iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

RFANET-OCR: AN ATTENTION-BASED END-TO-END FRAMEWORK FOR DETECTION AND RECOGNITION OF TEXT ON METAL SURFACES

Thota Usha,
Raavi Hemalatha,
Yasashwini Velineni,
B. Tech Students, Dept. of Computer Science and Engineering,
R.V.R & J.C College of Engineering, Chowdavaram, Guntur, Andhra Pradesh, India
P. Rama Krishna,
Assistant Professor, Dept. of Computer Science and Engineering,
R.V.R & J.C College of Engineering, Chowdavaram, Guntur, Andhra Pradesh, India

ABSTRACT
Industrial metal part images often suffer from low visual contrast, uneven illumination, surface corrosion, and cluttered backgrounds—posing significant challenges for accurately detecting and recognizing marking characters. These conditions frequently impair existing methods, leading to poor localization of low-contrast text regions and degraded recognition performance. In this work, we build upon the Refined Feature-Attentive Network (RFANet) framework, which combines regression-based and segmentation-based strategies for robust text localization. We retain its core architecture—including parallel feature integration, attention map generation, and proposal refinement—and extend it by integrating an Optical Character Recognition module for end-to-end industrial text spotting. This addition enables both detection and recognition within a unified pipeline. We evaluate our enhanced method on two large-scale industrial scene text datasets—MPSC and SynthMPSC—originally introduced in the RFANet paper, comprising over 102,000 annotated images and 1.9 million text instances across diverse backgrounds and character structures. Extensive experiments on these datasets and four public benchmarks demonstrate that our proposed extension achieves state-of-the-art performance in both text detection and recognition tasks.

Keywords:
Refined Feature-Attentive Network, Low-Contrast Text, Attention Mechanism, Optical Character Recognition, MPSC, SynthMPSC

INTRODUCTION

Text detection involves identifying text regions using bounding boxes, which may encompass curved, multifaceted, and horizontal content in various contexts. With advancements in laser marking technology, many metal components are now inscribed with Arabic numerals and Latin characters to record details such as serial numbers and production dates. This text identification is increasingly vital in industrial manufacturing, enhancing assembly line speed and improving logistics efficiency. Industrial scene text detection presents greater challenges than natural scene detection (e.g., billboards, traffic signs, and retail signage) due to poor visual contrast, corroded detection much more complex, as shown by the differences between natural and industrial scenes in Fig. 1.or uneven surfaces, and cluttered or reflective backgrounds. These unique visual artifacts make industrial text While current deep learning-based methods fall into three main categories—segmentation-based methods, regression-based methods, and hybrid models—traditional scene text detection (STD) techniques typically rely on shape detectors to first extract regions of interest [1], [2], followed by text region classification. Semantic segmentation forms the core of many segmentation-based techniques [3]–[10], which perform pixel-wise classification (text vs. non-text) to group pixels into bounding boxes. However, in industrial images, the edges of the text are often faint and inconsistent due to low contrast and surface corrosion, leading to misclassification and inaccurate localization during post-processing.

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

Figure 1. Visual comparisons between different scene text detection datasets.

Regression-based methods [11]–[20] predict text region geometry through bounding box regression, using either one-stage or two-stage detectors. One-stage detectors are faster but often sacrifice precision, while two-stage detectors, like those using a Region Proposal Network (RPN), generate initial bounding boxes and refine them based on confidence scores [21]. However, even advanced two-stage detectors like RRPN++ [22] struggle with accurate box placement in industrial environments, where background noise and inconsistent textures degrade detection quality. Fig. 2 shows the difference in candidate box centers between RRPN++ and our proposed Refined Feature-Attentive Network (RFN).

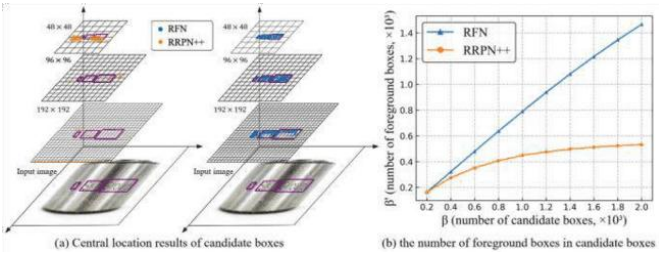


Figure 2. shows the central position findings of candidate boxes on an industrial image in (a)produced by two-stage detectors (RRPN++, RFN(ours)). Using our MPSC dataset, we also get the average number of foreground boxes under the same number of candidate boxes in (b). In particular, our approach has more foreground boxes than RRPN++ while maintaining the same number of candidate boxes. Keep in mind that if the centre point of a candidate box lies within a ground truth, it is considered a foreground box

To address these challenges, we propose a refined feature-attentive network (RFN) that improves localization accuracy using a segmentation-based foreground-focus module (SFF) and an attentive proposal refinement module (APR). The SFF module generates a high-quality attention map by adaptively fusing multi-resolution features, enhancing the detection of multi-scale and low-contrast text. The APR module refines candidate bounding boxes using this attention map, correcting localization errors and improving text region predictions. Our model demonstrates robust performance in detecting horizontal, multi-oriented, and multi-language text across public benchmarks such as MSRA-TD500 [23], USTB-SV1K [24], ICDAR2013 [25], and ICDAR2017-MLT [26], and achieves state-of-the-art results on our proposed MPSC dataset.

We also introduce a comprehensive benchmark dataset for industrial text detection and recognition: the Metal Part Surface Character (MPSC) dataset and its synthetic counterpart, Synth-MPSC. These dataset comprise diverse industrial scenarios featuring low visual contrast, corroded surfaces, varied scales, complex backgrounds, and multi-orientation texts. Together, they support both detection and recognition tasks and provide a valuable resource for further research.

1. We propose an enhanced feature-attentive network (RFN) for industrial text detection that generates high-quality bounding boxes by leveraging a segmentation-based foreground-focus module and an attentive proposal refinement mechanism.
2. We integrate an OCR module into our detection pipeline, enabling end-to-end text spotting that accurately recognizes industrial characters from detected regions, thereby expanding the framework's real-world applicability.

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

3. We construct two large-scale industrial datasets, MPSC and Synth-MPSC, which cover diverse challenges in real and synthetic environments and support both detection and recognition tasks.

RELATED WORK

To enhance practical applicability in industrial settings, we extend our RFN framework by integrating an Optical Character Recognition (OCR) module for end-to-end text spotting. While the detection module localizes character regions, the OCR component recognizes the actual text content—critical for tasks like part identification, serial number verification, and automated inspection. By applying OCR directly on the detected bounding boxes, our system converts visual features into accurate text strings, even in visually challenging industrial environments.

A. Regression-Base Methods

Regression-based methods [11]–[20] predict bounding boxes by estimating key offsets, often using pre-defined anchors for efficient end-to-end training, as in SSD [35]. Textboxes++ [12] employs multi-scale text-box layers, Wang et al. [13] use quadrilateral sliding windows, and Ma et al. [14] integrate angles for arbitrary orientations. While effective, one-stage detectors struggle in cluttered scenes, prompting the use of refinement techniques like RoI pooling [21] and RoIAlign [36]. Iterative refinement modules [16], [17] and anchor-free methods [18] aim to improve accuracy, yet still face challenges in complex backgrounds. To address this, our Refined Feature-Attentive Network (RFN) leverages segmentation-based features and proposal refinement for superior localization.

B. Segmentation-Based Methods

Inspired by Fully Convolutional Networks (FCNs) [3], segmentation-based methods [3]–[10] detect text using instance and semantic segmentation. He et al. [4] used cascaded networks for coarse-to-fine segmentation, while Deng et al. [5] applied pixel-level segmentation with post-processing. Wu et al. [6] added a border class to better separate text regions, and Tian et al. [7] used shape-aware losses for instance separation. Wang et al. [8] expanded kernel sizes for irregular text, and Liao et al. [9] introduced differentiable binarization for end-to-end training. Despite their success, these methods struggle with complex text structures, highlighting the need for hybrid approaches like our RFN, which integrates segmentation and regression with attention-based refinement for improved detection.

C .Combination of Segmentation and Regression Methods

Recent approaches combine segmentation and regression to enhance text detection. He et al. [37] used regional attention for predicting box positions and scores, while Xie et al. [38] reduced false positives using a text-context module with multi-scale features. Huang et al. [39] added a text mask branch to the Pyramid Attention Network to detect curved text. Others, like Yang et al. [41] and Dai et al. [42], used FCIS [43] for predicting text masks, classes, and boxes. Wang et al. [44] introduced Adaptive-RPN with contour-aware features for precise localization. These hybrid models form the basis for advanced frameworks like RFN, which further improves detection through proposal refinement and attention mechanisms.

D. Refined Feature-Attentive Network (RFN)

The Refined Feature-Attentive Network (RFN) advances hybrid regression and segmentation methods by introducing attention mechanisms to improve text localization in challenging industrial environments. It refines CNN-generated proposals by focusing on critical regions and iteratively enhancing bounding box accuracy. Designed for low-contrast, noisy, and complex metal surfaces, RFN effectively addresses the unique demands of industrial scene text detection.

MPSC & SYNTHMPSC DATASET

Most existing text detection datasets are from natural scenes, with few tailored for industrial applications. To address this gap, we introduce the Metal Part Surface Character (MPSC) dataset, designed to capture industrial-specific challenges such as low contrast, poor lighting, corrosion, cluttered backgrounds, and varied text orientations. Additionally, we present SynthMPSC, a synthetic extension created by overlaying characters onto real metal part images to enhance variability and scale for robust model training.

A.MPSC Dataset

To build the MPSC dataset, we collect images that represent diverse types and styles of metal parts and character markings. The final dataset comprises 3,194 images, with 2,555 images allocated for training and 639 for testing.

1) Dataset Construction: Over a three-month period, we performed data deduplication, cleaning, and word-level annotation using quadrilateral boxes aligned clockwise, following the ICDAR 2015 standard [45]. Each image was quality-checked and verified through three inspection rounds.

2) Dataset Analysis: MPSC includes transcriptions of industrial identifiers (e.g., “AlSi9Cu3,” “D151C-050506”) and precise bounding boxes. Most text instances range from 2–7 characters (average 5.5), with 70.6% having an aspect ratio >1. Moreover, 97.8% occupy <20% of image height and <8% of total area, reflecting the compact nature of industrial text.

B. SynthMPSC Dataset

1)Inspiration: While the MPSC dataset captures real-world industrial text variability, limitations like uneven character types, skewed aspect ratios, and restricted orientations hinder broader method generalization. To address this, we create the SynthMPSC dataset

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

using a simplified SynthText approach [49], enabling large-scale synthetic image generation with diverse text properties.
2) Data Synthesis: The SynthMPSC dataset contains 98,962 synthetic images and 1,933,234 text instances. It is built from 1,153 metal part images without text. The generation process involves selecting a background, estimating its depth map, segmenting it into meaningful regions, and randomly selecting zones for text placement. Synthetic texts, extracted from the Newsgroup20 dataset [51], are overlaid using compatible fonts and colors.

C. Comparison With Other Public Dataset

Text localization benchmarks like ICDAR2013 [25], ICDAR2015 [45], MSRA-TD500 [23], and USTB-SV1K [24] are widely used but primarily sourced from natural scenes, such as billboards and shop signs, where text is clear and designed for readability.

1. ICDAR 2013: Focuses on text detection and recognition with nearly horizontal text instances annotated using rectangular bounding boxes.
2. ICDAR 2015: Contains street-level scenes with quadrilateral annotations to capture skew and perspective.
3. MSRA-TD500: Includes long and large text lines, annotated with line-level bounding boxes, and features both Chinese and English text.
4. USTB-SV1K: Consists of low-resolution, blurred images with text in various orientations and perspectives.

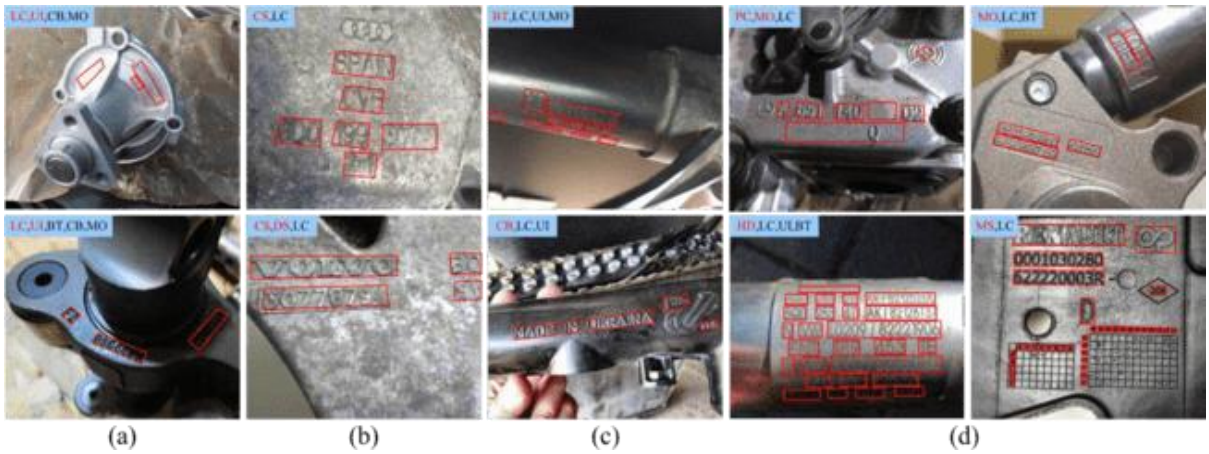


Figure 3. Sample images from the MPSC dataset showcasing various industrial scene challenges: (a) Material characteristics – low visual contrast (LC) and uneven illumination (UI) due to reflective metal surfaces; (b) Environmental effects – corroded (CS) and dirty surfaces (DS) caused by humidity and workshop conditions; (c) Scene noise – blurred text (BT) and complex backgrounds (CB) from unconstrained motion capture; (d) Design complexity – text appears in multiple orientations (MO), scales (MS), high densities (HD), and polymorphic character forms (PC). The blue region in the top-left corner of each image indicates the corresponding challenge. Ground-truth boxes and transcriptions are also provided

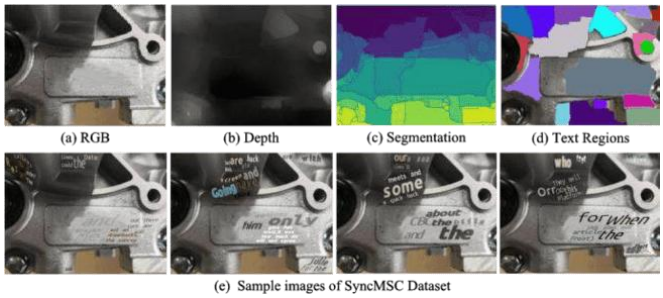


Figure 4. The generation procedure of Synth-MPSC dataset. (Top) Four flowcharts generated by the algorithm. The RGB image is first predicted to generate a depth map and a segmentation map, and then the segments suitable for placing texts in the segmentation map are defined as text regions to synthesize characters. (Bottom) Some synthetic images on the SynthMPSC dataset.

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

Dataset		Image			Label		Direction
	train	Test	all	character	word	line	
ICDAR 2013	229	233	462	✓	-	-	Horizontal
MSRA-TD500	300	200	500	-	✓	✓	Multiple
ICDAR 2015	1000	500	1500	-	✓	-	Multiple
USTB-SVIK	500	500	1000	-	-	✓	Multiple
MPSC (ours)	2555	639	3194	-	✓	-	Multiple
SynthMPSC (ours)	98962	-	98962	✓	✓	-	Multiple

Table I. Statistic comparison between our mpsc and other benchmarks

Table I compares our datasets with public benchmarks. The MPSC dataset offers 3,194 images (2,555 training, 639 test), surpassing ICDAR 2013 (462), MSRA-TD500 (500), ICDAR 2015 (1,500), and USTB-SV1K (1,000). SynthMPSC adds 98,962 synthetic images. MPSC provides 6.91× more samples than all benchmarks combined, addressing both natural and industrial challenges like low contrast, corrosion, and variable lighting—making it the first comprehensive benchmark for industrial text detection and evaluation of models like RFN.

REFINED FEATURE-ATTENTIVE NETWORK WITH INTEGRATED OCR

In this part, we suggest a robust and efficient industrial text detection technique. We begin by providing an overview of the general framework of our suggested approach, followed by an explanation of the specifics of the SFF, APR, and Re-scoring modules, and lastly , the introduction of our method's loss function.

A. Overall Pipeline

Our proposed RFN architecture consists of four main components: 1) a detection branch for classification and regression, 2) a semantic segmentation branch to enhance foreground features, 3) an Attentive Proposal Refinement (APR) module for improved localization, and 4) a ResNet-50 backbone with FPN for multi-scale feature extraction. The segmentation-based foreground-focus module highlights text regions, and the features are fed into classification and regression heads to generate initial boxes. The APR module refines these using attention maps, and a re-scoring strategy ranks the final boxes. The overall architecture is illustrated in Fig. 5.

B. Segmentation-Based Foreground-Focus Module

Metal surfaces exhibit visual complexity due to varying character shapes, lighting, and similar textures. Effective text extraction requires robust multi-scale feature representation. Low-resolution maps help detect large text with strong semantics, while high-resolution maps capture fine details for small text. However, bottom-up integration in existing methods often leads to weak representations on complex metal surfaces. To overcome this, we propose a network that improves multi-scale feature fusion and complementarity for better text detection..

1)Network Design: o enhance multi-scale text perception on complex metal backgrounds, we adaptively fuse multi-resolution features using a specialized extraction module. Features {X1, X2, X3, X4} are extracted via a ResNet-FPN backbone [53] and grouped into low-level (X1) and high-level (X2–X4) features with different resolutions si=(hi,wi),i={1,2,3,4}s_i = (h_i, w_i), i = \{1,2,3,4\}si=(hi,wi),i={1,2,3,4}. This fusion boosts adaptability to complex variations and addresses limitations of individual scale-specific layers.

We enhance textual information in the low-level input to improve semantic features. First, multiple convolutional layers with BatchNorm and ReLU process the input X1∈Rh1×w1×cX1 \in \mathbb{R}^{h1 \times w1 \times c}X1∈Rh1×w1×c. The foreground response accumulates after average pooling along the channel axis. Instead of the sigmoid function, an exponential operation amplifies the difference between foreground and background responses, activating the low-level attention map. Finally,

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

the attention map is element-wise multiplied with the input X1X1X1 to produce the low-level response maps LLL.

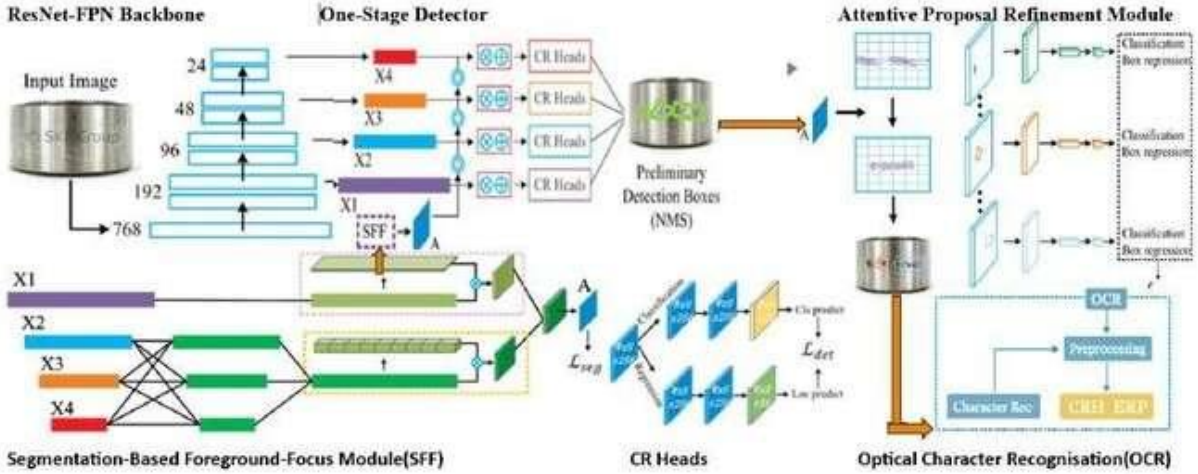


Figure 5. The overall framework of our proposed method. The entire industrial text detection process consists of "ResNet-FPN Backbone", "One-stage Detector", and "Attentive Proposal Refinement Module", shown in the three big red dotted boxes. Firstly, multi-scale features are extracted from the ResNet-FPN backbone and fused to form an attention map by the segmentation-based foreground-focus module. Then, multi-scale features weighted by the attention map are fed into the classification and regression subnets ("CR Hooks") to predict the preliminary detection boxes. After that, the attentive proposal refinement module mines high-quality candidate boxes attached to the foreground to correct location

We create a parallel structure for high-level input that uses mutual information exchange to merge multi- resolution feature maps. In order to retain more multi-scale text properties and enhance the spatial features, each subnet of the high-level input adaptively learns the characteristics of neighbouring subnets. Given Xi ∈ Rhi×wi×c,i ={2,3,4}, the parallel structure can be summed up as follows:

$$Y_k = \sum_{i=2}^{4} \mathcal{F}(X_i, s_k), \qquad (1)$$

where F(Xi,sk) denotes an operation that up- or down-samples Xi from resolution si to sk. In particular, the bilinear sampler is applied after 1 × 1 convolutional layers for the upsampling operation, while 3 × 3 convolutional layers with a stride of 2 are applied for the downsampling process. F(·) denotes 3 x 3 convolutional layers without a sampling layer if si = sk. To create the final high-level response map H, we fuse multi-resolution Yk as follows:

$$H = \varphi(\mathcal{T}(Y_2, s_1) \parallel \mathcal{T}(Y_3, s_1) \parallel \mathcal{T}(Y_4, s_1)), \qquad (2)$$

where ϕ(·) denotes using the channel-wise attention method to assign big responses for foreground features, and T (·) denotes upsampling Yk from resolution sk to s1. "" denotes concatenation along the channel axis.An attention map A is created by fusing multi-level text features (such as Low-level response map L and High- level response map H), which endows higher foreground response values and offers rich and discriminative semantic information. It directs subnetworks at every level to concentrate on textual features. We put the specifics into practice as follows:

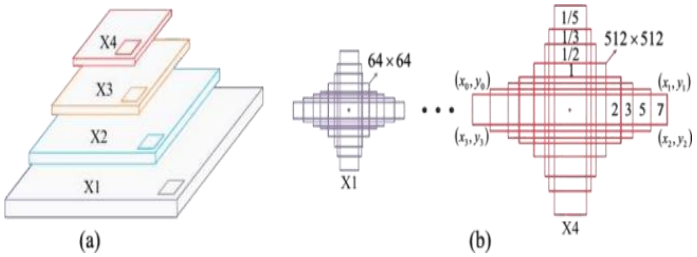
$$\hat{X}_i = X_i \odot (1 + e^{\mathcal{H}(A, s_i)}), \qquad (3)$$

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

where the downsampling of attention map A from resolution s1 to si is denoted by H(·).Each subnet ˆ Xi is then sent into the classification and regression branches, respectively. Four 3x3 convolutional

Figure 6: Our network's default boxes approach. (a) A pyramid network with X1, X2, X3, and X4 components. (b) The standard boxes with various aspect ratios and scales.

layers make up their common structure, which they adopt with different settings. For orientated texts, they add a 3x5



convolutional layer after that. Each subnet ˆ Xi produces a total of hi ×wi ×8 preliminary detection boxes, represented by Bi, and hi ×wi ×8 confidence scores, represented by Si, based on pre-defined anchors from the generation approach of Fig. 6.

To improve the segmentation of the supervised attention map, we propose a unique loss function. Due to the low contrast and imprecise text edges on metal surfaces, distinguishing foreground from background is challenging. We prioritize two objectives for the attention map's loss:a)Maximize detected textual elements.b) Minimize false positives.Using SgtS_{gt}Sgt, the ground truth foreground mask from quadrilateral bounding boxes, we apply the Dice coefficient [54] as an auxiliary loss to handle the imbalance between positive and negative samples..

$$\mathcal{L}_d = 1 - \frac{2 * \sum_{i=1}^{N} (\omega_i \omega_i^*)}{\sum_{i=1}^{N} (\omega_i) + \sum_{i=1}^{N} (\omega_i^*)},$$

where ωi and ω∗i represent the confidence scores of pixel I in Sgt and A, respectively, and N is the number of pixels in the attention map A. The false negative and false positive coefficients can therefore be computed as Da and Db, respectively

$$\begin{aligned} \omega_d &= \omega_i - \omega_i^*, \\ \mathcal{D}_a &= \frac{\sum_{i=1}^{N} \mathbb{1}_{[\omega_d \geqslant \frac{1}{2}]} (1 - \omega_i^*)}{\sum_{i=1}^{N} (\omega_i^*)}, \\ \mathcal{D}_b &= \frac{\sum_{i=1}^{N} \mathbb{1}_{[-\omega_d \geqslant \frac{1}{2}]} \omega_i^*}{\sum_{i=1}^{N} (\omega_i^*)} \end{aligned}$$

.

Finally the loss function are designed as follows:

where the threshold of acceptable false positive classification results is represented by γ, a balance parameter that modifies the

$$\mathcal{L}_g = \begin{cases} \mathcal{D}_a & \text{if } \mathcal{D}_b < \Delta, \\ \mathcal{D}_a + \mathcal{D}_b - \Delta & \text{if } \mathcal{D}_b \geqslant \Delta \end{cases}$$

$$\mathcal{L}_{seg} = \mathcal{L}_d + e^{-1 * \mathcal{L}_d * \gamma} * \mathcal{L}_g,$$

ratio of Lg and Ld in exchange for recognising more text features in the low-contrast and indistinguishable area.

C . Attentive Proposal Refinement Module

Most preliminary detection boxes only partially cover text instances, especially in industrial scenes with oriented rectangles. To improve localization, we propose a box selection technique that uses attention maps to identify high-quality candidate boxes attached to the foreground. These boxes are then prioritized and fed into the refining network for further accuracy improvement..

1)Box Selection Algorithm: Our objective is to choose the best β boxes to apply to the refinement network out of a set of prediction boxes with scores D ={(Bi,Si)|i = 1,...,l}. To create the mask map F, we first binarize the supervised attention map A.In order to eliminate invalid anchor points and retain just those anchors that lie on the anticipated foreground regions, the F will then be scaled into the map Fi at each resolution si. In particular, the set V ={Vi|i = 1,...,l} can be formed by collecting the points in Fi that have a pixel value of 1. Eight candidate boxes with various aspect ratios are represented by each point in V, and the best box is chosen based on the confidence score. As a result, we successfully filter background boxes to provide a collection of candidate boxes with multiple scales, V. Lastly, from ¯ V, the foreground boxes with the highest β confidence values are chosen.

2)Refinement Network: The selected boxes extract regions of interest (ROIs) as feature patches from the first four levels of the

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

ResNet-FPN backbone. Inspired by [14], these patches are flattened and passed through fully connected layers to generate high-dimensional feature vectors. Two additional fully connected layers then predict the classification and regression outputs for each box.

D. Optical Character Recognition (OCR) Module

The OCR module operates as a post-processing step following text detection, where it extracts and interprets textual content from detected regions. Its primary objective is to recognize the characters and words within the bounding boxes predicted by the detection network (such as RFANet). This mechanism is vital for converting images of text into machine-readable strings, which can be used in industrial applications like part identification, inventory management, and quality control. The OCR process includes the following steps:

1)Pre-processing of Text Regions: Before feeding the detected text regions into the OCR system, the following pre-processing steps are applied:

Binarization: Converts the image into black and white for clearer text visibility.Resizing: Scales text regions to a fixed size for standardized input.Noise Removal: Removes noise using techniques like Gaussian blur or median filtering.Contrast Enhancement: Boosts contrast between text and background for better visibility.

2)OCR Engine: The pre-processed text regions are passed to the OCR engine for recognition, which may include:Tesseract OCR: Open-source and customizable for industrial applications.EasyOCR: Deep learning-based library with high accuracy, especially for multilingual text.Google Cloud Vision API: Cloud-based service suitable for large-scale industrial applications.

3)Text Recognition Process: Involves:Feature Extraction: The OCR engine extracts text features like shape and texture.Character Segmentation: Segments the text region into individual characters.Text Recognition: Matches segmented characters to a trained model or dictionary to recognize the text.

4)Post-processing of OCR Results:Spell-checking and Grammar Correction: Fixes errors using dictionaries or language models.Formatting: Converts recognized text into usable data (e.g., part numbers, dates).Confidence Scoring: Provides a confidence score to assess recognition accuracy, with low-confidence results flagged for review.

E. Re-Scoring Mechanism

The non-maximum suppression (NMS) procedure, used in techniques like Faster R-CNN [21] and Mask R-CNN [36], preserves the highest-scoring prediction boxes. While the classification branch predicts confidence scores for each box, this method may miss boxes with better placements but lower scores. To address this, we include an instance score (SI) for each prediction box, as described below.

$$S_I = \frac{\Sigma_{j=1}^{N} \rho_j}{N}, \qquad (10)$$

where ρj stands for the attention map A's pixel value. We use the following numerical formulation to create an overall score S, which has a greater gradient value under the same classification score, as opposed to directly employing the weighted instance score SI and confidence score Sc.

$$S' = e^{S_c}(1 + \mu \frac{e^{S_I}}{e^{1-S_I}}), \qquad (11)$$

where μ is the trade-off coefficient. Finally, S is taken as the new confidence score and fed into the NMS algorithm to get the best prediction boxes.

F. Loss Function

Lseg, Ldet, and Lref make up the RFN's total loss function. First, in order to improve text feature representations, a segmentation loss Lseg optimises the SFF module's output attention map under supervised learning. Second, in order to accomplish preliminary detection, we compute the loss of the classification and regression subnetworks (CR Hooks) in the one-stage detector using the definition that follows:

$$\mathcal{L}_{det} = \frac{1}{M} \sum_{i=1}^{M} (\tau_i \mathcal{L}_{reg}(b_i, b_i') + \mathcal{L}_{cls}(s_i, s_i')), \qquad (12)$$

where M stands for the quantity of default boxes. The 8-vector locations of the i-th default box and prediction box are denoted by bi and bi`, respectively. A binary value called τi indicates whether the i-th default box by IOU corresponds to one of the ground-truth boxes. For Lcls between the label si and the confidence si, we use the focused loss [55]. The smooth L1 loss is used to compute

$$\mathcal{L}_{total} = \lambda_1 \mathcal{L}_{seg} + \lambda_2 \mathcal{L}_{det} + \lambda_3 \mathcal{L}_{ref}, \qquad (13)$$

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

the regression loss Lreg [21]. Thirdly, the classification and location regression losses of the sampled ROIs produced by APR modules—which Faster R- CNN implements—are represented by Lref [18]. Lastly, the following is the definition of the entire loss: where λ1, λ2,andλ3 represent the balance parameter and are set to 1 by default.

EXPERIMENTS

This section evaluates the performance of RFN on the MPSC dataset, comparing it with state-of-the-art methods. The robustness of RFN is further demonstrated on other public scene text datasets. Finally, an ablation study is conducted on the MPSC dataset to analyze the SFF, APR, and Re-scoring modules.

A. Implementation Details

1)Evaluation Metrics: We evaluate the method on the MPSC, MSRA TD500, USTB-SV1K, ICDAR2013, and ICDAR2017-MLT datasets, following standard evaluation techniques from [24]–[26], [45], and [56]. All experiments are conducted on a server with a 32GB NVIDIA Tesla V100 GPU.

2)Parameter Configurations: The method is optimized using SGD with a momentum of 0.9 and weight decay of 1×10^{-4}. The batch size is 12, and the image size is 768x768. The learning rate starts at 0.001 and is halved every 50 epochs. The experiment settings are: $0.01 * Sgt$, $\gamma = 0.1$, $\mu = 0.5$, and $\beta = 1000$.

B. Performance Evaluation on MPSC Dataset

We conduct comparative experiments on the MPSC dataset [8, 14, 18, 19, 22, 36, 40, 44, 58] to evaluate the effectiveness of our approach against advanced text identification techniques. Our method outperforms the best current approach [22] by 1.51% in precision and achieves an F-measure of 86.21%. However, RRPN++ surpasses our method in recall by adding a recognition branch. To improve performance, RFN can also incorporate this branch.

We pre-train the RFN network on the SynthMPSC dataset and refine it on the MPSC dataset. The final result, with an F-measure of 87.05%, shows a 0.84% improvement over training solely on the MPSC dataset, confirming the benefit of synthetic samples. Fig. 7 shows some qualitative results from the MPSC dataset.

The end-to-end text recognition rate depends on the quality of text detection results. We evaluate performance by adjusting the IOU threshold to determine the number of matched boxes. A predicted bounding box is considered matched if its IOU with a ground-truth box exceeds 0.6 or 0.8. Fig. 8 shows that RFN generates the most matched bounding boxes, indicating the effectiveness of the APR module in improving bounding box quality and correcting text localization errors. This demonstrates the efficacy of our approach for improving text recognition, especially in industrial settings where text visibility is low due to factors like corrosion and complex backgrounds. We focus on the text foreground feature to reduce the impact of these challenges.

SFF improves text detection by learning adaptive feature representations to capture more foreground information about metal components. Key aspects of SFF include:1)Fusing scale-sensitive features: Multi-resolution features are divided into low- and high-level categories. The exponential operation enhances foreground responses, while reciprocal information exchange in high-level features boosts spatial properties and preserves multi-scale text features.2)Optimizing foreground features: A loss mechanism with high weights focuses on foreground predictions. It balances false positives by recognizing text in low-contrast areas. The SFF output supports the refinement network, one-stage detector, and the localization process.APR generates high-priority foreground boxes to enhance region-of-interest features. Background boxes have minimal re-correction, as shown in Fig. 9, which compares results from different techniques

Algorithms	Precision (%)	Recall (%)	F-measure (%)
EAST [18]	76.33	73.04	74.65
Mask R-CNN [36]	85.28	79.25	82.15
RRPN [14]	81.98	78.91	80.42
PSENet [8]	85.42	78.40	81.76
PAN [57]	87.07	81.60	84.24
BDN [19]	86.60	77.49	81.79
ContourNet [44]	87.79	81.02	84.27
RRPN++ [22]	86.73	83.90	85.30
FCENet [58]	87.13	81.63	84.29
RFN (ours)	89.30	83.33	86.21
RFN* (ours)	89.82	84.45	87.05

Table II.Comparison results of text detection of metal parts

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

"After detecting text regions and predicting bounding box coordinates with RFANet, we integrated OCR to extract textual content. The OCR component was evaluated on the MPSC dataset based on character accuracy, word accuracy, and overall text recognition precision."

Figure 7. A few instances of multi-oriented detection using RFN on the MPSC dataset. To demonstrate how well our method for multi-oriented text identification works, five different kinds of metal components text are listed in five rows.

Algorithms	Precision (%)	Recall (%)	F-measure (%)
SegLink [59]	86.0	70.0	77.0
EAST [18]	87.3	67.4	76.1
TextSnake+ [60]	83.2	73.9	78.3
PixelLink* [5]	83.0	73.2	77.8
RRPN [14]	82.0	68.0	74.0
RRDt [61]	87.0	73.0	79.0
Lyu et al. [62]	87.6	76.2	81.5
AS-RPN [63]	84.7	80.4	82.5
CRAFT [64]	88.2	78.2	82.9
ATRR [65]	85.2	82.1	83.6
PANI [57]	84.4	83.8	84.1
RFN (ours)	88.4	80.0	84.0
RFN: (ours)	88.4	87.8	88.1

Table III. Comparison resultson the msra-td500dataset

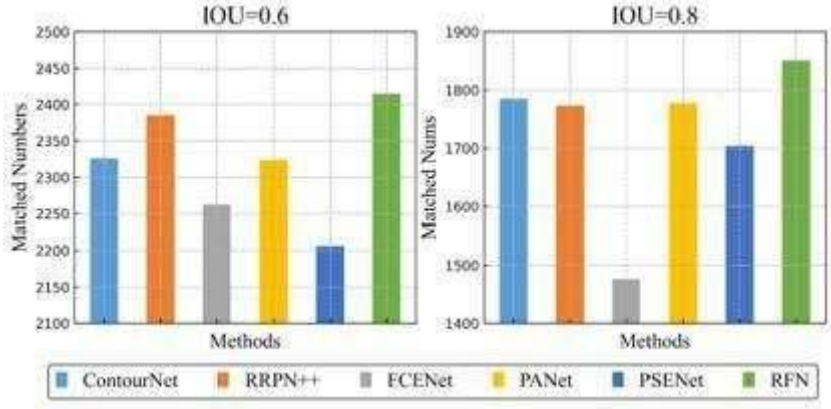


Figure 8. The number of validly matched bounding boxes obtained by different text detection methods in different IOU thresholds. The green bar represents the number of the matched bounding boxes generated by RFN.

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

C. Performance Comparison on Public Datasets

To demonstrate the robustness of our approach, we tested it on common benchmark datasets, using the SynthText dataset for pre-training. The results show that SFF and APR modules achieve comparable performance for scene text identification, while being specifically developed for metal part text detection. We present the top results from comparative techniques discussed in the original research.

1) Detecting Multi-Oriented Text: MSRA-TD500, a challenging multi-oriented text dataset with limited training samples and large text examples, tests our approach for identifying text at various orientations. Our results, shown in Table III, demonstrate that our method outperforms the comparison approaches in precision and F-measure. RFN achieves an F-measure of 84.0%, recall of 80.0%, and precision of 88.4% without additional data training, and 88.1% F-measure when compared to PAN with difficult samples excluded.

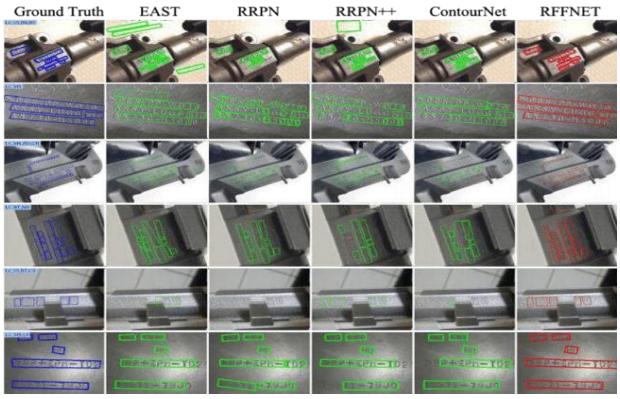


Figure 9. comparisons between the MPSC test set using the RFN and EAST, RRPN, RRPN++, and ContourNet approaches.

2) Detecting Horizontal Text: To evaluate RFN's performance on horizontal texts, we tested it on the ICDAR2013 dataset. As shown in Table IV, our method outperforms other techniques with a precision of 92.5%, recall of 90.7%, and F-measure of 91.6%. Notably, RFN uses single-scale images for both testing and training, without any additional strategies to boost efficiency, proving its reliability for horizontal text detection in natural scenes.

3) Detecting Multi-Language Text: We tested RFN on the ICDAR2017-MLT dataset, which includes short, dense texts in nine languages across various resolutions. Using high resolution and adjusting the candidate boxes and aspect ratios, our approach achieved an F-measure of 73.0%, performing competitively with advanced techniques. This demonstrates RFN's ability to effectively detect text in multiple languages, showcasing its robustness in multilingual environments.

Algorithms	Precision (%)	Recall (%)	F-measure (%)
SegLink* [59]	92.0	84.4	88.1
SSTD [37]	89.0	86.0	88.0
TextBoxes++* [12]	92.0	86.0	89.0
FOTS [67]	-	-	87.3
RRD* [61]	92.0	86.0	89.0
PixelLink* [5]	88.6	87.5	88.1
RRPN [14]	84.0	77.0	80.0
Melinda et al. [68]	93.9	91.5	92.6
FTPN [69]	93.2	91.9	92.5
Liu et al. [70]	90.2	86.3	88.2
Wei et al. [71]	93.7	87.4	90.4
RFN (ours)	92.5	90.7	91.6

Table IV.Comparison results on the icdar 2013 dataset

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

Algorithms	Precision (%)	Recall (%)	F-measure (%)
Sensetime OCR [26]	56.9	69.4	62.6
FOTS [67]	81.0	57.5	67.3
FOTS* [67]	81.9	62.3	67.3
LOMO [16]	78.8	60.6	68.5
LOMO* [16]	80.2	67.2	73.1
PSENet [8]	73.8	68.2	70.9
PSENett [8]	75.4	69.2	72.1
CharNet [72]	77.1	70.1	73.4
CRAFT [64]	80.6	68.2	73.9
Unrealtext [73]	82.2	67.4	74.1
ISNet [74]	78.0	67.4	72.3
RFN (ours)	79.4	67.6	73.0

Table V. Comparison results on the icdar2017-mlt dataset

4)Detecting Low-Resolution Text: We assessed RFN on the challenging USTB-SV1K dataset, containing low-quality and blurry images. Our approach outperformed Wang et al.'s method by 3.2% in F-measure, even without pre-training on SynthText. These results show that RFN excels in diverse and complex scenarios, reliably detecting multi-oriented, horizontal, multi-language, and low-resolution texts

"In this section, we compare the performance of the full pipeline, including both text detection and OCR, with a standalone text detection model. While text detection ensures accurate localization, integrating OCR significantly improves text recognition, particularly in complex industrial environments. This comparison across public datasets demonstrates the benefits of combining detection and recognition".

D. End-to-End Recognition with OCR

To improve text recognition in our RFANet framework, we integrate a CRNN-based OCR module that transcribes detected text regions into actual text. The combined pipeline is evaluated on the SynthMPSC and MPSC datasets, with performance measured by word-level recognition accuracy and average character edit distance. The results show that the OCR module significantly enhances overall recognition accuracy..

OCR Setup:

OCR Model Used: CRNN (Convolutional Recurrent Neural Network)

Training Dataset: SynthMPSC (or any other dataset you used for training the OCR model)

Input: Cropped image regions from RFANet-detected bounding boxes

Output: Recognized text strings corresponding to each bounding box.

E. Ablation Study

We conduct an ablation experiment on the MPSC dataset to assess the impact of each model component, including SFF, APR, Re-Scoring, and OCR. Five model configurations are evaluated individually using precision, recall, and F-measure metrics.

Figure 10. Correctly detected and recognized marking characters on industrial metal parts. These results highlight the ability of our integrated RFANet + OCR model to accurately detect text

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

Algorithms	Precision (%)	Recall (%)	F-measure (%)
Liu et al.t [70]	72.3	50.3	59.3
FTPN [69]	61.4	63.8	62.6
Wang et al.t [75]	73.0	67.0	70.0
RFN (ours)	80.3	67.2	73.2

Table VII. Comparison results on the ustb-sv1k dataset

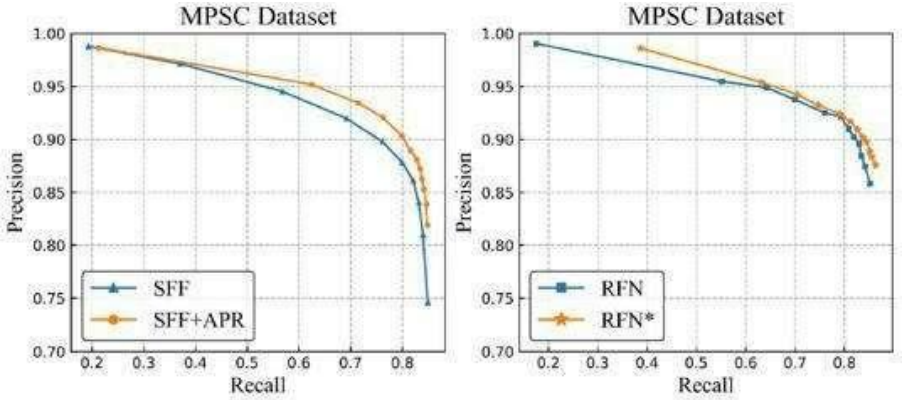


Figure 11. Precision-recall curves. We evaluate the robustness of the twocombinations("SFF"vs "SFF+APR", "RFN" vs "RFN*"), separately.

1)Segmentation-Based Foreground-Focus Module (SFF): The SFF module improves recall by 3.43% and accuracy by 2.78% by focusing on foreground text features using a mask branch. It enhances feature representation, helping the regression network make more accurate bounding box predictions, especially in noisy, complex metal part backgrounds.

SFF	APR	Re-score	Precision (%)	Recall (%)	F-measure (%)	Δ F (%)
✓			82.41	79.22	80.78	-
	✓		85.19	82.65	83.9	0.0312
	✓		85.44	83.09	84.25	0.0347
	✓	✓	89.18	83.19	86.08	0.053
	✓	✓	89.3	83.33	86.21	0.0543

Table VIII. Evaluate the effectiveness of the mpsc dataset in the proposed modules of sff, apr, re-score

2)Attentive Proposal Refinement Module (APR): When combined with SFF, APR increases the F-measure to 86.08%, enhancing detection precision. It refines foreground proposals, reduces false positives, and improves IOU alignment with ground truth, leading to more true positives.

3)Re-Scoring Mechanism: The re-scoring module improves all module combinations by assigning confidence scores, enhancing prediction reliability. It boosts true positives by retaining high-instance score boxes, even with low classification scores. Its impact is most significant when used with SFF alone, as APR already improves box localization. This step promotes better-localized boxes, improving both detection accuracy and text recognition, crucial for industrial settings. The overall performance improvement is shown in the precision-recall curve on the MPSC dataset (Fig. 12).

4)Recognition Head: Control experiments were conducted to validate the RFN framework's recognition head. Removing the recognition head from the RRPN++ baseline caused a 1.36% drop in recall, showcasing RFN's superior localization and identification. Adding the recognition branch to RFN improved recall by 0.52% and accuracy by 2.73% over RRPN++, highlighting the recognition head's importance in enhancing detection and recognition performance, especially for industrial applications.

We conducted experiments to assess the impact of the Optical Character Recognition (OCR) component by disabling it and comparing the performance metrics—precision, recall, and F1-score—between the full model with OCR and the version without it.

The results showed a significant improvement in all metrics when OCR was integrated, highlighting its vital role in accurately interpreting text, especially in industrial scenarios with distorted, rotated, or occluded text. OCR not only enhanced recognition but

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

also improved bounding box confidence, boosting the model's overall robustness. These findings emphasize OCR's importance for practical industrial text detection and tracking applications.

SFF	APR	Re-score	OCR	Precision(%)	Recall(%)	F-measure(%)	ΔF (%)
✓				82.41	79.22	80.78	–
✓	✓			85.19	82.65	83.90	3.12
✓	✓	✓		85.44	83.09	84.05	3.47
✓	✓	✓		89.18	83.19	86.08	5.3
✓	✓	✓	✓	89.3	83.33	86.21	5.43

Table IX. Ablation study on the effectiveness of individual modules in rfn

F. Limitations

Some failures, shown in Figure 12, lead to decreased performance. The first row highlights false negatives in low-resolution industrial images with low-salient text. The second row shows mislocated sentence- and word-level texts due to inconsistent word, sentence, and character spacing across different labels.

Some failures, shown in Figure 12, lead to decreased performance. The first row illustrates false negatives in low-resolution industrial images with low-salient text. The second row shows mislocated sentence- and word-level texts due to inconsistent word, sentence, and character spacing.

Figure 12. Fail examples of text detection results generated by RFN. The green bounding boxes are the labels, the red

bounding boxes are generated by our proposed method, and the yellow boxes are ignored in training stage (the text label is '###').

CONCLUSION

In this study, we present RFANet, an efficient and robust text detection framework for identifying marking characters on metal parts in industrial environments. RFANet addresses localization challenges in complex backgrounds through the integration of the Spatial Feature Fusion (SFF) module, Anchor Point Refinement (APR) module, and a re-scoring mechanism. Extensive experiments on the MPSC dataset show state-of-the-art performance in industrial scene text detection, and competitive results on public benchmark datasets, demonstrating its strong generalization ability for diverse real-world scenarios. We contribute two benchmark datasets, MPSC and SynthMPSC, focused on metal components, advancing research in industrial text detection. Additionally, we extend the framework with a CRNN-based Optical Character Recognition (OCR) model, enabling full text understanding and transcription of industrial scene content. In future work, we aim to deploy this system for automatic part information extraction in manufacturing lines, enhancing automation and traceability in smart factories.

REFERENCES

[1] C. Lu, S. Xia, M. Shao, and Y. Fu, "Arc-support line segments revisited: An efficient high-quality ellipse detection," *IEEE Trans. Image Process.*, vol. 29, pp. 768–781, 2020.

[2] C. Lu, S. Xia, W. Huang, M. Shao, and Y. Fu, "Circle detection by arc-support line segments," in *Proc. ICIP*, 2017, pp. 76–80.

[3] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Trans. Pattern Anal.*

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

Mach. Intell., vol. 39, no. 4, pp. 640–651, Apr. 2017.
[4] T. He, W. Huang, Y. Qiao, and J. Yao, "Accurate text localization in natural image with cascaded convolutional text network," 2016, *arXiv:1603.09423*.
[5] D. Deng, H. Liu, X. Li, and D. Cai, "PixelLink: Detecting scene text via instance segmentation," in *Proc. AAAI*, 2018, pp. 6773–6780.
[6] Y. Wu and P. Natarajan, "Self-organized text detection with minimal post-processing via border learning," in *Proc. ICCV*, 2017, pp. 5010–5019.
[7] Z. Tian *et al.*, "Learning shape-aware embedding for scene text detection," in *Proc. CVPR*, 2019, pp. 4229–4238.
[8] W. Wang *et al.*, "Shape robust text detection with progressive scale expansion network," in *Proc. CVPR*, 2019, pp. 9336–9345.
[9] M. Liao, Z. Wan, C. Yao, K. Chen, and X. Bai, "Real-time scene text detection with differentiable binarization," in *Proc. AAAI*, 2020, pp. 11474–11481.
[10] Y. Cai *et al.*, "Scale-residual learning network for scene text detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 7, pp. 2725–2738, Jul. 2021.
[11] M. Liao, B. Shi, X. Bai, X. Wang, and W. Liu, "TextBoxes: A fast text detector with a single deep neural network," in *Proc. AAAI*, 2017, pp. 4161–4167.
[12] M. Liao, B. Shi, and X. Bai, "TextBoxes++: A single-shot oriented scene text detector," *IEEE Trans. Image Process.*, vol. 27, no. 8,pp. 3676–3690, Aug. 2018.
[13] Y. Liu and L. Jin, "Deep matching prior network: Toward tighter multioriented text detection," in *Proc. CVPR*, 2017, pp. 1962–1969.
[14] J. Ma *et al.*, "Arbitrary-oriented scene text detection via rotation proposals," *IEEE Trans. Multimedia*, vol. 20, no. 11, pp. 3111–3122, Nov. 2018.
[15] S. Zhang, Y. Liu, L. Jin, and C. Luo, "Feature enhancement network: A refined scene text detector," *AAAI*, pp. 2612–2619, 2018.
[16] C. Zhang *et al.*, "Look more than once: An accurate detector for text of arbitrary shapes," in *Proc. CVPR*, 2019, pp. 10552–10561.
[17] X. Yang, J. Yan, Z. Feng, and T. He, "R3Det: Refined single-stage detector with feature refinement for rotating object," 2019, *arXiv:1908.05612*.
[18] X. Zhou *et al.*, "EAST: An efficient and accurate scene text detector," in *Proc. CVPR*, 2017, pp. 5551–5560.
[19] Y. Liu *et al.*, "Exploring the capacity of an orderless box discretization network for multi-orientation scene text detection," 2019, *arXiv:1912.09629*.
[20] P. Cheng, Y. Cai, and W. Wang, "A direct regression scene text detector with position-sensitive segmentation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 11, pp. 4171–4181, Nov. 2020.
[21] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.
[22] J. Ma, "RRPN++: Guidance towards more accurate scene text detection," 2020, *arXiv:2009.13118*.
[23] C. Yao, X. Bai, W. Liu, Y. Ma, and Z. Tu, "Detecting texts of arbitrary orientations in natural images," in *Proc. CVPR*, 2012, pp. 1083–1090.
[24] X. C. Yin, W. Y. Pei, J. Zhang, and H. W. Hao, "Multi-orientation scene text detection with adaptive clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1930–1937, Sep. 2015.
[25] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, and L. Heras, "ICDAR 2013 robust reading competition," in *Proc. ICDAR*, 2013, pp. 1484–1493.
[26] N. Nayef, Y. Fei, I. Bizid, H. Choi, and J. M. Ogier, "ICDAR2019 robust reading challenge on multi-lingual scene text detection and recognition—RRC-MLT-2019," in *Proc. ICDAR*, 2017, pp. 1454–1459. [27] P. Shivakumara, T. Q. Phan, and C. L. Tan, "New Fourier-statistical features in RGB space for video text detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 11, pp. 1520–1532, Nov. 2010.
[28] K. S. Raghunandan, P. Shivakumara, S. Roy, G. H. Kumar, U. Pal, and T. Lu, "Multi-script-oriented text detection and recognition in video/scene/born digital images," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 4, pp. 1145–1162, Apr. 2019.
[29] W. Huang, L. Zhe, J. Yang, and J. Wang, "Text localization in natural images using stroke feature transform and text

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

covariance descriptors," in *Proc. ICCV*, 2013, pp. 1241–1248.
[30] X.-C. Yin, X. Yin, K. Huang, and H.-W. Hao, "Robust text detection in natural scene images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 5, pp. 970–983, May 2014.
[31] K. S. Raghunandan *et al.*, "Riesz fractional based model for enhancing license plate detection and recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 9, pp. 2276–2288, Sep. 2018.
[32] J. J. Lee, P. H. Lee, S. W. Lee, A. Yuille, and C. Koch, "Adaboost for text detection in natural scene," in *Proc. ICDAR*, 2011, pp. 429– 434.
[33] W. Kai, B. Babenko, and S. Belongie, "End-to-end scene text recognition," in *Proc. ICCV*, 2011, pp. 1457–1464.
[34] A. Coates, B. Carpenter, C. Case, S. Satheesh, and B. Suresh, "Text detection and character recognition in scene images with unsupervised feature learning," in *Proc. ICDAR*, 2011, pp. 440–445.
[35] W. Liu *et al.*, "SSD: Single shot multibox detector," in *Proc. ECCV*, 2016, pp. 21–37.
[36] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 386–397, Feb. 2020.
[37] P. He, W. Huang, T. He, Q. Zhu, Y. Qiao, and X. Li, "Single shot text detector with regional attention," in *Proc. ICCV*, 2017, pp. 3047–3055.
[38] E. Xie, Y. Zang, S. Shao, G. Yu, C. Yao, and G. Li, "Scene text detection with supervised pyramid context network," in *Proc. AAAI*, 2019, pp. 9038–9045.
[39] Z. Huang, Z. Zhong, L. Sun, and Q. Huo, "Mask R-CNN with pyramid attention network for scene text detection," in *Proc. WACV*, 2019, pp. 764–772.
[40] H. Li, P. Xiong, J. An, and L. Wang, "Pyramid attention network for semantic segmentation," 2018, *arXiv:1805.10180*.
[41] Q. Yang *et al.*, "InсepText: A new inception-text module with deformable PSROI pooling for multi-oriented scene text detection,"
2018, *arXiv:1805.01167*.
[42] Y. Dai *et al.*, "Fused text segmentation networks for multi-oriented scene text detection," in *Proc. ICPR*, 2018, pp. 3604–3609.
[43] Y. Li, H. Qi, J. Dai, X. Ji, and Y. Wei, "Fully convolutional instanceaware semantic segmentation," in *Proc. CVPR*, 2017, pp. 2359– 2367.
[44] Y. Wang, H. Xie, Z. Zha, M. Xing, Z. Fu, and Y. Zhang, "ContourNet: Taking a further step toward accurate arbitrary-shaped scene
text detection," in *Proc. CVPR*, 2020, pp. 11750–11759.
[45] D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, S. Ghosh, and E. Valveny, "ICDAR 2015 competition on robust reading," in *Proc. ICDAR*, 2015, pp. 1156–1160.
[46] C. Lu, C. Gu, K. Wu, S. Xia, H. Wang, and X. Guan, "Deep transfer neural network using hybrid representations of domain discrepancy," *Neurocomputing*, vol. 409, pp. 60–73, Mar. 2020.
[47] C. Lu, H. Wang, C. Gu, K. Wu, and X. Guan, "Viewpoint estimation for workpieces with deep transfer learning from cold to hot," in
Neural Information Processing. Krong Siem Reap, Cambodia: Springer, 2018, pp. 21–32.
[48] X. Wu, C. Lu, C. Gu, K. Wu, and S. Zhu, "Domain adaptation for viewpoint estimation with image generation," in *Proc. ICCAIS*, 2021, pp. 341–346.
[49] A. Gupta, A. Vedaldi, and A. Zisserman, "Synthetic data for text localisation in natural images," in *Proc. CVPR*, 2016, pp. 2315–2324.
[50] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898–916, Aug. 2011.
[51] K. Lang, "Newsweeder: Learning to filter netnews," in *Proc. ICML*, 1995, pp. 331–339.
[52] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, 2016, pp. 770–778.
[53] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. CVPR*, 2017, pp. 2117–2125.
[54] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-Net: Fully convolutional neural networks for, volumetric medical image segmentation,"
in *Proc. 3DV*, 2016, pp. 565–571.
[55] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, Feb. 2020.
[56] K. Dasgupta, S. Das, and U. Bhattacharya, "Stratified multi-task learning for robust spotting of scene texts," in *Proc. ICPR*, 2021, pp. 3130–3137.
[57] W. Wang *et al.*, "Efficient and accurate arbitrary-shaped text detection with pixel aggregation network," in *Proc. ICCV*, 2019, pp. 8439–8448.

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

[58] Y. Zhu, J. Chen, L. Liang, Z. Kuang, L. Jin, and W. Zhang, "Fourier contour embedding for arbitrary-shaped text detection," in *Proc. CVPR*, 2021, pp. 3123–3131.
[59] B. Shi, X. Bai, and S. Belongie, "Detecting oriented text in natural images by linking segments," in *Proc. CVPR*, 2017, pp. 2550– 2558.
[60] S. Long, J. Ruan, W. Zhang, X. He, W. Wu, and C. Yao, "TextSnake: A flexible representation for detecting text of arbitrary shapes,"
in *Proc. ECCV*, 2018, pp. 19–35.
[61] M. Liao, Z. Zhu, B. Shi, G. Xia, and X. Bai, "Rotation-sensitive regression for oriented scene text detection," in *Proc. CVPR*, 2018,
pp. 5909–5918.
[62] P. Lyu, C. Yao, W. Wu, S. Yan, and X. Bai, "Multi-oriented scene text detection via corner localization and region segmentation," in
Proc. CVPR, 2018, pp. 7553–7563.
[63] A. Zhu, H. Du, and S. Xiong, "Scene text detection with selected anchors," in *Proc. ICPR*, 2021, pp. 6608–6615.
[64] Y. Baek, B. Lee, D. Han, S. Yun, and H. Lee, "Character region awareness for text detection," in *Proc. CVPR*, 2019, pp. 9365–9374.
[65] X. Wang, Y. Jiang, Z. Luo, C.-L. Liu, H. Choi, and S. Kim, "Arbitrary shape scene text detection with adaptive text region representation," in *Proc. CVPR*, 2019, pp. 6449–6458.
[66] C. Yao, X. Bai, and W. Liu, "A unified framework for multioriented text detection and recognition," *IEEE Trans. Image Process.*, vol. 23, no. 11, pp. 4737–4749, Nov. 2014.
[67] X. Liu, D. Liang, S. Yan, D. Chen, Y. Qiao, and J. Yan, "FOTS: Fast oriented text spotting with a unified network," in *Proc. CVPR*, 2018, pp. 5676–5685.
[68] L. Melinda and C. Bhagvati, "Parameter-free table detection method," in *Proc. ICDAR*, 2019, pp. 454–460.
[69] F. Liu, C. Chen, D. Gu, and J. Zheng, "FTPN: Scene text detection with feature pyramid based text proposal network," *IEEE Access*, vol. 7, pp. 44219–44228, 2019.
[70] Z. Liu, W. Zhou, and H. Li, "Scene text detection with fully convolutional neural networks," *Multimedia Tools Appl.*, vol. 78, no. 13,
pp. 18205–18227, Jul. 2019.
[71] G. Wei, W. Rong, Y. Liang, X. Xiao, and X. Liu, "Toward arbitraryshaped text spotting based on end-to-end," *IEEE Access*, vol. 8,
pp. 159906–159914, 2020.
[72] L. Xing, Z. Tian, W. Huang, and M. R. Scott, "Convolutional character networks," in *Proc. ICCV*, 2019, pp. 9126–9136.
[73] S. Long and C. Yao, "UnrealText: Synthesizing realistic scene text images from the unreal world," 2020, *arXiv:2003.10608*.
[74] P. Yang *et al.*, "Instance segmentation network with self-distillation for scene text detection," *IEEE Access*, vol. 8, pp. 45825–45836, 2020.
[75] X. Wang, X. Feng, and Z. Xia, "Scene video text tracking based on hybrid deep text detection and layout constraint,"
Neurocomputing, vol. 363, pp. 223–235, Mar. 2019.