Scene Text Detection Using Pyramid-Based Text Proposal Network and Transformation Component Network

A.S. Venkata Praneel

Department of Computer Science and Engineering, GST
GITAM (Deemed-to-be University).
Visakhapatnam-530045, AP, India
praneelsri@gmail.com

Dr.T. Srinivasa Rao

Department of Computer Science and Engineering, GST
GITAM (Deemed-to-be University).
Visakhapatnam-530045, AP, India
sthamada@gitam.edu

Abstract

Text identification and Text Recognition in Natural Scene Text importance is growing attention in recent research. Scene text contains a wealth of semantic information that can be used in a broad spectrum of vision-oriented applications. Text Recognition and Detection in the image of a natural scene play a vital role in accessing information and understanding the environment. Scene text orientations comprise horizontal, arbitrarily, curved, and vertically oriented scene texts. In this study, a Mask Scoring R-ConvNN based text detection strategy in this research can reliably detect curved text and multi-oriented from real input images. For learning good capability mask scores for the expected instance, a Mask Scoring R-ConvNN network frame is employed in this model. The mask scoring system corrects the misalignment between mask quality and score while also improving instance segmentation performance by emphasizing more accurate mask predictions. Using a Pyramid-based Text Proposal Network (PBTPN) and a Transformation Component Network (TCN) to improve the Feature extraction capabilities of Mask Scoring R-ConvNN for text identification and segmentation. Studies showed that Pyramid Networks are more successful at suppressing false alarms triggered by text-like backgrounds. By employing testing on a scale and with a single model, this strategy can obtain higher performance on Multi Oriented text and curved text benchmark datasets.

***Keywords*: Text Detection, Instance Segmentation, Transformation Component Network, Pyramid-based Text Detection Network, Mask Scoring R-ConvNN.**

1. Introduction

The method of finding text in an image and enclosing it with a rectangular box is known as text detection. There are two types of techniques to detect text. One is an image-based approach and the second one is a frequency-based approach. An image is split into several segments into image-based approaches. Each segment is made up of pixels with identical features that are joined together. To organize and shape the text, statistical properties of related components are used. To categorize the components into text and non-text, machine learning techniques such as support vector machines and convolutional neural networks are utilized. The high-frequency coefficients are extracted using discrete Fourier transform (DFT) or discrete wavelet transform (DWT) in frequency-based approaches. The text in an image is believed to have high-frequency components and picking just the high-frequency coefficients separates the text from the non-text regions. In recent years, increasing emphasis has been placed on the task of making deep neural networks knowledgeable of their predictive performance in the context of picture segmentation. For instance segmentation, most techniques employ a valid estimation of instance segmentation as the mask's quality score. This approach is based on the mask quality, which is not accurately related to the classification score as a quantified instance mask. Huang et al. looked through the problem and came out with the Mask Scoring R-ConvNN [1]. The object identification model paired with sliding window technology gained a lot of traction and quick growth by generating predictive bounding boxes with dense and regular spatial meshes. As a result, Chen et al. introduced the Tensor Mask [2] paradigm, which used dense sliding windows to achieve instance segmentation. This is a field that is still in its infancy.

Data analysis, retrieval of information, OCR translation, robot navigation, and augmented reality all benefit from Scene Text Detection and localization, the Computer Vision community is becoming increasingly interested in it. [3,4,5,6,7]. Text identification and localization in natural scene images are still a completely unanswered question due to a variety of text variations in colors, fonts, orientations, languages, and scales, as well as immensely complicated backgrounds that look like text, in addition to some disruptions and non-uniform lighting, poor contrast, limited picture resolution, and occlusion are all artifacts created during image capture.

With the incredible advancement of deep learning, tremendous improvement was done in this discipline. Many Convolutional Neural Networks (ConvNN) based object identification and segmentation frameworks are borrowed, to overcome the text detection problem, including Faster R-CNN (FRCNN) [8], SSD [9], and Fully Convolutional Networks (FCN) [10], which significantly outclass MSER [11] or SWT [12] based bottom-up text identification strategies. There are, some techniques [13,14] that treat text recognition as a semantic segmentation issue and use an FCN to produce text/non-text predictions, which may then be used to construct a text saliency map. Because the saliency map can indeed recognize coarse text blocks, complicated post-sequencing processes are required to extract precise Bounding boxes for the text.

Not as FCN-based approaches, another class of techniques views the text as one distinct object and uses frameworks such as R-CNN (RCNN) [15], FRCNN [8], SSD [9], YOLO [16], and DenseBox [17] to automatically detect words or text from images. Even though these techniques are made up of simpler pipelines, they also have trouble detecting curved text. Some newer techniques, such as PixelLink [18], FTSN [19], and IncepText [20], attempt to construct text detection as an instance segmentation challenge, allowing for the recognition of both straight and curved text in a single framework. PixelLink offers to identify text by connecting pixels inside the exact text instances collectively, whereas [19,20] tackles the text detection problem using the FCIS framework (FCISF) [21].

This paper used a Mask Scoring R-ConvNN based text detection system in this research that can recognize Curved Text and Multi-oriented from real Scene Images in a homogeneous way. In this regard, to improve Mask Scoring R-ConvNN's capabilities, have used the Pyramid Based Text Proposal Network, which is derived from Attention Pyramid Network (APN) [22] as a new backbone network and transformation component network with it. Experiments show that APN is more successful at limiting false alarms triggered by text like backgrounds on ICDAR-2015 [23] and SCUT-CTW1500 [24] identification benchmark datasets, the proposed technique outscored the competition.

2. Related work

In this current section, the recently proposed text identification algorithms using CNN, as well as recent advances in instance segmentation problems are studied in the relevant work done by authors in the same area.

2.1. *Text Detection*

In recent years, Convolutional Neural Network-based Object Detection are vastly applied to tackle the challenge of Text Detection. The algorithms [13,14] take a page from semantic segmentation and use an FCN to estimate text/non-text at the pixel level, resulting in a text transmission map for text detection. However, the method by [13,14] only recognizes coarse text blocks. Hence, sophisticated post-sequencing techniques are required to extract correct Bounding boxes of text. Different approaches by researchers handle text as one distinct object and use cutting-edge Object Detection techniques to immediately detect lines of text or words from images [3,4,6,7,25,26,27,28]. RCNN was developed for text detection by Jaderberg et al. [25]; however, the conventional region proposal generating techniques constrained its performance. Gupta et al. [26] used the You only look ones [YOLO] to accomplish text identification and Bounding Box Regression (BBR) at every scaling and location in an Image. To address the word-level horizontal text issue, [4,27] used the FRCNN and SSD frameworks. Accordingly, [7,28] offered Quadrilateral anchors to look for inclined text suggestions that might best fit multi-oriented text examples to expand FRCNN and SSD. [6] and [3] adapted the notion of DenseBox and employed a one-stage FCN to generate pixel-wise texts scores along with Quadrilateral bounding boxes over every scale and orientations of an Image to answer the inefficiencies of the anchor system [3]. Even though these techniques are made up of smaller pipelines, they still have trouble detecting curved text. [5,29] used Object Detection methods to identify text portions first, then gathered selected text portions from Words or Lines using basic text-line clustering techniques or gathering material on connections, respectively, instead of identifying full words or text lines immediately.

These techniques can be used to identify Curved text automatically, however, they add to the total complexity significantly. Text detection was formulated as an instance segmentation task by [5,29], who applied Object identification algorithms to identify text portions first, then sorted these text portions from Words or Lines. Deng et al. [18] proposed connecting pixels inside the same text occurrences with each other to detect text. To tackle the text detection issue, [19,20] used the FCISF [21]. To improve the performance of Text Detection performance in this research, have used Mask Scoring R-ConvNN, and it is the most recent instance segmentation technique.

2.2. *Instance Segmentation*

There are majorly two types of instance segmentation techniques now in use. The first is focused on detection, and the second is focused on segmentation. Detection-based approaches use detectors to determine the section of every instance and afterward forecast the mask for every section, such as FRCNN [30], and R-FCN [31]. In a sliding window approach,[32] proposes DeepMask to segment and categorize the central object. Instance-sensitive FCNs are proposed by Dai et al. [33] to build position-sensitive maps, which are merged to get the final masks. The instance segmentation results are obtained by FCISF [21] using position-sensitive layouts with inside/outside scores. Mask R-CNN (MRCNN) is proposed by [34], which is developed on the basis of FRCNN and adds an instance-level semantic segmentation branch. Chen et al. [35] offer MaskLab, which is developed on MRCNN [36] and employs position-sensitive scores to improve outcomes. However, one flaw in these systems is that mask quality is only determined by classification scores, which leads to the concerns mentioned above.

Predicting the category labels of each pixel first, then grouping those to generate instance segmentation results, is how segmentation-based algorithms work. To cluster the pixels, Liang et al. [37] employ spectral clustering. Other studies, such as [38,39], include border detection data in the clustering process. Bai et al. [40] employed watershed algorithms to categorize pixels and forecast pixel-level energy levels. Some recent studies [41,42,43,44] have used metric learning to learn pixel embedding. These approaches train an embedding for each pixel to ensure that pixels from the same instance have identical embeddings. After that, the learned embedding is clustered to provide the final instance labels. Because these algorithms lack explicit ratings to assess the quality of an instance mask, they must rely on averaged pixel-level categorization scores instead.

The alignment between mask score and mask quality is ignored in both groups of the above approaches. Because mask score seems unreliable, a mask hypothesis with a greater IoU versus ground truth is at risk of being given low priority if someone has a low mask score. As a result, the final average precision is degraded in this instance.

For each example, all the accurate features of the image, as well as exact segmentation, are included, that is why Instance Segmentation is considered a difficult problem. Dai et al. [45] suggested a multi-stage process that anticipates segment proposals from bounding-box suggestions and then classifies them. For FCISF, [21] integrated the portion proposal method in [33] with R-Fully Convolutional Network [19]. FCISF, despite its speed, makes systematic mistakes on overlapping instances and generates erroneous edges [34]. FRCNN [8] was recently expanded by MRCNN [34], which included a branch for predicting an object mask in addition to the current branch for bounding box identification. It used ResneXt [46] as the basic network and included RoIAlign [34] to replace RoIPool [47] to correct the pixel mismatch. Besides, it is using the FPN [48], to improve feature representation and partly solve the small object detection problem. Here we used a Pyramid-based text detection Network in Mask Scoring R-ConvNN architecture to improve the feature representation capabilities. Experiments show that Pyramid Network is much more successful at suppressing false alarms triggered by text-like backgrounds.

2.3. *Mask Scoring R-ConvNN*

For most instance segmentation frameworks, every instance classification is regarded as a score reflecting the confidence of mask quality for instance tasks. Yet, there seems to be no evidence of a correlation between MaskIoU's quality and the classification score. In Mask Scoring R-ConvNN, a novel branch is proposed that combines the instance feature with the associated projected mask to infer the MaskIoU. The discrepancy in the Mask Quality and Mask Score is corrected by the Mask Score system. The effectiveness of instance segmentation has increased because of the more precise Mask predictions being prioritized during data analysis.

3. Proposed Method

3.1. *Framework*

Figure 1 depicts the whole structure of the supplied approach. A backbone Pyramid Based Text Proposal Network, a Region Proposal Network (RPN) to create Text candidate boxes, a framework for BBR, a MaskIoU head branch, and a Masked branch for text Instance Segmentation make up the framework. During the learning phase, the proposed technique generates many text candidate boxes using RPN and then links the ROI features with the anticipated mask as the MaskIoU head's input. The candidate boxes RoI features are forwarded to the Fast R-CNN branch. The Mask branch creates a Text candidate box and Text Instance Segmentation map, which are accurate.

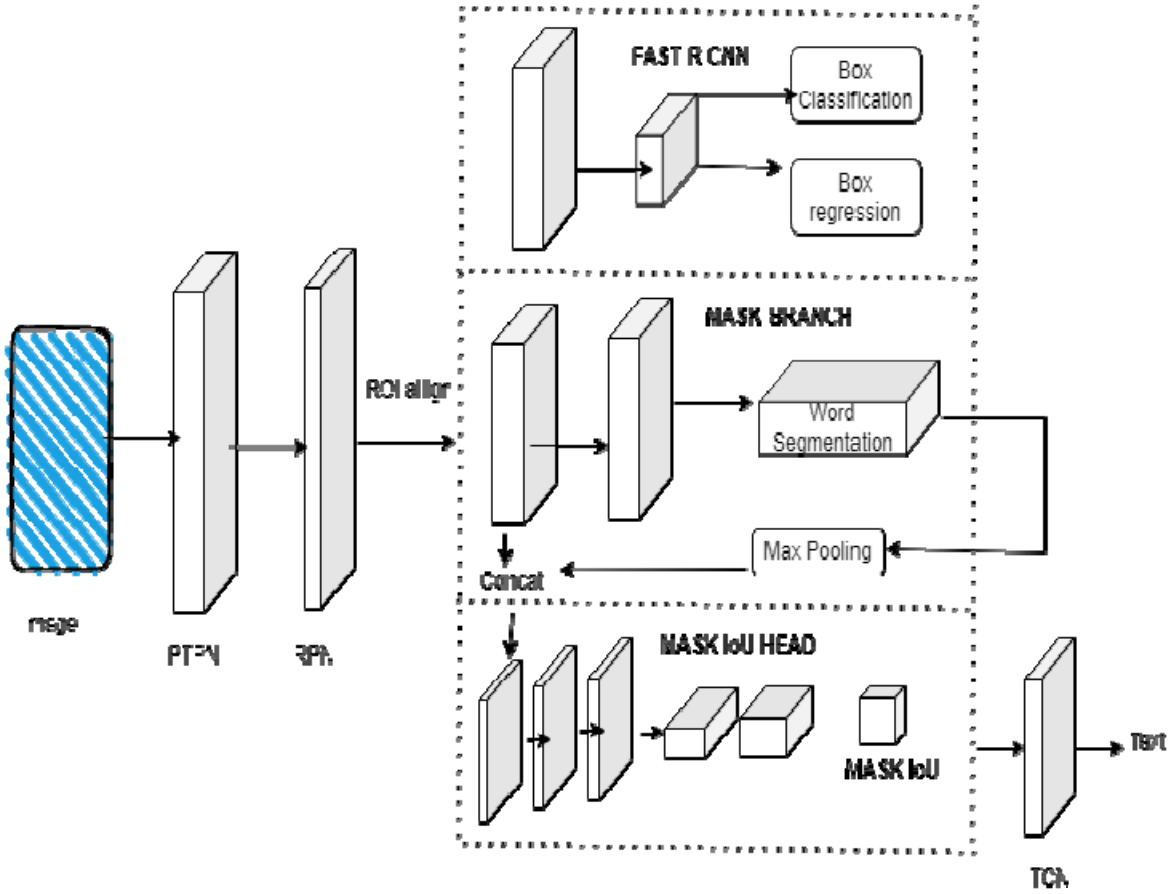

Figure 1: Structure of the Method

3.1.1 Backbone

In the natural image, its text size changes significantly. We use a ResNet50 backbone with just a Pyramid Based Text Proposal Network backbone to build a map with a lot of semantic features of all sizes. But after the top-down model, the Pyramid-based Text Proposal network may use the context information to combine features from different resolutions to uniform scale input. Pyramid-based Text Detection network promotes feature map resolution for tiny targets, such as text, by operating on bigger feature maps, allowing us to obtain more pertinent information regarding small targets.

3.1.2 RPN:

RPN generates text recommendations for the Fast R-CNN branch as well as the mask branch. In this method, allocate the anchors to various phases based on their sizes. The sizes of distinct anchors are specifically stated as 32 x 32, 64 x 64, 128 x 128, 256 x 256, 512 x 512 pixels on five phases P2; P3; P4; P5; P6 accordingly. All of the targets can't be regular squares. As a consequence, At each phase, we adjusted the aspect ratios to 1:2, 1:1, and 2:1; now, the pyramid structure has 15 anchors in total. As a result, RPN can deal with areas of any size or aspect ratio. For acquiring the regional proposal, have employed RoI Align rather than RoI Pooling. RoI Align

saves more exact position information, which is important for the mask branch's segmentation assignment. A 3 × 3 Conv layer is suitable for Feature extraction in classic object detection situations. However, the majority of text items in nature scenes are rectangular, with relatively great length and breadth. Text characteristics cannot be extracted using the standard 3 × 3 convolution layer. The RPN regression process novelty convolution kernel is based on this. To extract anchors with varied aspect ratios, we include a 1 × 5 convolution layer and then a 5 × 1 convolution layer just after the usual 3 × 3 convolutions.

3.1.3 Fast R-CNN

The classification and regression tasks are now the most important in the FRCNN branch. Its principal goal is to construct the shortest bounding rectangle box and provide more accurate bounding boxes for detection. RoI Align, which is derived from region suggestions, provides a feature map with a resolution of 7 × 7 pixels, which is then submitted to the Fast R-CNN branch, and believe that a 7 × 7 map is adequate for detecting.

3.1.4 Mask Branch

Mask branch's major responsibility is to instance segment the total text. Having an RoI with a specified size of 16 × 64 as an input. The mask branch predicts two types of maps: a global text instance map and a backdrop map, using four Conv layers and one deConv layer. The Text instance map can show the precise location of a text area, not referring to the text instance's form. Character regions for post-processing are not included in the character backdrop map.

3.1.5 MaskIoU head

This MaskIoU head decides to regress the IoU produced by the expected Mask and its Ground truth mask. It passes the feature from the RoI Align layer along with the predicted mask to the MaskIoU head.

3.1.6 Transformation Component Network

With a predicted 2D transformation, the transformation component corrects an input image.

3.2. *PBTPN*

The Pyramid Based Text Proposal Network (PBTPN) is made of two segments: a Pyramid Network (PN) and an Up-sample Unit (USU) segment. The PN segment aims to combine pooling with spatial pyramid focus to learn better high-level image features based on high-level characteristics.

The USU segment is installed on every decoder layer to give global context and low-level feature selection assistance for category localization specifics. PBTPN delivers the best segmentation accuracy because of these deft innovations. We plan to employ PBTPN as a new Backbone Network for the Mask Scoring R-Convolutional Neural Network-based text detection model to increase feature representation learning.

PBTPN is based on ResNet50 [49] and ResNeXt50 [46]. PBTPN implementations mainly follow [22] with a few minor variations. The PN module, as shown in Figure 2, uses the output features of the Res-4 layers in ResNet50 or ResNeXt50 as input and executes 3 × 3 Dilated convolutions for 3, 6, and 12 example rates to capture Context information more effectively. After that, a 1 × 1 convolution layer is used to concatenate the three feature maps and minimize their dimension. Following that, PN executes a further 1 × 1 conv on the input Res-4 features, the pixel-wise outcome of these is multiplied only with previous context characteristics. To achieve overall pyramid attention features, the extracted features are combined with the output features of the global pooling branch. As demonstrated in Figure. 3, the USU module performs 3 × 3 Conv on Low-level

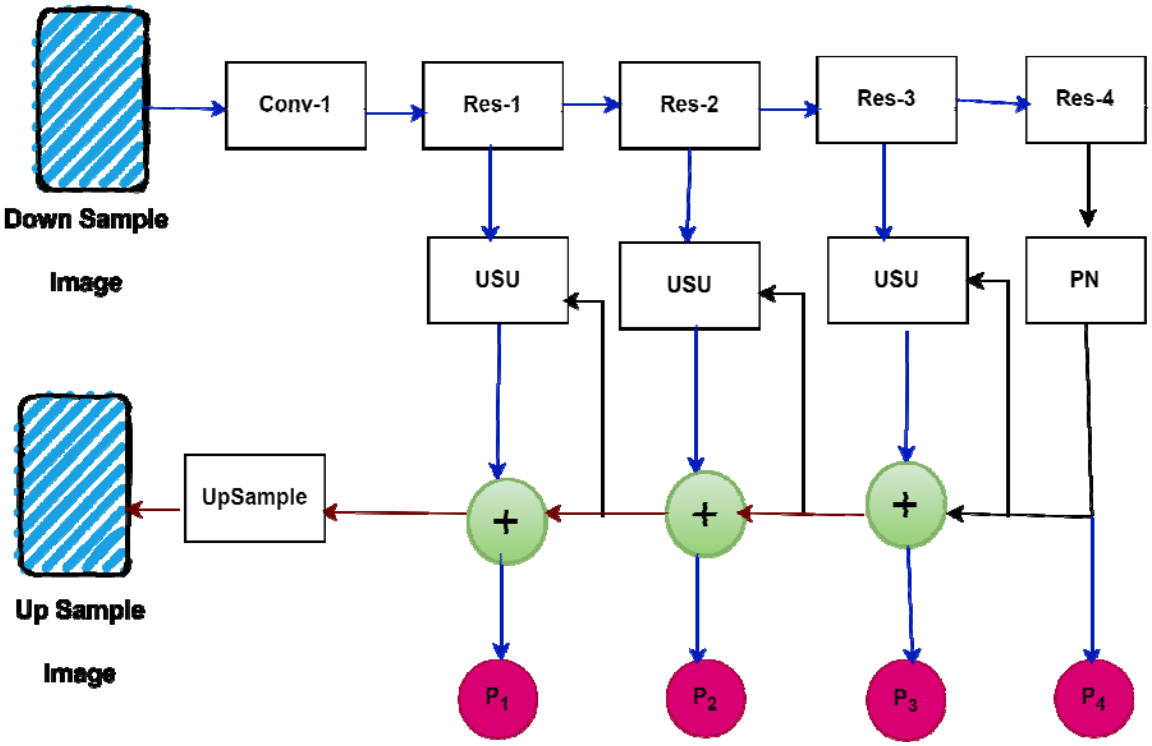

Figure 2: Structure of the PBTPN

features to minimize channels in CNN Feature maps. The context created from High-level features is multiplied by the Low-level features after a 1 × 1 Conv with Instance Normalization [50] and ReLU Nonlinearity. Finally, the USU features are generated by combining the high-level features after upsampling them with the weighted low-level features. A strong feature pyramid was built with three tiers using the PN and USU modules, namely P1, P2, P3, and P4, with strides of 2,4, 8, and 16, respectively.

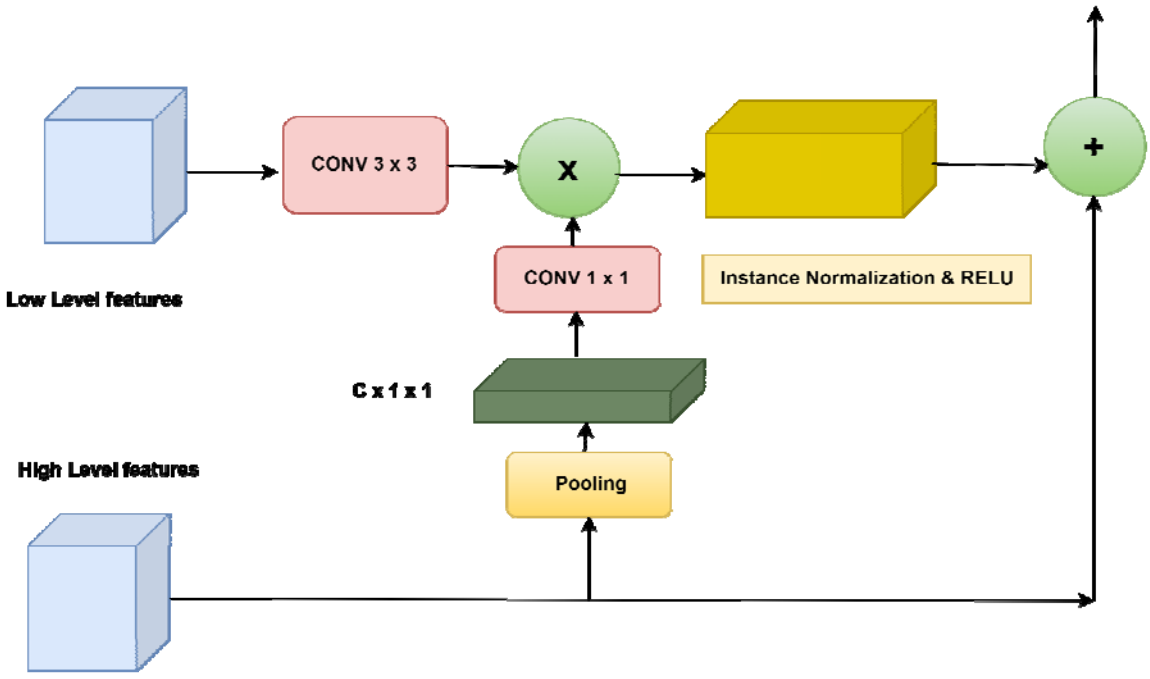

Figure 3: Structure of the USU

3.3. *Transformation Component Network*

With a predicted 2D transformation, the transformation component corrects an input image. The transformation we use is Thin-Plate-Spline (TPS) [52]. TPS has a lot of uses in image transformation and matching, for example, [51]. It has more flexibility than other 2D transformations like affine and projective. TPS deforms images in a non-rigid manner, allowing it to handle a wide range of distortions. Figure 3 shows several common rectifications. TPS can correct both perspective and curved text, which are two common kinds of irregular text. The Spatial Transformer Network (STN) [53] is used to create the transformation component. As a learnable network layer, STN is based on the concept of spatial transformation modelling. The design of the transformation component is shown in Figure 4. The network's localization network predicts a collection of control points initially. The control points are then used to compute a TPS transformation, which is then passed to the grid component and sampler to create the corrected image It. The transformation component requires no further inputs except the input picture because the control points are anticipated from 'I'.

3.3.1 Localization Component

To begin, we'll look at Figure 4 to see how TPS corrects text. Control points which are two sets of the same size, represented by C, determine a TPS transformation. In the output picture, the control points are evenly spaced at fixed places along the Top and Bottom image boundaries. Once the control points upon this input image are predicted along the top and bottom text borders, the TPS transformation provides a transformed image with regular text. As a result, anticipating the control locations on the input Image is the crux of the text rectification challenge. A CNN is used to make the prediction. Assume C control points on both I and I_t, with C_p' and C_p, respectively, as their coordinates. C' is regressed directly from I_s, it is down sampled from 'I' by the localization component. All modules in the transformation component are differentiable, as we'll see later. For that, the localization component is trained only using back-propagated gradients during training, eliminating the need for manual annotations on the Control points.

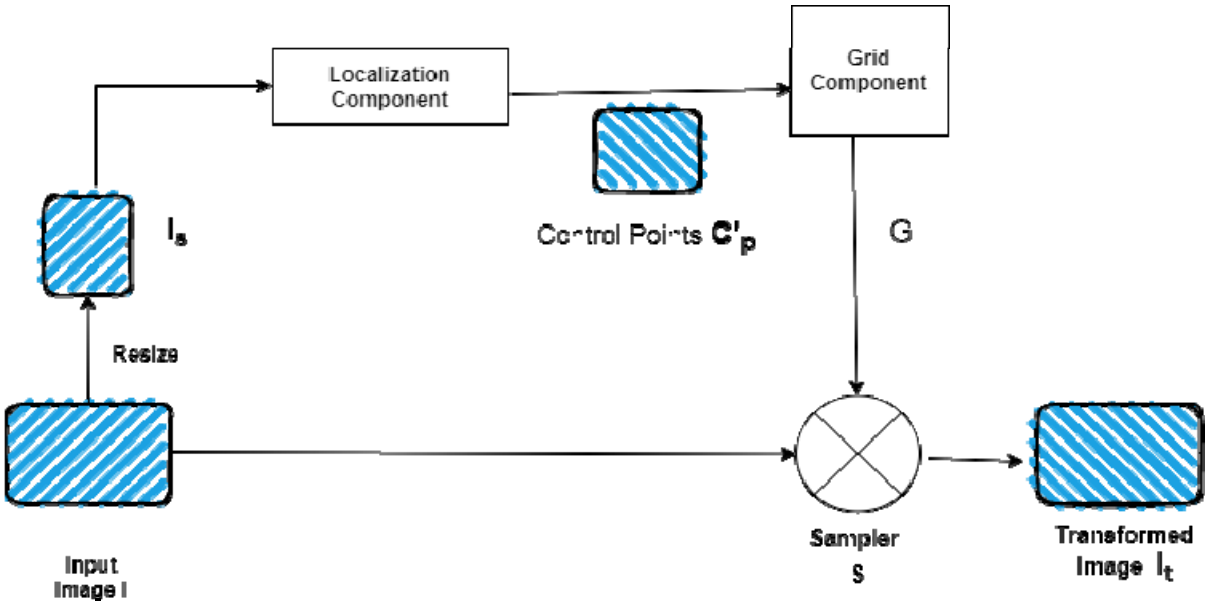

Figure 4: Structure of the Transformation Component Network

3.3.2 Grid Component

This grid component generates a sample grid $G = \{G_i\}$ on I_t by computing a modification and applying that to every pixel position in I_t. The TPS solution and transformation process may easily be described as a neural network module, as shown in Figure 4. It accepts as inputs C_p' and outputs g' the expected control points C_p' and every pixel position 'p' on the corrected image. Furthermore, the pixel placements in a picture of a specified resolution are fixed. As a consequence, for photos of the same resolution, the calculation from p to p' can be stored and utilized multiple times.

3.3.3 Sampler

The sampler creates the corrected image at the transformation component output:

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

By interpolating the neighbor pixels of p', the sampler calculates the value of p. Because p' might be outside the image area, value clipping is used before sampling to keep the sampling points within the image area. The sampler has been made differently, which means it can backpropagate I_t to p gradients. The differentiable image sampling approach does this.

4. Datasets and Results

Conducted tests and compared the technique to other efficient algorithms on two standard datasets: ICDAR2015, and SCUT-CTW1500, to validate its capabilities.

4.1. *ICDAR2015*

A multi-oriented text detection dataset. ICDAR2015 focused on text detection, which is more surprising, rather than text in certain circumstances, which was the focus of ICDAR2013. It has 1500 Images in the Dataset, where 1000 images are for training and the rest for testing.

4.2. *SCUT-CTW1500*

Is a Curved Text Detection Dataset, including 1,000 Images. In which 500 are used for training and the rest for testing.

Method	**Comparison with existing Methods on ICDAR-2015.**		
	P	**R**	**F**
Proposed	**0.912**	**0.828**	**0.867**
Mask R CNN + PAN[36]	0.908	0.815	0.859
InceptText[20]	0.905	0.806	0.853
FTSN [19]	0.886	0.800	0.841
R2CNN[54]	0.856	0.749	0.825
DDR[3]	0.820	0.800	0.810
EAST [6]	0.832	0.783	0.774
RRPN[28]	0.822	0.732	0.774
Seglink[5]	0.786	0.731	0.749

Table 1: Comparison with existing Methods on ICDAR-2015. P, R, and F stand for Precision, Recall, and F-measure, respectively.

Method	**Comparison with existing Methods on SCUT – CTW 1500**		
	P	**R**	**F**
Proposed	**0.876**	**0.846**	**0.860**
Mask R CNN + PAN[36]	0.868	0.832	0.850
CDT+TLOC[24]	0.774	0.698	0.734
DMPNet[7]	0.699	0.560	0.622
EAST [6]	0.787	0.491	0.604
CTPN[29]	0.604	0.538	0.569

Table 2: Comparison with existing Methods on SCUT - CWT. P, R, and F stand for Precision, Recall and F-measure, respectively.

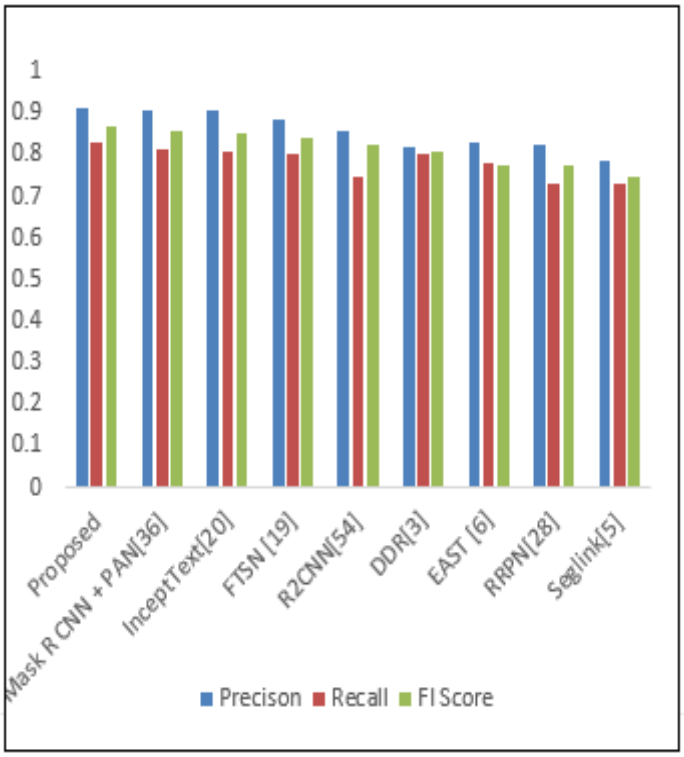

Figure 5: Comparison of Existing Methods on ICDAR 2015 Dataset

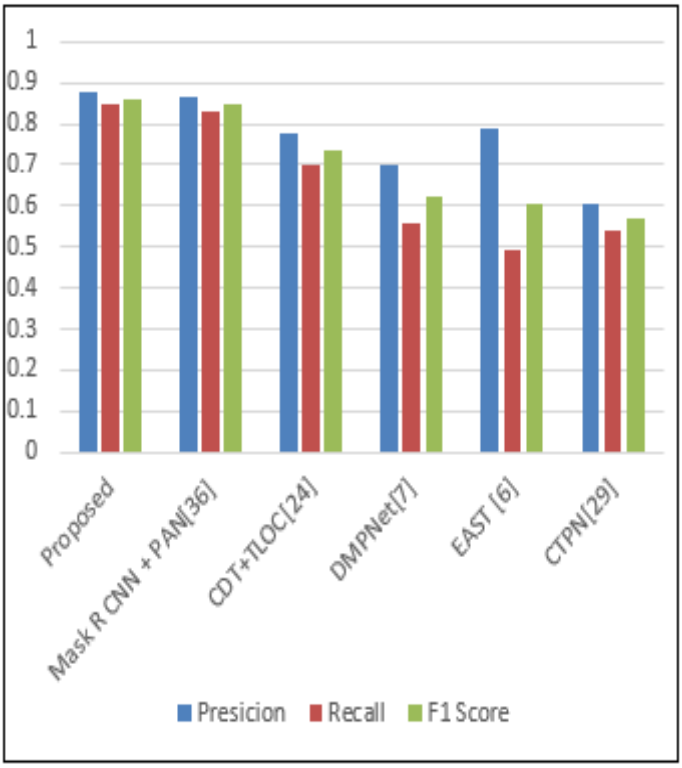

Figure 6: Comparison with Existing Methods on SCUT-CWT Dataset.

5. Conclusion

The paper proposes a technique for text detection based on Mask Scoring R-ConvNN. Because of the adaptability of Mask Scoring R-ConvNN, the proposed technique can distinguish curved text and Multi-oriented Scene Images in a synchronized manner. This shows that using the PBTPN as a new backbone network for Mask Scoring R-ConvNN enhances the feature extraction capabilities of Mask Scoring R-ConvNN and reduces false alarms that are caused by text-like backdrops with P, R, F values as 0.912,0.828,0.867 respectively The Multi-oriented text (ICDAR-2015) and curved text (SCUT-CTW1500) identification benchmark tasks were improved by using only standard and discrete testing. This work has a few limitations, which will be answered in a future study.

Funding

No funding is provided for the preparation of manuscript.

Conflict Of Interest Statement

The authors have no conflicts of interest to declare.

References

[1] Z. Huang, L. Huang, Y. Gong, C. Huang, and X. Wang, Mask Scoring R-CNN, IEEE Conference on Computer Vision and Pattern Recognition (CVPR2019), Long Beach, CA, USA, pp.6409-6418,2019.
[2] X. Chen, R. B. Girshick, K. He and P. Doll´ar, TensorMask: A foundation for dense object segmentation, IEEE/CVF International Conference on Computer Vision (ICCV2019), Seoul, Korea.
[3] W. He, X. Zhang, F. Yin, and C. Liu. Deep direct regression for multi-oriented scene text detection. In ICCV, pages 745–753, 2017.
[4] M. Liao, B. Shi, X. Bai, X. Wang, andW. Liu. Textboxes: A fast text detector with a single deep neural network. In AAAI, pages 4161–4167, 2017.
[5] B. Shi, X. Bai, and S. J. Belongie. Detecting oriented text in natural images by linking segments. In CVPR, pages 3482– 3490, 2017.
[6] X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, and J. Liang. East: An efficient and accurate scene text detector. In CVPR, pages 2642–2651, 2017
[7] Y. Liu and L. Jin. Deep matching prior network: Toward tighter multi-oriented text detection. In CVPR, pages 3454–3461, 2017.
[8] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-CNN: Towards real-time object detection with region proposal networks. In NIPS, pages 91–99, 2015)
[9] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. SSD: Single shot multibox detector. In ECCV, pages 21–37, 2016
[10] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In CVPR, pages 3431– 3440, 2015.
[11] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. Image and Vision Computing, 22(10):761–767, 2004.
[12] B. Epshtein, E. Ofek, and Y.Wexler. Detecting text in natural scenes with stroke width transform. In CVPR, pages 2963– 2970, 2010
[13] Z. Zhang, C. Zhang, W. Shen, C. Yao, W. Liu, and X. Bai.Multi-oriented text detection with fully convolutional networks. In CVPR, pages 4159–4167, 2016
[14] C. Yao, X. Bai, N. Sang, X. Zhou, S. Zhou, and Z. Cao. Scene text detection via holistic, multi-channel prediction. CoRR, abs/1606.09002, 2016.
[15] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In CVPR, pages 580–587, 2014.
[16] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In CVPR, pages 779–788, 2016.
[17] L. Huang, Y. Yang, Y. Deng, and Y. Yu. Densebox: Unifying landmark localization with end to end object detection. arXiv preprint arXiv:1509.04874, 2015.
[18] D. Deng, H. Liu, X. Li, and D. Cai. Pixellink: Detecting scene text via instance segmentation. In AAAI, pages 6773– 6780, 2018.
[19] Y. Dai, Z. Huang, Y. Gao, and K. Chen. Fused text segmentation networks for multi-oriented scene text detection. arXiv preprint arXiv:1709.03272, 2017
[20] Q. Yang, M. Cheng, W. Zhou, Y. Chen, M. Qiu, and W. Lin. Inceptext: A new inception-text module with deformable psroi pooling for multi-oriented scene text detection. arXiv preprint arXiv:1805.01167, 2018.
[21] Y. Li, H. Qi, J. Dai, X. Ji, and Y. Wei. Fully convolutionalinstance-aware semantic segmentation. In *IEEE Inter*national Conference on Computer Vision, pages 4438–4446,2017.
[22] H. Li, P. Xiong, J. An, and L. Wang. Pyramid attention network for semantic segmentation. In BMVC, 2018
[23] D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, S. Ghosh, A. Bagdanov, M. Iwamura, J. Matas, L. Neumann, V. R.Chandrasekhar, S. Lu, et al Icdar 2015 competition on robust reading. In ICDAR, pages 1156–1160, 2015.
[24] Y. Liu, L. Jin, S. Zhang, and S. Zhang. Detecting curve text in the wild: New dataset and new solution. CoRR, abs/1712.02170, 2017.
[25] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman.Reading text in the wild with convolutional neural networks. International Journal of Computer Vision, 116(1):1– 20, 2016.
[26] A. Gupta, A. Vedaldi, and A. Zisserman. Synthetic data for text localization in natural images. In CVPR, pages 2315– 2324, 2016.
[27] Z. Zhong, L. Jin, and S. Huang. Deeptext: A new approach for text proposal generation and text detection in natural images. In ICASSP, pages 1–18, 2017
[28] J. Ma, W. Shao, H. Ye, L. Wang, H. Wang, Y. Zheng, and X. Xue. Arbitrary-oriented scene text detection via rotation proposals. IEEE Transactions on Multimedia, 20(11):3111– 3122, 2018.
[29] Z. Tian, W. Huang, T. He, P. He, and Y. Qiao. Detecting text in natural image with connectionist text proposal network. In ECCV, pages 56–72, 2016.
[30] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In Advances in Neural Information Processing Systems, pages 91–99, 2015.
[31] J. Dai, Y. Li, K. He, and J. Sun. R-fcn: Object detection via region-based fully convolutional networks. In *Advances in* Neural Information Processing Systems, pages 379–387,2016.

[32] C. Yao, X. Bai, N. Sang, X. Zhou, S. Zhou, and Z. Cao. Scene text detection via holistic, multi-channel prediction. CoRR, abs/1606.09002, 2016.
[33] J. Dai, K. He, Y. Li, S. Ren, and J. Sun. Instance-sensitive fully convolutional networks. In *European Conference on Computer Vision*, pages 534–549, 2016.
[34] K. He, G. Gkioxari, P. Doll´ar, and R. Girshick. Mask R-CNN.In *IEEE International Conference on Computer Vision*, pages 2980–2988, 2017.
[35] L.-C. Chen, A. Hermans, G. Papandreou, F. Schroff,P. Wang, and H. Adam. Masklab: Instance segmentation by refining object detection with semantic and direction features.arXiv preprint arXiv:1712.04837, 2017.
[36] Zhida Huang, Zhuoyao Zhong, Lei Sun, Qiang Huo, Mask R-CNN with Pyramid Attention Network for Scene Text Detection, arXiv:1811.09058.
https://doi.org/10.48550/arXiv.1811.09058
[37] X. Liang, Y. Wei, X. Shen, J. Yang, L. Lin, and S. Yan. Proposal-free network for instance-level object segmentation. *arXiv preprint arXiv:1509.02636*, 2015
[38] L. Jin, Z. Chen, and Z. Tu. Object detection free instance segmentation with labeling transformations. *arXiv preprint* arXiv:1611.08991, 2016.
[39] A. Kirillov, E. Levinkov, B. Andres, B. Savchynskyy, and C. Rother. Instancecut: from edges to instances with multicut. In IEEE Conference on Computer Vision and Pattern *Recognition*, pages 7322–7331, 2017.
[40] M. Bai and R. Urtasun. Deep watershed transform for instance segmentation. In IEEE Conference on Computer Vision and Pattern Recognition, pages 2858–2866, 2017.
[41] B. De Brabandere, D. Neven, and L. Van Gool. Semantic instance segmentation with a discriminative loss function.arXiv preprint arXiv:1708.02551, 2017
[42] A. W. Harley, K. G. Derpanis, and I. Kokkinos.Segmentation-aware convolutional networks using local attention masks. In IEEE International Conference on Com*puter Vision*, pages 5048–5057, 2017.
[43] A. Newell, Z. Huang, and J. Deng. Associative embedding: End-to-end learning for joint detection and grouping. In Advances in Neural Information Processing Systems, pages 2277–2287, 2017.
[44] A. Fathi, Z. Wojna, V. Rathod, P. Wang, H. O. Song, S. Guadarrama, and K. P. Murphy. Semantic instance segmentation via deep metric learning. *arXiv preprint* arXiv:1703.10277, 2017.
[45] J. Dai, K. He, and J. Sun. Instance-aware semantic segmentation via multi-task network cascades. In CVPR, pages 3150–3158, 2016.
[46] A. Kirillov, E. Levinkov, B. Andres, B. Savchynskyy, and C. Rother. Instancecut: from edges to instances with multicut. In IEEE Conference on Computer Vision and Pattern *Recognition*, pages 7322–7331, 2017.
[47] R. Girshick. Fast r-CNN. In ICCV, pages 1440–1448, 2015.
[48] T. Lin, P. Doll´ar, R. B. Girshick, K. He, B. Hariharan, andS. J. Belongie. Feature pyramid networks for object detection. In CVPR, pages 936–944, 2017.
[49] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In CVPR, pages 770–778, 2016.
[50] Ulyanov, A. Vedaldi, and V. S. Lempitsky. Instance normalization: The missing ingredient for fast stylization. CoRR, abs/1607.08022, 2016.
[51] S. J. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. IEEE Trans. Pattern Anal. Mach.Intell., 24(4):509–522, 2002.
[52] F. L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. IEEE Trans. Pattern Anal. Mach. Intell., 11(6):567–585, 1989.
[53] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu.Spatial transformer networks. In NIPS, 2015.
[54] Y. Jiang, X. Zhu, X. Wang, S. Yang, W. Li, H. Wang, P. Fu,and Z. Luo. R2cnn: rotational region cnn for orientation robust scene text detection. arXiv preprint arXiv:1706.09579,2017.

Authors Profile

A.S. Venkata Praneel received his B. Tech in Computer Science and Engineering from JNTU, Hyderabad, India, in 2002. He pursued his Master's in Computer Science from California State University, Fresno, CA, the USA, in 2006. He is Pursuing his Doctor of Philosophy (Ph.D.) in the field of Computer Science and Engineering at GITAM University, Visakhapatnam, India. He worked as a Technical Development Lead in many MNCs like Weyerhaeuser, Western Union, Dell Technologies, WPS (Wisconsin Public Services), and Brainware Inc., in the USA.

Currently, he is working as an Assistant Professor in the Department of Computer Science and Engineering at the GITAM School of Technology, GITAM University, Visakhapatnam, Andhra Pradesh, India. His research interests include Machine Learning, Deep Learning, Computer vision, Pattern Recognition, Digital Image Processing, and Data Analytics. In recent years, he is focusing on collaborating actively with researchers in several other disciplines and trying to develop interdisciplinary projects.

Dr. T. Srinivasa Rao received his B.Tech in Computer Science and Systems Engineering from GITAM. He pursued his M.Tech in Computer Science and Technology from the College of Engineering, Andhra University. He was awarded Ph.D. Computer Science and Systems Engineering from Andhra University.

Currently, he is working as an Associate Professor in the Department of Computer Science and Engineering at the GITAM School of Technology, GITAM University, Visakhapatnam, Andhra Pradesh, India. His research interests include Software engineering, Image processing Machine learning with applications to data mining and communications. In his 23 years of teaching experience, he has guided and awarded 6 scholars. In recent years, he is focusing on collaborating actively with researchers in several other disciplines and trying to develop interdisciplinary projects.