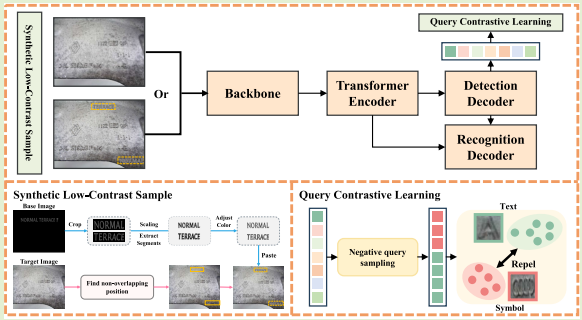
ContraText-DETR: Boosting Industrial Scene Text Detection Based on Contrastive Learning and Synthetic Low-Contrast Text

Yunseo Jeong, Seokjun Kwon, Jeongmin Shin, *Graduate Student Member, IEEE*, and Yukyung Choi

***Abstract*—The scene text detection in industrial environments is challenging due to low contrast, corrosion, and glare on metallic surfaces, which affect the detection accuracy. Furthermore, symbols such as engraved logos resemble text, leading to misclassification and increased false positive rates. In this study, we propose a framework that applies synthetic data augmentation to handle low-contrast conditions and employs contrastive learning to enhance text–symbol differentiation. This framework improves robustness and generalization by training the model on diverse low-contrast scenarios. Furthermore, it reduces misclassification by using false positives as negative samples, enhancing the separation between text and symbols. Since the proposed method is adopted only during the training phase, it avoids computational overhead during inference. Moreover, it does not rely on modifications to the architecture, allowing flexible integration into existing text detectors. In addition, we observed ambiguities in the annotation criteria of the MPSC dataset, particularly in distinguishing individual text instances and labeling their ground-truth bounding boxes. To resolve this, we reannotate the dataset by defining a consistent standard for text instance labeling. Our experimental results demonstrate that the proposed model achieves state-of-the-art performance on the MPSC dataset, validating its applicability to industrial text detection tasks.**



***Index Terms*—Contrastive learning, industrial manufacturing, industrial scene text, synthetic augmentation, text detection.**

I. Introduction

DEEP learning-based vision systems are widely used in intelligent manufacturing to enable automated inspection with faster processing, improved reliability, and lower cost. Leveraging data from vision sensors, these systems perform key tasks, such as anomaly detection [1], [2], defect detection [3], [4], and text detection [5]. In particular, the scene text detection in the industrial manufacturing domain enhances productivity and efficiency by identifying text on components to enable quality control and automation. While traditional text detectors [6], [7], [8] have predominantly focused on natural scene text, addressing challenges, such as arbitrary-shaped or oriented text [9], [10], [11], industrial text detection involves additional challenges.

Specifically, industrial text detectors face issues such as corrosion and reflective glare that often appear on metallic components. Moreover, the laser-etching process commonly used for marking text on metal surfaces often results in low-contrast text, which hinders precise detection. As shown in Fig. 1, industrial scenes often suffer from low contrast, uneven illumination, and surface corrosion, whereas natural scenes typically feature clear contrast, even illumination, and clean surfaces. To further highlight the contrast differences, we compared commonly used natural scene datasets, such as Total-Text [12], TextSeg [13], and CTW1500 [14], with the industrial dataset MPSC [15]. Contrast levels below 30 were

Received 30 May 2025; accepted 19 June 2025. Date of publication 1 July 2025; date of current version 15 August 2025. This work was supported in part by the Institute for Information & Communications Technology Planning & Evaluation (IITP)-Information Technology Research Center (ITRC) under Grant IITP-2025-RS-2024-00437494 (25%); in part by the ICT Challenge and Advanced Network of HRD (ICAN) Program funded by Korean Government (MSIT) under Grant IITP-2025-RS-2022-00156345 (25%); and in part by the Technology Innovation Program (Development of an AI-Based High Resolution Low Power Smart Camera and Machine Vision Integrated Solution for Defect Detection in Manufacturing) funded by the Ministry of Trade, Industry & Energy (MOTIE), South Korea, under Grant 20023583 (50%). The associate editor coordinating the review of this article and approving it for publication was Dr. Chiman Kwan. *(Corresponding author: Yukyung Choi.)*

Yunseo Jeong, Seokjun Kwon, and Jeongmin Shin are with Sejong University, Seoul 05006, Republic of Korea (e-mail: ysjeong@rcv.sejong.ac.kr; sjkwon@rcv.sejong.ac.kr; jmshin@rcv.sejong.ac.kr).

Yukyung Choi is with Sejong University, Seoul 05006, Republic of Korea, and also with the Artificial Intelligence and Robotics Institute (AIRI), Seoul 05006, Republic of Korea (e-mail: ykchoi@rcv.sejong.ac.kr).

Digital Object Identifier 10.1109/JSEN.2025.3582931

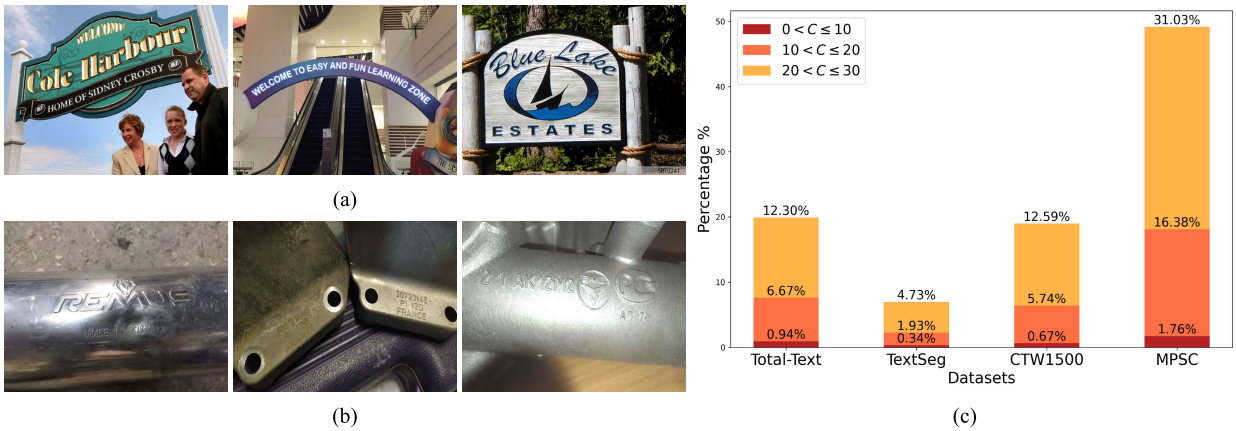


Fig. 1. Comparison of scene text detection datasets. (a) Example of a text image from a natural scene and (b) text image from an industrial scene, which presents challenges, such as uneven lighting, cluttered metallic surfaces, and low contrast. (c) Differences between natural scene datasets (e.g., Total-Text, TextSeg, and CTW1500) and industrial scene datasets (e.g., MPSC) with respect to low-contrast samples. The contrast levels (C) are categorized as 0–10, 10–20, and 20–30, corresponding to calculated contrast values within text bounding boxes.

categorized as low contrast and divided into three uniform ranges. As shown in Fig. 1(c), low-contrast regions account for nearly 50% of the MPSC dataset, whereas natural scene datasets rarely exceed 20%.

Recent research [15], [16] has employed attention mechanisms and boundary refinement techniques to enhance the feature representation for effective industrial text detection. While they introduce some improvements, their strategies primarily focus on enhancing boundary clarity and feature refinement rather than directly addressing the visibility of text in severely low-contrast conditions. As a result, the persistent challenge of low-contrast text has yet to be adequately addressed.

Another challenge in industrial text detection is the misclassification of engraved logos or certification marks as text, as these symbols often exhibit shapes closely resembling textual characters. This misclassification arises from the inherent limitations of traditional supervised learning, which relies primarily on appearance-based features. These approaches often lack the training signal required to effectively differentiate between text and symbols. Consequently, existing industrial text detection models [15], [16], [17] struggle with this limitation, leading to a high rate of false positives.

In this article, we present a comprehensive approach to enhance the robustness of industrial text detection by addressing the challenges. Specifically, we propose a synthetic data augmentation designed to address the low-contrast problem. This method enhances the model's robustness by introducing low-contrast variations during training, improving its ability to handle challenging scenarios. In addition, to address misclassification between text and symbols, we propose a learning framework that incorporates contrastive learning into the detection pipeline. This framework encourages the model to better distinguish between text and symbols by treating the false positives as negative samples and maximizing the distance between the query and false positive features. Moreover, bounding box annotations in the MPSC dataset were often labeled based on inconsistent criteria, particularly in cases involving serial codes, where the division of text instances is ambiguous due to a lack of semantic meaning (see Fig. 2). These inconsistencies hinder model training and evaluation. To address this, we reannotate and provide a revised version of the dataset with more consistent annotations. The proposed approach demonstrates its effectiveness by achieving state-of-the-art performance on the MPSC dataset, validating its utility in RGB sensor-based industrial text detection scenarios. The main contributions of this article are summarized as follows.

1) We propose a synthesis-based augmentation strategy for industrial scene text detection, which is a simple yet effective method to enhance the model robustness against low-contrast conditions.
2) We introduce a contrastive learning-based framework to mitigate false positives caused by symbols resembling text. This approach enhances the discriminative ability, enabling more precise differentiation between text and nontext symbols.
3) We refine the annotation criteria of the MPSC dataset and relabel ground-truth boxes accordingly. The new annotations improve consistency and provide more reliable supervision for training and evaluation.

II. Related Works

A. Text Detection in Natural Scene

Deep learning-based text detection and text spotting methods have significantly improved natural scenes. Existing approaches can be broadly categorized into CNN-based and Transformer-based methods. Early CNN-based methods, such as [6] and [7], primarily focused on detecting regular-shaped text but faced limitations when handling curved or arbitrarily shaped text. To address these challenges, methods such as [10], [11], [18], [19], [20], [21], and [22] were introduced. TextSnake [18] improved the detection of curved text by representing text as a sequence of circles. ABCNet [10], in particular, significantly enhanced performance by employing Bezier curves to detect curved text effectively. With the advent of Transformers [23], which have shown exceptional

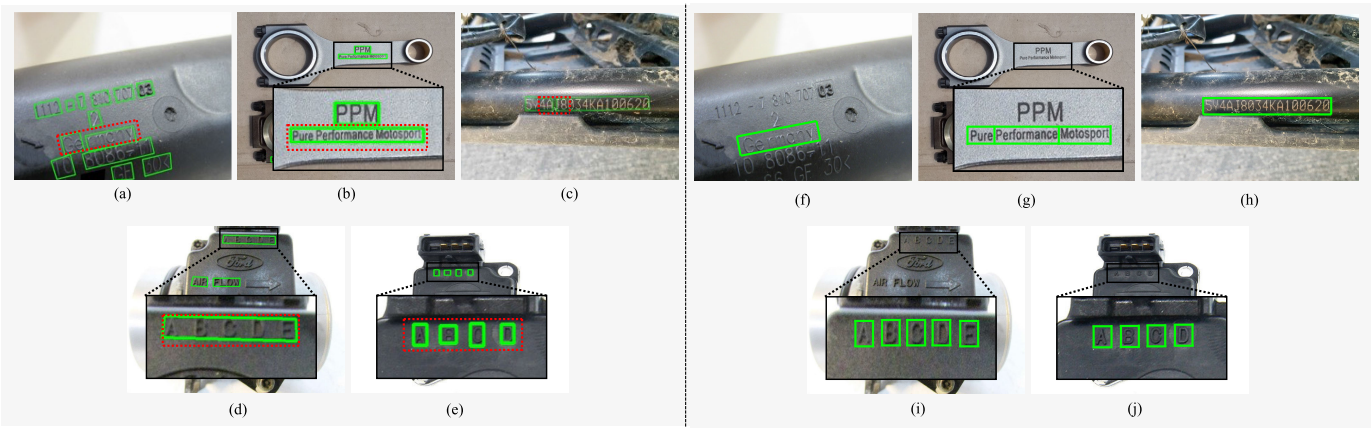


Fig. 2. Examples of annotation errors in the original MPSC dataset and their corresponding revised annotations. The green bounding boxes are the ground-truth bounding boxes, and the red dashed boxes indicate inconsistent regions. The original annotations are displayed on the left side, while the revised versions are shown on the right side. (a) Single-text instance incorrectly annotated as two separate bounding boxes. (b) Three distinct text instances into a single-bounding box. (c) Case where two instances are arbitrarily separated despite the absence of spacing. (d) and (e) Inconsistent annotation styles for text with similar characteristics, observed across different images. The revised annotations ensure more consistent labeling and improve model reliability. (f)–(j) Illustrate revised annotations based on word semantics and character spacing for consistent labeling.

results in vision tasks such as [24] and [25], these architectures began to be applied to text detection and spotting. TESTR [26], one of the early Transformer-based studies, introduced a dual-decoder architecture for text detection and recognition, enabling joint optimization of the two tasks. Later, DPText-DETR [27] leveraged control points with a point query formulation to effectively detect arbitrarily shaped and curved text. ESTextSpotter [28] introduced Bezier control points as positional priors and applied denoising training to address the limitations of the conventional bipartite matching algorithm. DeepSolo [29] proposed a method that utilizes the encoder–decoder structure of Transformers to generate curve centerlines, effectively creating positional queries that separate text location and content information. The aforementioned detectors and spotters enabled efficient and accurate detection, mainly in natural scenes, by leveraging high-contrast and visually prominent text features. However, previous methods have insufficient consideration for industrial environments where low contrast and uneven illumination are common. In contrast, our synthetic augmentation method can improve the model's performance in low-contrast cases and can be applied to various existing methods, regardless of their network architectural design.

B. Text Detection in Industrial Scene

Vision sensor-based systems are widely adopted in industrial environments for various inspection tasks [30], [31], [32]. Among these tasks, the text detection in industrial environments poses unique challenges distinct from natural scene text detection, such as low-visual contrast and corroded surfaces on components. To address these challenges, RFN [15] introduced the first industrial text detection dataset, MPSC, and proposed the refined feature-attentive network to improve localization accuracy. Building on this, BARD [16] tackled issues such as uneven lighting, cluttered backgrounds, and low contrast by introducing a boundary feature fusion mechanism (BFFM) and a refined boundary discrimination module (RBDM). MEAST [17], based on the EAST [6] framework, enhances vehicle identification number (VIN) localization in challenging industrial scenarios by leveraging spatial consistency between original and mirrored images. In addition, Gao et al. [33] addressed the detection of curved text on metal surfaces by combining text rectification with character clustering techniques. Recent work [34] enhances meter reading in substations and other industrial environments using YOLOv5 [35], CRAFT [36], and CRNN [37]. While these methods achieve good results, they overlook false positive cases, such as symbols that resemble text. On the other hand, our simple but effective contrastive learning strategy encourages the model to distinguish features between symbols and text, enhancing the detection accuracy in industrial environments.

III. Methods

A. Annotation Refinement for Industrial Text Detection

Industrial text detection datasets are limited due to security and confidentiality concerns, with the MPSC dataset [15] being one of the few publicly available for benchmarking. However, unlike natural scene datasets, such as total text, which support semantic word-level annotations, industrial datasets are primarily composed of serial numbers, making word-level instance definition inherently ambiguous. This ambiguity is reflected in the MPSC dataset, which exhibits inconsistent annotation standards, as illustrated in Fig. 2. In (a), a meaningful word instance such as "Germany" is split across two bounding boxes, while in (b), multiple word instances are grouped into a single-bounding box. In (c)–(e), serial numbers are inconsistently annotated, with bounding boxes arbitrarily split or merged regardless of spacing, demonstrating a lack of uniform annotation standards. These inconsistencies not only slow down model convergence during training but also reduce

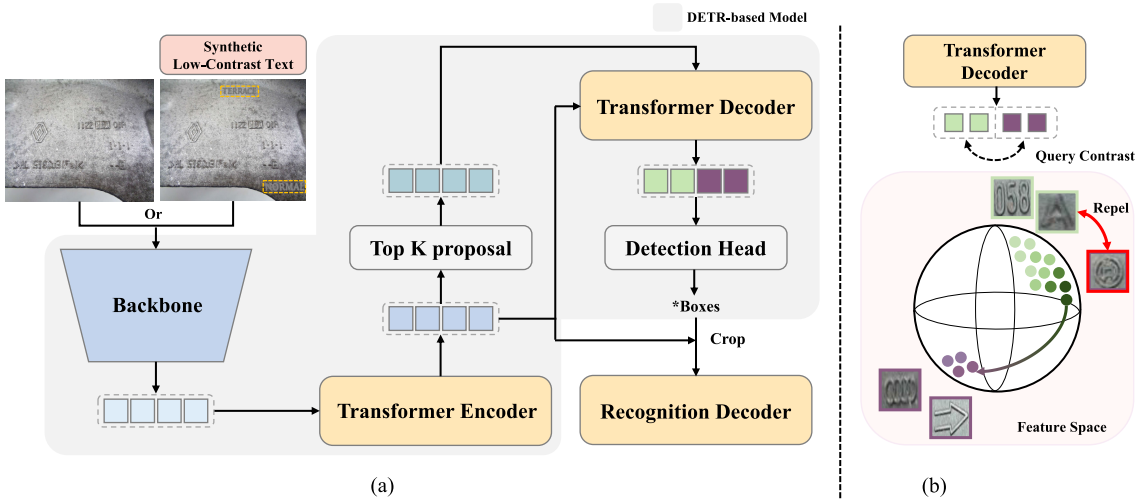


Fig. 3. Overall architecture of our proposed framework. Contrastive learning is integrated to enhance text–symbol differentiation, with embeddings from the Transformer decoder used to derive positive and negative pairs. Green boxes indicate positive samples, while purple boxes represent negative samples in the query contrast. (a) Overview. (b) Query contrast.

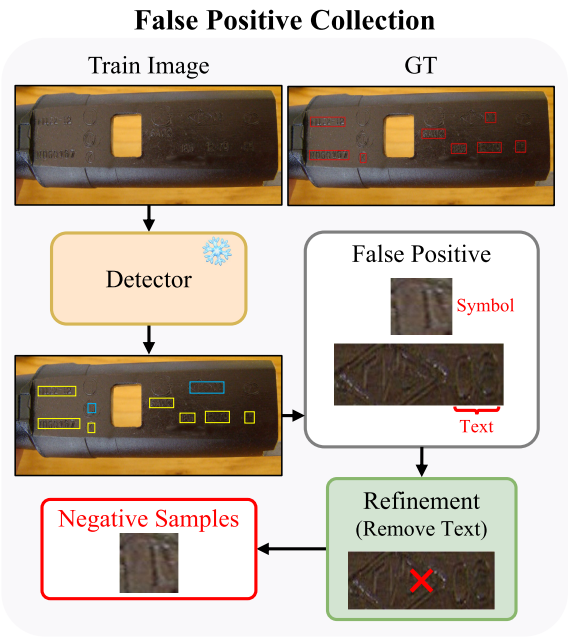


Fig. 4. Illustration of the false positive collection process. False positives (blue) are refined by removing actual text instances, and the remaining samples are used as negative examples for contrastive learning.

clarity during inference. Consequently, they lead to evaluation errors by misaligning the model's predictions with the ground truth.

Therefore, we reannotated the MPSC dataset by introducing revised annotation standards to eliminate ambiguities in the original dataset. First, instances with semantic content are annotated at the word level, following the conventions of natural scene text detection datasets [12], [14]. For example, in Fig. 2(f) and (g), the words "Germany" and "Pure" are treated and annotated as one instance. Second, for nonsemantic sequences such as serial codes, where it is difficult to divide the instances using the word-level semantic criterion, we established a criterion based on the spacing between characters. Fig. 2(h), for instance, illustrates that a single-bounding box is labeled when the spacing between characters is narrow and uniform. In contrast, as shown in Fig. 2(i) and (j), we assigned bounding box labels to each character when the spacing between characters in a single line was deemed wide enough to annotate the box for each character. Finally, the unrecognizable text is labeled as "###" according to the original MPSC standard. To minimize subjectivity across the dataset, all reannotations were conducted by a single annotator in accordance with the defined guidelines. Although a certain level of subjective judgment from the annotator is involved in revising the labels for serial code cases, we believe that our revised annotation standards ensure consistency of text labels, improving the model convergence and reducing the evaluation error.

B. Overview

The overall architecture is depicted in Fig. 3. Our model is based on DPText-DETR [27], which consists of a CNN backbone network, a deformable Transformer encoder, and a decoder from deformable DETR [38]. We extend this framework with the addition of a recognition decoder. Specifically, the input image is processed by the backbone network to extract multiscale features, which are subsequently passed to the deformable DETR encoder. The final layer of the encoder outputs a fixed number of box proposals. Each proposal, defined by its center point and scale, is uniformly sampled along its upper and lower edges to serve as initial control points. In the decoder, the control point coordinates are integrated with learnable content queries, generating composite queries that are fed into the decoder. The decoder outputs are fed into the detection head, which generates N control point coordinates and a class confidence score for each text instance. Next, utilizing the target box coordinates, a fixed-size recognition feature is extracted from the encoder features through region of interest (RoI) pooling. This feature is subsequently passed to the recognition head, which predicts the text instance.

C. Enhancing Text and Symbol Differentiation

1) Text–Symbol Contrasting: The industrial text detection faces challenges due to the presence of symbols. These symbols share visual similarities with text, making them difficult to differentiate. This similarity can lead to false positives and reduce detection accuracy. To address this, we introduce contrastive learning to enhance the model's ability to distinguish text from symbols. Specifically, false positives, such as symbols, are explicitly treated as negative samples, while genuine text regions serve as positive samples. However, false positive cases cannot be directly assigned as negative samples since there is no supervision of symbols and other backgrounds on the MPSC dataset.

Therefore, the base model is first trained using the bounding boxes of text regions only, and false positive predictions are collected. The collected samples are then manually inspected to remove those that include actual text, and the remaining ones are treated as negative labels, as illustrated in Fig. 4. This refinement is necessary because the inherent characteristics of industrial datasets, as described in Section III-A, lead to inevitable false positives even when text is detected.

After extracting bounding box information for the refined false positives, the model is retrained using a contrastive loss to explicitly improve the distinction between text and symbol embeddings. Specifically, predictions are matched with ground truth using a predetermined matching cost. Query embeddings from matched predictions are used as positive pairs for contrastive loss. For the remaining predictions, potential false positives are identified based on confidence scores exceeding a predetermined threshold. These high-confidence predictions are further processed to calculate the intersection over union (IoU) with preselected false positives, and those exceeding a fixed IoU threshold are identified as negative samples. The selected positive and negative samples are then used to compute the contrastive loss based on InfoNCE loss [39]. We note that the embeddings are derived from the decoder's output feature maps, as shown in Fig. 3(b), ensuring that the embeddings of text and false positives are distinctly separated in the feature space. For a given query embedding $\mathbf{z}_i$, positive samples are denoted by $\mathbf{z}_i^+$, and negative samples are denoted by $\mathbf{z}_j^-$. The loss is then computed as follows:

$$\mathcal{L}_{\text{cl}} = -\log \frac{\exp\left(\mathbf{z}_i \cdot \mathbf{z}_i^+ / \tau\right)}{\exp\left(\mathbf{z}_i \cdot \mathbf{z}_i^+ / \tau\right) + \sum_{\mathbf{z}_j^-} \exp\left(\mathbf{z}_i \cdot \mathbf{z}_j^- / \tau\right)} \tag{1}$$

where τ is a temperature parameter that is set to 0.5. This approach can be applied to any natural scene text detector without requiring any modifications to its architecture. Moreover, our contrastive learning-based approach does not increase the computational cost of existing models during inference.

2) Multitask Learning With Integrated Recognition: Existing text detection models [6], [40], [41] rely solely on visual cues for text detection, while humans utilize both visual and prior linguistic knowledge. Inspired by this observation, we propose that text detection requires both identifying text and understanding its content. To this end, we extend the text detection pipeline by incorporating a recognition head, enabling the model to achieve robust discrimination between the text and background.

In this way, the recognition branch takes the 2-D encoder features extracted from the detection branch as input instead of a cropped version of the original image. The recognition head consists of six layers of a Transformer decoder, where cross-modality interaction between visual features and text is applied in the second multihead attention module. This module leverages the current character features to extract subsequent character features from the 2-D feature map.

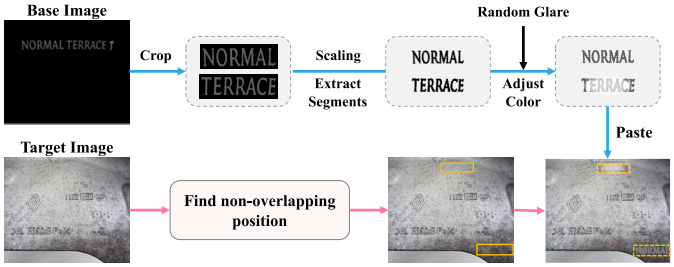


Fig. 5. Pipeline for generating synthetic low-contrast samples. This involves segmentation-based cropping, masking, and adjusting text region colors to blend with the background (see Section III-D).

D. Synthetic Augmentation for Low Contrast

Text detection datasets in industrial settings encounter low-contrast issues due to uneven lighting or deteriorated metallic surfaces. Moreover, text on products is frequently engraved or printed in colors that closely resemble the background. To address low-contrast issues, we propose a synthetic data augmentation that enhances model generalization by generating synthetic low-contrast samples to be used in training.

We utilized the TextSeg dataset [13], which provides pixel-level segmentation and bounding box labels for diverse text styles, such as scene text (e.g., road signs and billboards) and design text (e.g., artistic text on posters). Fig. 5 illustrates the process of generating synthetic samples. First, the text region is cropped on the segmentation map using bounding box labels. The average bounding box scale factor α is then computed from the MPSC dataset, and the text segment is resized using α. This scaling process aligns the scale distribution of the bounding boxes between the MPSC and TextSeg datasets, stabilizing the model's training. Next, the text instances are segmented using segmentation labels and are filled with a color that closely resembles the background region where the text will be pasted. Subsequently, the target annotations are updated to reflect the synthetic samples. The position of these synthetic samples is determined by considering the locations of existing text in the target image. Specifically, a random point within the target image is selected as the top-left corner of the paste region, and its area is calculated using the scaled width and height of the text segment. The candidate area is confirmed for placement if it does not overlap with any existing target bounding boxes. We note that if no nonoverlapping area is identified after multiple attempts, the synthetic augmentation is deemed unsuitable and is not applied to the target image.

In addition, to simulate lighting reflections often observed on metallic or glossy surfaces, we introduce a random glare effect to the text segment by overlaying semitransparent ellipses. This step may reduce local text–background contrast and help the synthetic data better reflect real-world industrial conditions. By exposing the model to diverse low-contrast scenarios that closely mimic real-world industrial conditions, these synthetic samples significantly enhance its robustness in text detection in challenging environments.

E. Optimization

1) Bipartite Matching: The proposed model generates a fixed number of predictions for each image, which surpasses the number of ground-truth examples in most cases. Therefore, matching between predictions and ground-truth instances is necessary for effective loss computation. In accordance with DETR [24], we utilize the Hungarian algorithm for bipartite matching, where σ represents a permutation that minimizes the matching cost C, as follows:

$$\arg\min_{\sigma} \sum_{g=1}^{G} C\left(Y^{(g)}, \widehat{Y}^{(\sigma(g))}\right). \tag{2}$$

Here, G represents the number of ground-truth instances within an image. The cost function $C(Y^{(g)}, \widehat{Y}^{(\sigma(g))})$ measures the matching cost between a ground-truth instance $Y^{(g)}$ and a prediction $\widehat{Y}^{(\sigma(g))}$ and is defined as follows:

$$\begin{aligned} C\left(Y^{(g)}, \widehat{Y}^{(\sigma(g))}\right) &= \lambda_{\text{cls}} \text{FL}'(\hat{b}^{(\sigma(g))}) \\ &\quad + \lambda_{\text{coord}} \sum_{n=1}^{N} \left\| p_n^{(g)} - \widehat{p}_n^{(\sigma(g))} \right\|_1 \end{aligned} \tag{3}$$

where λ_{cls} and λ_{coord} represent weights for each loss and $\hat{b}^{(\sigma(g))}$ denotes the predicted probability of the text class. FL$'$ is defined following [26] as a weighted combination of positive and negative terms: $\text{FL}'(p) = -\alpha(1-p)^{\gamma}\log(p) + (1-\alpha)p^{\gamma}\log(1-p)$, derived from the focal loss [42]. α and γ are parameters that control the balance between positive and negative samples and adjust the focus on hard classified samples. In this work, α and γ are set to 0.25 and 2.0, respectively. The second term is computed as the $L1$ distance between the ground truth p_n and the predicted control point coordinates $\widehat{p}_n$, where N represents the number of control points.

2) Control Point Loss: $L1$ distance loss is employed for control point coordinates regression as follows:

$$\mathcal{L}_{\text{coord}}^{(j)} = \mathbb{1}_{\{j \in \text{Im}(\sigma)\}} \sum_{n=1}^{N} \left\| p_n^{(\sigma^{-1}(j))} - \widehat{p}_n^{(j)} \right\|_1. \tag{4}$$

Here, j represents the index of a query and $\mathbb{1}_{\{j\in\text{Im}(\sigma)\}}$ is an indicator function that is active only for queries matched to ground-truth instances through the permutation σ. $p_n^{(\sigma^{-1}(j))}$ and $\widehat{p}_n^{(j)}$ denote the ground-truth and predicted control points, respectively.

3) Text Classification Loss: For the classification loss of text instances, we utilize focal loss [42], defined as follows:

$$\begin{aligned} \mathcal{L}_{\text{cls}}^{(j)} &= -\mathbb{1}_{\{j\in\text{Im}(\sigma)\}} \alpha (1-\hat{b}^{(j)})^{\gamma} \log(\hat{b}^{(j)}) \\ &\quad - \mathbb{1}_{\{j\notin\text{Im}(\sigma)\}} (1-\alpha)(\hat{b}^{(j)})^{\gamma} \log(1-\hat{b}^{(j)}) \end{aligned} \tag{5}$$

where $\hat{b}^{(j)}$ represents the confidence score for the jth query. The indicator functions $\mathbb{1}_{\{j\in\text{Im}(\sigma)\}}$ and $\mathbb{1}_{\{j\notin\text{Im}(\sigma)\}}$ determine whether a query is matched or unmatched.

4) Recognition Loss: For recognition loss, each character is treated as a distinct class, and the cross-entropy loss is used as follows:

$$\mathcal{L}_{\text{rec}}^{(j)} = \mathbb{1}_{\{j\in\text{Im}(\sigma)\}} \sum_{i=1}^{M} \left(-c_i^{(\sigma^{-1}(j))} \log \hat{c}_i^{(j)} \right) \tag{6}$$

where M is the total number of character classes. $c_i^{(j)}$ is the one-hot vector of the ground-truth character and $\hat{c}_i^{(j)}$ is the confidence score for the ith class.

5) Overall Loss: The decoder loss combines the previously mentioned losses with a contrastive loss L_{cl}, which is defined as follows:

$$\mathcal{L}_{\text{dec}} = \sum_{j} \left(\lambda_{\text{cls}} \mathcal{L}_{\text{cls}}^{(j)} + \lambda_{\text{coord}} \mathcal{L}_{\text{coord}}^{(j)} + \lambda_{\text{rec}} \mathcal{L}_{\text{rec}}^{(j)} \right) + \lambda_{\text{cl}} \mathcal{L}_{\text{cl}}. \tag{7}$$

In addition, to provide intermediate supervision for the encoder, the following loss is employed:

$$\mathcal{L}_{\text{enc}} = \sum_{i} \left(\lambda_{\text{cls}} \mathcal{L}_{\text{cls}}^{(i)} + \lambda_{\text{coord}} \mathcal{L}_{\text{coord}}^{(i)} + \lambda_{\text{GIoU}} \mathcal{L}_{\text{GIoU}}^{(i)} \right) \tag{8}$$

where $\mathcal{L}_{\text{GIoU}}$ is the generalized IoU loss [43] defined as follows:

$$\mathcal{L}_{\text{GIoU}} = 1 - \text{IoU} - \frac{|C| - |A \cup B|}{|C|}. \tag{9}$$

Here, A and B represent the predicted and ground-truth bounding boxes, respectively, and C represents their smallest enclosing box that contains both A and B. The term $|A \cup B|$ is the area of the union of A and B, and $|C|$ is the area of the enclosing box.

Finally, the overall loss for the proposed model is defined as the sum of the encoder and decoder losses

$$\mathcal{L} = \mathcal{L}_{\text{enc}} + \mathcal{L}_{\text{dec}}. \tag{10}$$

IV. Experiments

A. Dataset

MPSC: The MPSC dataset [15] is a text detection dataset designed for industrial environments. It contains 3194 images, including 2555 training images and 639 testing images. Unlike natural scene datasets rich in curved or arbitrarily shaped text, the MPSC dataset focuses on industrial challenges, including low-visual contrast, uneven illumination, corroded and dirty surfaces, and high-text density. This dataset addresses various challenges influenced by four principal factors: metal properties, industrial conditions, artificial design, and scene noise. Specifically, it encompasses issues such as low visual contrast, uneven illumination, corroded and dirty surfaces, and high-text density. As introduced in Section III-A, the MPSC

Fig. 6. Qualitative comparisons of text detection under challenging conditions, including uneven lighting and low contrast, on the MPSC dataset.

dataset is reannotated with our revised standards to address its annotation inconsistencies. We note that all experiments in this study, including baselines and ablation studies, were conducted using the revised version to ensure consistency and fairness in evaluation.

B. Evaluation Metrics

Precision, recall, and F-measure are used as the main evaluation metrics for text detection. Precision measures the ratio of true positives to the total predicted positives, evaluating the accuracy of the predictions. Recall calculates the proportion of accurately predicted positive samples to the total number of positive samples, evaluating the model's ability to identify text instances. The F-measure, a harmonic mean of precision and recall, evaluates the balance between these two metrics. The equations are as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{11}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{12}$$

$$F - \text{measure} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{13}$$

where TP, FP, and FN denote the true positive, false positive, and false negative, respectively.

C. Implementation Details

We trained our model with a batch size of 4 using an RTX 3090 GPU and the AdamW optimizer. Following the settings in [27], ResNet50 was used as the backbone, and the encoder and decoder were configured with six layers. In addition, the number of queries was fixed at 100. The model was fine-tuned on the MPSC dataset for 30k iterations, starting with a base learning rate of 5×10^{-5}, which was further reduced by a factor of 10 at the 16k iteration. The loss weights were set as follows: $\lambda_{\text{cls}} = 2.0$, $\lambda_{\text{coord}} = 5.0$, $\lambda_{\text{GIoU}} = 2.0$, $\lambda_{\text{rec}} = 0.5$, and $\lambda_{\text{cl}} = 4.0$.

D. Comparison With Existing Methods

We evaluated the effectiveness of our framework by applying it to TESTR, DeepSolo, RT-DETR, and DPText-DETR.

TABLE I
SCENE TEXT DETECTION RESULTS ON THE MPSC DATASET. ALL MODELS WERE REIMPLEMENTED BASED ON PUBLICLY AVAILABLE CODE AND TRAINED ON OUR REVISED MPSC DATASET

Method	Precision	Recall	F-measure
Mask R-CNN [44]	80.1	81.2	80.7
TextSnake [18]	86.5	77.4	81.7
PAN [20]	81.9	72.4	76.9
PSENet [19]	74.1	70.4	72.2
DBNet [40]	83.0	78.9	80.9
FCENet [45]	77.0	72.3	74.6
DBNet++ [41]	85.7	82.2	83.9
TESTR [26]	88.7	82.8	85.6
ESTextSpotter [28]	80.9	79.9	80.4
DeepSolo [29]	88.4	79.8	83.8
RT-DETR [46]	84.5	87.3	85.9
DPText-DETR [27]	88.1	87.9	88.0
TESTR + Ours	90.1	83.0	86.7
DeepSolo + Ours	90.5	79.9	84.9
RT-DETR + Ours	86.9	**88.5**	87.7
DPText-DETR + Ours	**92.8**	88.1	**90.4**

All models were trained on the revised MPSC dataset. As shown in Table I, our method consistently improved the performance of all models. In particular, DPText-DETR with our framework resulted in a notable improvement, increasing precision by 4.9% and F-measure by 2.4%. This result demonstrates the effectiveness of our proposed techniques in reducing false positives and achieving robust performance in industrial text detection scenarios. Fig. 6 presents the qualitative comparisons with DeepSolo [29] and TESTR [26] under challenging conditions, such as uneven lighting and low contrast. While both DeepSolo and TESTR struggle in these scenarios, our method effectively detects text in such conditions, demonstrating its applicability to industrial settings. In this section, we present the experimental results when comparing the proposed framework with existing methods to assess its effectiveness.

E. Ablation Studies

1) Ablation Study on Revised Annotation: As shown in Fig. 7, we compared the performance of models trained with original and revised annotations, respectively. Note that each model used the same annotation for both training and eval-

TABLE II

IMPACT OF THE PROPOSED APPROACH ON IMPROVING TEXT AND SYMBOL DISTINCTION. "REC DECODER" REFERS TO THE MULTITASK LEARNING PERFORMED BY INCORPORATING THE RECOGNITION DECODER, WHILE "TEXT–SYMBOL CL" DENOTES TEXT–SYMBOL CONTRASTIVE LEARNING. "FP" INDICATES FALSE POSITIVE

Rec Decoder	Text-Symbol CL	P	R	F	#FP	GFLOPs
-	-	88.1	87.9	88.0	356	184.5
✓	-	89.3	**88.4**	88.8	315	184.5
✓	✓	**93.4**	87.7	**90.5**	**190**	184.5

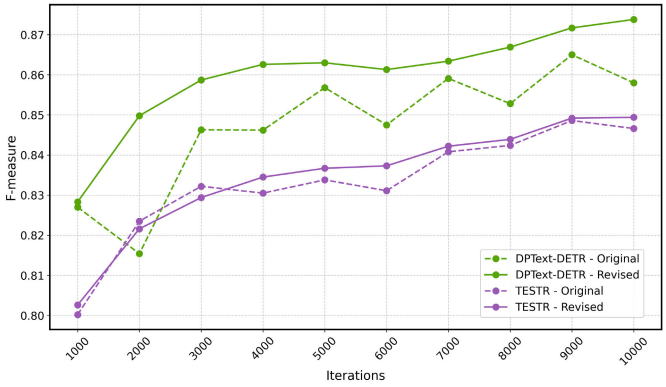
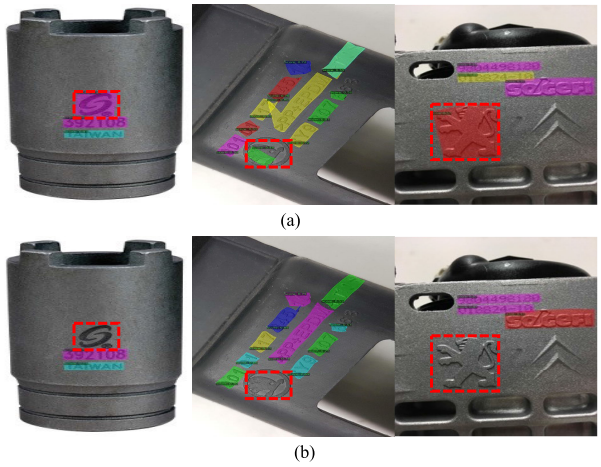


Fig. 7. *F*-measure curves over training iterations for DPText-DETR and TESTR trained with original and revised annotations. The revised annotations lead to smoother convergence.



(a) (b)

Fig. 8. Visualization of the proposed method for improving text and symbol differentiation. (a) DPText-DETR. (b) Proposed method.

uation to assess the impact of annotation consistency. Both models trained with the original annotations show unstable convergence during the initial training phase. This instability is caused by inconsistent labeling in the training data, leading to noisy learning signals, as well as mismatched annotation standards between training and evaluation. In contrast, the revised annotations lead to more stable convergence, likely due to clearer and more consistent instance definitions, which reduce the noisy supervision and improve alignment between predictions and ground-truth annotations.

2) Ablation Study on Improving Text and Symbol Differentiation: Table II presents the experimental results verifying the

TABLE III

QUANTITATIVE ANALYSIS OF THE PROPOSED SYNTHETIC AUGMENTATION. THE RESULTS ARE EVALUATED ON THE MPSC DATASET BY DIVIDING IT INTO TWO SETS BASED ON CONTRAST LEVELS (C): $0 < C \leq 10$ AND $10 < C \leq 20$. HERE, C REPRESENTS THE CONTRAST VALUE CALCULATED WITHIN THE BOUNDING BOX OF EACH SAMPLE, WHICH IS USED TO CATEGORIZE THEM INTO THE RESPECTIVE RANGES

Method	Synth Aug	$0 < C \leq 10$			$10 < C \leq 20$		
		P	R	F	P	R	F
TESTR [26]	✗	83.4	74.2	78.6	90.1	77.0	83.0
	✓	**85.1**	**75.8**	**80.2**	**91.3**	**77.7**	**83.9**
DPText-DETR + Ours	✗	85.9	78.4	82.0	89.5	82.8	86.0
	✓	**86.4**	**80.4**	**83.3**	**90.6**	**83.8**	**87.0**

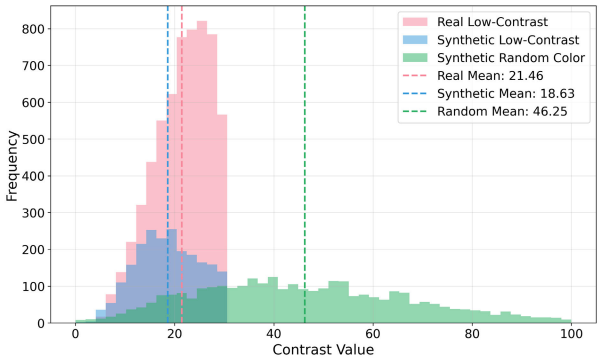


Fig. 9. Distribution of contrast values for real and synthetic text samples. The synthetic samples generated by the proposed method follow a contrast distribution that resembles real data.

effectiveness of the proposed learning strategy, which incorporates text recognition and text–symbol contrastive learning to improve the distinction between text and false positives (e.g., symbols). First, multitask learning with text detection and recognition improved the F-measure by 0.8%. This indicates that training text recognition encourages the model to understand the content of the text, enhancing its text detection performance. Furthermore, employing contrastive learning between symbols and text led to a further 1.7% improvement, achieving a final F-measure of 90.5%. This improvement is further evidenced by a notable reduction in false positives, from 356 to 190. This substantial decrease demonstrates the effectiveness of the method in distinguishing symbols from text, resulting in a 4% increase in precision. In addition, the proposed method maintains computational efficiency, as the GFLOPs remain constant at 184.5. This learning-based approach improves accuracy without introducing additional computational costs or compromising inference speed. Fig. 8 presents a qualitative comparison of our model and the baseline. The baseline model designed for natural scenes frequently misidentifies symbols as text, revealing its limitations in industrial settings. In contrast, our method successfully detects only the texts, demonstrating that our learning strategy is effective in reducing false positive predictions.

3) Ablation Study of the Synthetic Augmentation: Table III presents the experimental results to assess the effectiveness of the proposed synthetic augmentation method in low-contrast

TABLE IV
HYPERPARAMETER STUDY ON (a) IOU THRESHOLD REQUIRED FOR DETERMINING NEGATIVE SAMPLES AND (b) WEIGHT OF λ_{cl} IN CONTRASTIVE LOSS

Threshold	Precision	Recall	F-measure
0.1	92.0	87.4	89.7
0.2	92.5	87.7	90.0
0.3	92.8	**87.8**	90.3
0.4	**93.4**	87.7	**90.5**

(a) IoU threshold

λ	Precision	Recall	F-measure
1	92.5	87.4	89.9
2	92.3	**88.2**	90.2
3	92.8	87.5	90.1
4	**93.4**	87.7	**90.5**

(b) Weight of $\mathcal{L}_{cl}$

cases. To evaluate the generalization performance of the proposed augmentation, it is applied to our baseline architecture [27], which we extend with a recognition decoder and text–symbol contrastive learning. The same augmentation is also applied to TESTR [26] without modifying its original architecture.

In this experiment, we define low-contrast samples as those with contrast values below 30, based on our observation that detection performance noticeably degrades within this range. To better examine the effectiveness of the proposed augmentation under particularly difficult conditions, we focused our analysis on two low-contrast intervals: 0–10 and 10–20. The results indicate that the model has difficulty with low-contrast samples, as shown by a 3–5% reduction in F-measure for samples with contrast values below 10, compared with the 10–20 range. When the proposed augmentation method was applied to TESTR, the F-measure improved by 1.6% for samples in the 0–10 range and by 0.9% for those in the 10–20 range. Similarly, our synthetic data augmentation also led the baseline model to achieve performance gains of 1.3% and 1.0% in the same ranges, respectively. These results imply that the proposed augmentation method can effectively utilize natural scene datasets to address low-contrast issues in industrial text detection.

Furthermore, we analyzed the distribution of contrast values to examine the similarity between our synthetic low-contrast samples and real ones. We also included a variant of our synthetic method that uses random color assignment instead of background-aware adjustment. As shown in Fig. 9, we compared the contrast statistics across three groups: real low-contrast samples (red), synthetic samples generated using random color assignment (green), and synthetic samples generated using our proposed low-contrast adjustment (blue). Specifically, we computed contrast values for text regions cropped from real and synthetic samples. Although the sample counts differ, our analysis targets the contrast value range, not the frequency. The contrast range of our synthetic samples is similar to that of real low-contrast samples, with mean values of 18.63 and 21.46, respectively. In contrast, random color assignment resulted in an average contrast of 46.25, which is considerably higher than the average in real samples. This indicates that our synthetic samples better reflect the contrast properties of real data and improve detection robustness under industrial scenarios.

4) Ablation Study of the Hyperparameters: Table IV shows the experiments performed on the hyperparameters used in training the proposed model. To calculate the contrastive loss, negative samples (symbols) must be selected based on the IoU between the predefined symbol annotations and the predicted boxes that do not match the ground truth. Table IV(a) presents the validation experiments conducted by varying the IoU threshold. Lowering the threshold allows the model to include more queries located near symbols as negative samples; however, increasing the threshold limits the negative samples to queries that strongly misclassify symbols as text. The results indicate that performance improves as the threshold increases, demonstrating that contrastive learning is most effective when it focuses on embeddings from clear misclassifications as negative samples. In addition, Table IV(b) presents experimental results that analyze the weight of the contrastive loss. The optimal detection performance of 90.5% F-measure was achieved with a loss weight of 4.

TABLE V
COMPARISON OF INFERENCE SPEED ON THE RTX 3090 AND JETSON ORIN NANO. ALL MODELS WERE TRAINED AND EVALUATED WITH AN INPUT RESOLUTION OF 640 × 640. INFERENCE ON JETSON WAS CONDUCTED WITH TENSORRT AT FP16 PRECISION

Model	F-measure (%)	#Params (M)	FPS (RTX 3090)	FPS (Jet)
DPText-DETR + Ours	87.6	45.8	28.1	–
RT-DETR + Ours	86.5	20.1	35.3	19.1

5) Ablation Study on Deployment Efficiency on Edge Devices: Real-time inference is essential for text detection in industrial scenarios, such as manufacturing and automated inspection. To enable real-time deployment, our method is designed to enhance detection performance without modifying the architecture or parameters of the base detector during inference. Accordingly, it can be applied directly to existing real-time models without affecting their inference speed. To examine this, our method was applied to two detectors: RT-DETR, a recent lightweight detector designed for real-time performance, and DPText-DETR as our main baseline. We then evaluated inference speed on two hardware platforms using an NVIDIA RTX 3090 GPU and a Jetson Orin Nano. On the Jetson, we converted RT-DETR using TensorRT to enable efficient inference. Note that we cannot report the inference time of DPText-DETR since it fails to convert to TensorRT due to being out of memory on the Jetson Orin Nano.

As shown in Table V, RT-DETR with our method achieves an 86.5 F-measure and 35.3 frames/s on the RTX 3090, and reaches 19.1 frames/s on the Jetson Orin Nano. This

indicates that our method maintains real-time performance on edge devices. DText-DETR with our method records a 1.1% higher F-measure but operates 7.2 frames/s slower. These results demonstrate a tradeoff between detection accuracy and inference speed, depending on the base detector. While DPText-DETR combined with our framework is more suitable for high-capacity computing environments, RT-DETR integrated with our framework, remains more practical for real-time applications on edge devices. Since our method does not modify the base architecture during inference, it can be flexibly integrated into various detection frameworks based on deployment needs.

V. Conclusion

In this article, we addressed key challenges in industrial text detection, such as low contrast and symbol misclassification. We introduced a synthetic augmentation designed to enhance detection performance in low-contrast conditions. We also revised the MPSC dataset annotations to address irregularities in the original labels, improving the detector's training stability and increasing confidence in evaluation. Our experiments demonstrated that the proposed augmentation strategy handled low-contrast samples, and the learning strategy significantly reduced false positives. Consequently, our final model achieved state-of-the-art performance in industrial text detection when evaluated on the MPSC dataset.

Our contrastive learning approach currently depends on preprocessing to manage false positives, particularly through manual inspection to identify symbol-like distractors. While this process ensures high-quality negative samples, it may limit scalability in large-scale applications. Future work will explore semi-supervised or self-supervised learning to automate negative sample collection and facilitate end-to-end training with minimal human intervention. We hope our approach inspires further research into improving text detection under challenging industrial settings.

References

[1] H. Nizam, S. Zafar, Z. Lv, F. Wang, and X. Hu, "Real-time deep anomaly detection framework for multivariate time-series data in industrial IoT," *IEEE Sensors J.*, vol. 22, no. 23, pp. 22836–22849, Dec. 2022.

[2] H. Shao, J. Peng, M. Shao, and B. Liu, "Multiscale prototype fusion network for industrial product surface anomaly detection and localization," *IEEE Sensors J.*, vol. 24, no. 20, pp. 32707–32716, Oct. 2024.

[3] P. Peng, K. Fan, X. Fan, H. Zhou, and Z. Guo, "Real-time defect detection scheme based on deep learning for laser welding system," *IEEE Sensors J.*, vol. 23, no. 15, pp. 17301–17309, Aug. 2023.

[4] H. Liu, C. Chen, R. Hu, J. Bin, H. Dong, and Z. Liu, "CGTD-net: Channel-wise global transformer-based dual-branch network for industrial strip steel surface defect detection," *IEEE Sensors J.*, vol. 24, no. 4, pp. 4863–4873, Feb. 2024.

[5] Q. Tang, Y. Lee, and H. Jung, "The industrial application of artificial intelligence-based optical character recognition in modern manufacturing innovations," *Sustainability*, vol. 16, no. 5, p. 2161, Mar. 2024.

[6] X. Zhou et al., "EAST: An efficient and accurate scene text detector," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5551–5560.

[7] B. Shi, X. Bai, and S. Belongie, "Detecting oriented text in natural images by linking segments," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2550–2558.

[8] 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.

[9] M. Liao, G. Pang, J. Huang, T. Hassner, and X. Bai, "Mask textspotter v3: Segmentation proposal network for robust scene text spotting," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Cham, Switzerland: Springer, 2020, pp. 706–722.

[10] Y. Liu, H. Chen, C. Shen, T. He, L. Jin, and L. Wang, "ABCNet: Real-time scene text spotting with adaptive bezier-curve network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9809–9818.

[11] Y. Liu et al., "ABCNet v2: Adaptive bezier-curve network for real-time end-to-end text spotting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 11, pp. 8048–8064, Nov. 2022.

[12] C. K. Ch'ng and C. S. Chan, "Total-text: A comprehensive dataset for scene text detection and recognition," in *Proc. 14th IAPR Int. Conf. Document Anal. Recognit. (ICDAR)*, vol. 1, Nov. 2017, pp. 935–942.

[13] X. Xu, Z. Zhang, Z. Wang, B. Price, Z. Wang, and H. Shi, "Rethinking text segmentation: A novel dataset and a text-specific refinement approach," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 12045–12055.

[14] L. Yuliang, J. Lianwen, Z. Shuaitao, and Z. Sheng, "Detecting curve text in the wild: New dataset and new solution," 2017, *arXiv:1712.02170*.

[15] T. Guan, "Industrial scene text detection with refined feature-attentive network," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 9, pp. 6073–6085, Sep. 2022.

[16] Y. Yang, M. Hu, J. Yu, and B. Jing, "Real-time industrial text detection with boundary awareness and refined differentiation," *Available at SSRN 4959851*. 2024.

[17] G. Yin, S. Huang, T. He, J. Xie, and D. Yang, "Mirrored EAST: An efficient detector for automatic vehicle identification number detection in the wild," *IEEE Trans. Ind. Informat.*, vol. 20, no. 3, pp. 4594–4605, Mar. 2024.

[18] 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. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 20–36.

[19] W. Wang et al., "Shape robust text detection with progressive scale expansion network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9336–9345.

[20] W. Wang et al., "Efficient and accurate arbitrary-shaped text detection with pixel aggregation network," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8440–8449.

[21] G. Wei, W. Rong, Y. Liang, X. Xiao, and X. Liu, "Toward arbitrary-shaped text spotting based on end-to-end," *IEEE Access*, vol. 8, pp. 159906–159914, 2020.

[22] X. Han, J. Gao, C. Yang, Y. Yuan, and Q. Wang, "Focus entirety and perceive environment for arbitrary-shaped text detection," *IEEE Trans. Multimedia*, vol. 27, pp. 287–299, 2025.

[23] A. Vaswani et al., "Attention is all you need," in *Proc. NeurIPS*, 2017, pp. 1–12.

[24] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Cham, Switzerland: Springer, 2020, pp. 213–229.

[25] A. Dosovitskiy, "An image is worth 16×16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2020, pp. 1–22.

[26] X. Zhang, Y. Su, S. Tripathi, and Z. Tu, "Text spotting transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 9519–9528.

[27] M. Ye, J. Zhang, S. Zhao, J. Liu, B. Du, and D. Tao, "DPText-DETR: Towards better scene text detection with dynamic points in transformer," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, vol. 37, 2023, pp. 3241–3249.

[28] M. Huang et al., "ESTextSpotter: Towards better scene text spotting with explicit synergy in transformer," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 19495–19505.

[29] M. Ye et al., "DeepSolo: Let transformer decoder with explicit points solo for text spotting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 19348–19357.

[30] L. Liu, X. Feng, F. Li, Q. Xian, Z. Chen, and Z. Jia, "Surface defect detection of industrial components based on improved YOLOv5s," *IEEE Sensors J.*, vol. 24, no. 15, pp. 23940–23950, Aug. 2024.

[31] K. Qiu, L. Tian, and P. Wang, "An effective framework of automated visual surface defect detection for metal parts," *IEEE Sensors J.*, vol. 21, no. 18, pp. 20412–20420, Sep. 2021.

[32] L. Xiao, B. Wu, and Y. Hu, "Surface defect detection using image pyramid," *IEEE Sensors J.*, vol. 20, no. 13, pp. 7181–7188, Jul. 2020.
[33] F. Gao, S. Li, H. You, S. Lu, and G. Xiao, "Text spotting for curved metal surface: Clustering, fitting, and rectifying," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–12, 2021.
[34] H. Fan and Y. Li, "Image recognition and reading of single pointer meter based on deep learning," *IEEE Sensors J.*, vol. 24, no. 15, pp. 25163–25174, Aug. 2024.
[35] G. Jocher. (Aug. 10, 2020). *YOLOv5*. [Online]. Available: https://github.com/ultralytics/yolov5
[36] Y. Baek, B. Lee, D. Han, S. Yun, and H. Lee, "Character region awareness for text detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9365–9374.
[37] B. Shi, X. Bai, and C. Yao, "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 11, pp. 2298–2304, Nov. 2016.
[38] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: Deformable transformers for end-to-end object detection," in *Proc. Int. Conf. Learn. Represent.*, 2020, pp. 1–16.
[39] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," 2018, *arXiv:1807.03748*.
[40] M. Liao, Z. Wan, C. Yao, K. Chen, and X. Bai, "Real-time scene text detection with differentiable binarization," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 7, 2020, pp. 11474–11481.
[41] M. Liao, Z. Zou, Z. Wan, C. Yao, and X. Bai, "Real-time scene text detection with differentiable binarization and adaptive scale fusion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 1, pp. 919–931, Jan. 2023.
[42] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.
[43] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 658–666.
[44] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2961–2969.
[45] Y. Zhu, J. Chen, L. Liang, Z. Kuang, L. Jin, and W. Zhang, "Fourier contour embedding for arbitrary-shaped text detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Apr. 2021, pp. 3123–3131.
[46] Y. Zhao et al., "DETRs beat YOLOs on real-time object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 16965–16974.

Yunseo Jeong received the B.S. degree from the Department of Intelligent Mechatronics Engineering, Sejong University, Seoul, Republic of Korea, in 2023, where she is currently pursuing the M.S. degree with the Department of Artificial Intelligence and Robotics.

Seokjun Kwon received the B.S. degree from the Department of Intelligent Mechatronics Engineering, Sejong University, Seoul, Republic of Korea, in 2023, where he is currently pursuing the M.S. degree with the Department of Convergence Engineering for Intelligent Drone.

His current research interests include computer vision and machine learning.

Jeongmin Shin (Graduate Student Member, IEEE) received the B.S. degree in intelligent mechatronics engineering from Sejong University, Seoul, South Korea, in 2022, where he is currently pursuing the joint M.S. and Ph.D. degrees with the Department of Convergence Engineering for Intelligent Drone.

Yukyung Choi received the B.S. degree from the Department of Information and Communication Electronics Engineering, Soongsil University, Seoul, Republic of Korea, in 2006, the M.S. degree from the Department of Electrical and Electronic Engineering, Yonsei University, Seoul, in 2008, and the Ph.D. degree in the electrical and electronic engineering/robotics from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Republic of Korea.

She is currently an Associate Professor with the Department of Artificial Intelligence and Robotics, Sejong University, Seoul, and the Director of the Robotics and Computer Vision (RCV) Laboratory. She is also with the Artificial Intelligence and Robotics Institute (AIRI), Seoul. Her research interests include computer vision and robotics.