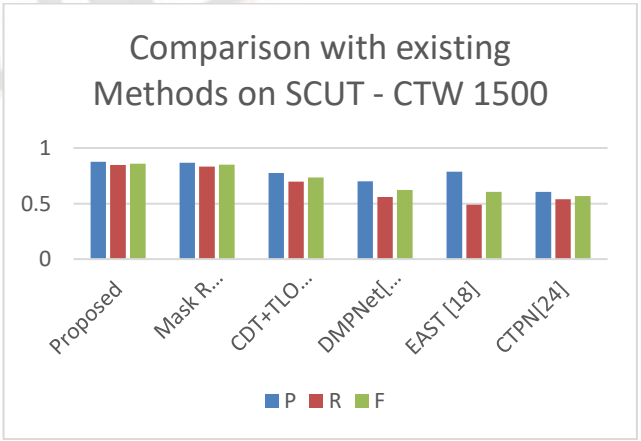

Figure 7: Comparison of Existing Methods on SCUT - CTW 1500 Dataset

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
Article Received: *16 August 2023* ***Revised:*** *08 October 2023* ***Accepted:*** *24 October 2023*

Original Image Ground Truth

Our Method

Figure 8: Comparisons on SCUT - CTW 1500

VIII.ABLATION STUDY

Within this section, we delve into an examination of the factors that impact our model's performance.

A. ROI pooling Vs ROI Align

RoI Pooling and RoI Align are object identification and instance segmentation algorithms. RoI Pooling splits a RoI into fixed-size bins, aggregates features using max-pooling, and is computationally efficient, however, it has misalignment difficulties and lacks differentiability. RoI Align, on the other hand, uses bilinear interpolation to precisely compute features inside fine-scale bins, reducing misalignment issues and preserving exact spatial information. It is differentiable, which makes it acceptable for end-to-end training, but it is more computationally costly. The decision between both strategies is determined by the accuracy and computational cost requirements of the individual work, with RoI Align being preferred for high-precision applications such as instance segmentation, while RoI Pooling may serve for simpler tasks using aligned RoI's.

RoI Pooling separates RoI's into fixed bins, which can cause misalignment issues, especially when RoI borders do not completely match with the grid. This mismatch might lead to the loss of spatial information, lowering detection accuracy. Furthermore, RoI Pooling is not differentiable, making it unsuitable for smooth incorporation into end-to-end deep learning model training.

RoI Align, on the other hand, provides an appealing option. By computing features inside fine-scale bins with sub-pixel precision, it excels at maintaining spatial information with amazing accuracy. This significantly reduces misalignment difficulties, making it ideal for activities with irregular or non-axis-aligned RoI's. Furthermore, RoI Align is distinguishable, making it a feasible option for end-to-end deep learning model training.

B. Sample Ablation

Deactivating the image correction sampler will have a notable impact on tasks that require accurate geometric adjustments and image alignment. Image correction is a common technique used to rectify distorted or non-rigid objects within images, and its importance varies based on the application and the characteristics of the input data. For instance segmentation tasks, in particular, image correction plays a pivotal role. These tasks involve segmenting individual instances of objects in images, and without image correction, misalignment issues can arise, resulting in imprecise instance segmentation outcomes. This misalignment challenge becomes especially pronounced when dealing with overlapping instances or objects with irregular shapes.

C. PBTPN Abalation

PBTPN removal from the architecture can have significant consequences:

a) Feature Extraction: Without PBTPN, feature extraction is entirely dependent on the ResNet50 backbone. As a result, the model can only extract features at a restricted number of resolutions. PBTPN removal may result in a reduced range of characteristics, thus limiting the model's ability to gather information from tiny or multi-oriented text.

b) Text Detection ability: The absence of PBTPN may have an effect on the model's text detection ability. PBTPN is designed specifically to improve feature representation and feature map resolution for tiny text targets. The model's ability to recognize text in photos with varying font sizes, orientations, and backgrounds may be jeopardized without PBTPN.

IX. CONCLUSION

The paper describes a brand-new method for text detection that makes use of Mask Scoring R-ConvNN. The versatility of this method makes it unique since it can distinguish between and manage curved text and photos of multi-oriented scenes at the same time. Given the diversity and complexity of text in natural landscapes, flexibility is a key quality.

This study makes a significant contribution by creating the PBTPN as a fresh backbone system for mask scoring R-ConvNN. The model's feature extraction skills are improved, making it more reliable and accurate for the difficult task of text detection. The usefulness of the suggested strategy is thus demonstrated by the success in lowering false alarms that frequently arise as a result of backdrops that resemble text.

The technique's performance was assessed on two benchmark datasets, SCUT-CTW1500 (for curved text) and ICDAR-2015 (for multi-oriented text). Precision (P), Recall (R), and the F1-score (F) all achieved values of 0.912, 0.828, and 0.867 in the reported performance metrics, demonstrating the model's ability to correctly detect text in a variety of contexts.

Despite the fact that this work represents a substantial advancement in text detection, it is crucial to recognize its shortcomings. There are some areas that need more investigation and improvement, like with any research. Future studies will address these limitations, as the article acknowledges. This dedication to constant development and improvement is a typical and essential component of scientific inquiry.

Experiments on scene text detection benchmarks clearly demonstrate the proposed method's remarkable performance. Several intriguing options for future study need investigation. To begin, we plan to investigate the possibility of training the Transformation Scaling Extension algorithm alongside the network in an end-to-end method. Second, we intend to apply the Transformation Scaling Extension technique to larger instance-level segmentation tasks, particularly in benchmarks with a high density of object instances.

In conclusion, this study presents a text detection method that excels at handling curved and multi-oriented text in natural scene images. PBTPN's use as a backbone network improves feature extraction and lowers false alarm rates. The approach's effectiveness is highlighted by the published performance metrics on common benchmark datasets, and the recognition of its limitations provides the path for further development in this area.

REFERENCES

[1] Shelhamer, Evan, Jonathan Long, and Trevor Darrell. "Fully convolutional networks for semantic segmentation." IEEE Trans. Pattern Anal. Mach. Intell. 39, no. 4 (2017): 640-651.

[2] Ren, Shaoqing, Kaiming He, Ross Girshick, and Jian Sun. "Faster r-cnn: Towards real-time object detection with region proposal networks." Advances in neural information processing systems 28 (2015).

[3] Liu, Wei, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. "Ssd: Single shot multibox detector." In Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14, pp. 21-37. Springer International Publishing, 2016.

[4] Matas, Jiri, Ondrej Chum, Martin Urban, and Tomás Pajdla. "Robust wide-baseline stereo from maximally stable extremal regions." Image and vision computing 22, no. 10 (2004): 761-767.

[5] Epshtein, Boris, Eyal Ofek, and Yonatan Wexler. "Detecting text in natural scenes with stroke width transform." In 2010 IEEE computer society conference on computer vision and pattern recognition, pp. 2963-2970. IEEE, 2010.

[6] Zhang, Zheng, Chengquan Zhang, Wei Shen, Cong Yao, Wenyu Liu, and Xiang Bai. "Multi-oriented text detection with fully convolutional networks." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4159-4167. 2016.

[7] Yao, Cong, Xiang Bai, Nong Sang, Xinyu Zhou, Shuchang Zhou, and Zhimin Cao. "Scene text detection via holistic, multi-channel prediction." arXiv preprint arXiv:1606.09002 (2016).

[8] Huang, Lichao, Yi Yang, Yafeng Deng, and Yinan Yu. "Densebox: Unifying landmark localization with end to end object detection." arXiv preprint arXiv:1509.04874 (2015).

[9] Redmon, Joseph, Santosh Divvala, Ross Girshick, and Ali Farhadi. "You only look once: Unified, real-time object detection." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 779-788. 2016.

[10] Girshick, Ross, Jeff Donahue, Trevor Darrell, and Jitendra Malik. "Rich feature hierarchies for accurate object detection and semantic segmentation." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 580-587. 2014.

[11] Deng, Dan, Haifeng Liu, Xuelong Li, and Deng Cai. "Pixellink: Detecting scene text via instance segmentation." In Proceedings of the AAAI conference on artificial intelligence, vol. 32, no. 1. 2018.

[12] Yang, Qiangpeng, Mengli Cheng, Wenmeng Zhou, Yan Chen, Minghui Qiu, Wei Lin, and Wei Chu. "Inceptext: A new inception-text module with deformable psroi pooling for multi-oriented scene text detection." arXiv preprint arXiv:1805.01167 (2018).

[13] Dai, Yuchen, Zheng Huang, Yuting Gao, Youxuan Xu, Kai Chen, Jie Guo, and Weidong Qiu. "Fused text segmentation networks for multi-oriented scene text detection." In 2018 24th international conference on pattern recognition (ICPR), pp. 3604-3609. IEEE, 2018.

[14] Praneel, AS Venkata, and T. Srinivasa Rao. "Scene Text Detection Using Pyramid-Based Text Proposal Network and Transformation Component Network." (2023).

[15] Li, Hanchao, Pengfei Xiong, Jie An, and Lingxue Wang. "Pyramid attention network for semantic segmentation." arXiv preprint arXiv:1805.10180 (2018).

[16] He, Wenhao, Xu-Yao Zhang, Fei Yin, and Cheng-Lin Liu. "Deep direct regression for multi-oriented scene text detection." In Proceedings of the IEEE international conference on computer vision, pp. 745-753. 2017.

[17] Liao, Minghui, Baoguang Shi, Xiang Bai, Xinggang Wang, and Wenyu Liu. "Textboxes: A fast text detector with a single deep neural network." In Proceedings of the AAAI conference on artificial intelligence, vol. 31, no. 1. 2017.

[18] Zhou, Xinyu, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, and Jiajun Liang. "East: an efficient and accurate scene text detector." In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pp. 5551-5560. 2017.

[19] Liu, Yuliang, and Lianwen Jin. "Deep matching prior network: Toward tighter multi-oriented text detection." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1962-1969. 2017.

[20] Jaderberg, Max, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. "Reading text in the wild with convolutional neural networks." International journal of computer vision 116 (2016): 1-20.

[21] Zhong, Zhuoyao, Lianwen Jin, and Shuangping Huang. "Deeptext: A new approach for text proposal generation and text

detection in natural images." In 2017 IEEE international conference on acoustics, speech and signal processing (ICASSP), pp. 1208-1212. IEEE, 2017.

[22] Gupta, Ankush, Andrea Vedaldi, and Andrew Zisserman. "Synthetic data for text localisation in natural images." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2315-2324. 2016.

[23] Shi, Baoguang, Xiang Bai, and Serge Belongie. "Detecting oriented text in natural images by linking segments." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2550-2558. 2017.

[24] Tian, Zhi, Weilin Huang, Tong He, Pan He, and Yu Qiao. "Detecting text in natural image with connectionist text proposal network." In Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII 14, pp. 56-72. Springer International Publishing, 2016.

[25] Dai, Jifeng, Yi Li, Kaiming He, and Jian Sun. "R-fcn: Object detection via region-based fully convolutional networks." Advances in neural information processing systems 29 (2016).

[26] Yao, Cong, Xiang Bai, Nong Sang, Xinyu Zhou, Shuchang Zhou, and Zhimin Cao. "Scene text detection via holistic, multi-channel prediction." arXiv preprint arXiv:1606.09002 (2016).

[27] Dai, Jifeng, Kaiming He, Yi Li, Shaoqing Ren, and Jian Sun. "Instance-sensitive fully convolutional networks." In Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VI 14, pp. 534-549. Springer International Publishing, 2016.

[28] Li, Yi, Haozhi Qi, Jifeng Dai, Xiangyang Ji, and Yichen Wei. "Fully convolutional instance-aware semantic segmentation." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2359-2367. 2017.

[29] He, Kaiming, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. "Mask r-cnn." In Proceedings of the IEEE international conference on computer vision, pp. 2961-2969. 2017.

[30] Chen, Liang-Chieh, Alexander Hermans, George Papandreou, Florian Schroff, Peng Wang, and Hartwig Adam. "Masklab: Instance segmentation by refining object detection with semantic and direction features." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4013-4022. 2018.

[31] Liang, Xiaodan, Liang Lin, Yunchao Wei, Xiaohui Shen, Jianchao Yang, and Shuicheng Yan. "Proposal-free network for instance-level object segmentation." IEEE transactions on pattern analysis and machine intelligence 40, no. 12 (2017): 2978-2991.

[32] Kirillov, Alexander, Evgeny Levinkov, Bjoern Andres, Bogdan Savchynskyy, and Carsten Rother. "Instancecut: from edges to instances with multicut." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 5008-5017. 2017.

[33] Bai, Min, and Raquel Urtasun. "Deep watershed transform for instance segmentation." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 5221-5229. 2017.

[34] Venkata Praneel, A. S., T. Srinivasa Rao, and M. Ramakrishna Murty. "A survey on accelerating the classifier training using various boosting schemes within cascades of boosted ensembles." In Intelligent Manufacturing and Energy Sustainability: Proceedings of ICIMES 2019, pp. 809-825. Springer Singapore, 2020.

[35] Bookstein, Fred L. "Principal warps: Thin-plate splines and the decomposition of deformations." IEEE Transactions on pattern analysis and machine intelligence 11, no. 6 (1989): 567-585..

[36] Jaderberg, Max. "karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks." In Proc. of Neural Information Processing Systems. 2015.

[37] Karatzas, Dimosthenis, Lluis Gomez-Bigorda, Anguelos Nicolaou, Suman Ghosh, Andrew Bagdanov, Masakazu Iwamura, Jiri Matas et al. "ICDAR 2015 competition on robust reading." In 2015 13th international conference on document analysis and recognition (ICDAR), pp. 1156-1160. IEEE, 2015.

[38] Yuliang, Liu, Jin Lianwen, Zhang Shuaitao, and Zhang Sheng. "Detecting curve text in the wild: New dataset and new solution." arXiv preprint arXiv:1712.02170 (2017).

[39] Huang, Zhida, Zhuoyao Zhong, Lei Sun, and Qiang Huo. "Mask R-CNN with pyramid attention network for scene text detection." In 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 764-772. IEEE, 2019.

[40] Jiang, Yingying, Xiangyu Zhu, Xiaobing Wang, Shuli Yang, Wei Li, Hua Wang, Pei Fu, and Zhenbo Luo. "R2CNN: Rotational region CNN for orientation robust scene text detection." arXiv preprint arXiv:1706.09579 (2017).

[41] Ma, Jianqi, Weiyuan Shao, Hao Ye, Li Wang, Hong Wang, Yingbin Zheng, and Xiangyang Xue. "Arbitrary-oriented scene text detection via rotation proposals." IEEE transactions on multimedia 20, no. 11 (2018): 3111-3122.