Dimension Decoupling Vision-Language Transformer for Industrial Container Marking and Natural Scene Text Spotting

Zhangzhao Liang[a,1], Ying Xu[a,1], Yikui Zhai[a,*], Hufei Zhu[a,1], Jiangtao Xi[b,1], Pasquale Coscia[c] and Angelo Genovese[c]

[a]School of Electronics and Information Engineering, Wuyi University, , Jiangmen, 529020, Guangdong, China
[b]School of Electrical, Computer and Telecommunications Engineering, University of Wollongong, , Wollongong, 2522, New South Wales, Australia
[c]Depatimento di Information, Università, Degli Studi di Milano, , Milano, 20133, Lombardia, Italy

ARTICLE INFO

Keywords:
Container Marking Spotting
Text Spotting
Vision-Language Models
Transformer
Intelligent Logistics

ABSTRACT

The exceptional performance of Deep Learning has significantly advanced the widespread application of text spotting across various downstream tasks, such as electronic document recognition, traffic sign recognition, and bill number recognition. A challenging and crucial application is Container Marking Text Spotting (CMTS), which aims to swiftly capture logistics information from container surfaces and enhance the overall operational efficiency of logistics systems. Unlike the extensively studied natural scene text, text on container markings typically comprises contextless texts (such as "45G1", "CWFU 1810810"), presenting a unique spotting challenge. In addition, part of the vertical text and the widely used non-end-to-end models in this field also limit the performance of container text spotting. Overall, due to the lack of further research in the task of container text spotting, the performance of the current model is unsatisfactory. This greatly affects the intelligence and informatization of the container industry. Therefore, there is an urgent need for a high-performance and easy to deploy method to improve the spotting accuracy of container surface text. This can not only effectively reduce the cost of obtaining container information, but also improve the overall intelligence level of the industry. In this paper, we propose a Dimension Decoupling Vision-Language Transformer (DVLT) for achieving high-performance in CMTS tasks. To address the challenges of contextless texts, our approach incorporates a Semantic Augmentation Module that leverages prior knowledge without adding computational overhead during inference. Additionally, we introduce center-line proposals to enhance the model's adaptability to vertical text. Finally, DVLT improves the model's comprehensive text spotting capabilities through a novel Dimension Decoupling Decoder. DVLT is a completely end-to-end text spotting transformer, which achieved state-of-the-art on the CMTS task (dataset publicly available) and also demonstrated competitive results on well-known benchmarks such as CTW1500, ICDAR2015 and Total-Text. The code and dataset are available at: https://github.com/yikuizhai/DVLT.

1. Introduction

Artificial intelligence is gradually occupying an important position in the industrial field. With the improvement of industrial intelligence, the efficiency of industrial production and management has made significant progress [1–3]. For example, Choi et al. [4] employed text recognition algorithms to recognize the serial numbers of steel products, thereby optimizing the production process and enhancing transportation and warehouse management. The primary objective of scene text detection and recognition is to extract textual information from images, making it a vital component in acquiring industrial product data. Due to the superior performance and simpler training methods, text spotters are increasingly being adopted in systems such as visual question answering, image-text retrieval, and item information recognition, replacing the traditional "detector + recognizer" cascaded approach.

Many challenging tasks involving text spotting have also emerged, such as identifying logistics information on container. With the rapid increase in container throughput, Automated Container Terminal (ACT) and Intelligent Logistics System (ILS) urgently need a high-performance and generalizable Container Marking Text Spotting (CMTS) method to achieve automated container information management and improve operational efficiency [5]. ACT can utilize this information for scheduling loading and unloading equipment, optimizing stacking-transportation routes to achieve a better balance of resources and efficiency. While ILS can process this information to automate container transportation tasks, including tracking goods and registering their entry and exit from warehouses. Fig. 1 shows an example of CMTS.

There are three main difficulties in studying generalized and high-performance CMTS methods. Firstly, the markings on the container's surface consist predominantly of text and patterns. Unlike the well-studied natural scene texts [6, 7], these markings text numerous disordered and semantically unclear character combinations, which can significantly impair the model's accuracy in predicting subsequent characters. In addition to contextless texts, there are other key and semantically clear texts on the surface of the container. As a tool for extracting logistics information, it is necessary

*Corresponding author
zhangzhaoliang416@163.com (Z. Liang); xuying117@163.com (Y. Xu); yikuizhai@163.com (Y. Zhai); zhuhufei@aliyun.com (H. Zhu); jiangtao@uow.edu.au (J. Xi); pasquale.coscia@unimi.it (P. Coscia); angelo.genovese@unimi.it (A. Genovese)
[1]Contributed equally to this work

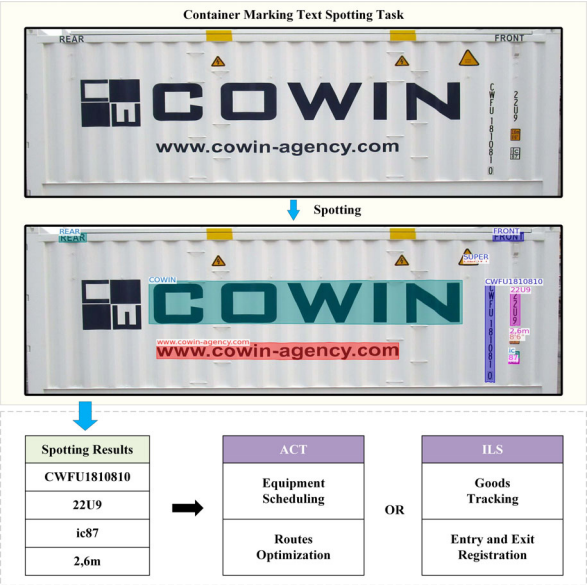

Figure 1: Explanation of CMTS. The results of spotting will provide a basis for resource scheduling and information management in ACT and ILS.

to require the model to have excellent generalization to different texts to ensure the accuracy of spotting information. Secondly, the existing text spotters for natural scene text are difficult to recognize vertical text on containers, which also poses great challenges for automated CMTS. Finally, adopting a non-end-to-end optimization increases the risk of model sub-optimization [8]. Most mainstream spotting systems currently choose to optimize text detector and text recognizer separately [9, 10], while only in the testing stage the two are cascaded to complete text spotting. This non-end-to-end model has also become widely used in the current container field, resulting in unsatisfactory performance of the algorithm in CMTS tasks.

This paper presents a high-performance and generalizable end-to-end text spotter for CMTS, called Dimension Decoupling Vision-Language Transformer (DVLT). Firstly, DVLT leverages a Vision-Language Models (VLMs) to overcome the training difficulties caused by contextless texts and vertical texts. Specifically, the Semantic Augmentation Module (SAM) uses language models to introduce prior knowledge into DVLT, improving the model's predictive ability for contextless text. This module only participates in the training phase and incurs no additional computational overhead during the inference. To address the vertical text on the container surface, we have added the Center-Line Proposals (CLP) generator in vision model, which can generate visual proposals in the form of curves. Then, the Dimension Decoupling Decoder (D^2-Dec) employs attention mechanisms to integrate features from both vision and language models comprehensively. The proposed modules ensures that the target queries obtains more semantic information and facilitates accurate predictions of contextless and vertical texts laying on the surface of container, while also maintaining excellent generalization to text in other scenarios. The Hungarian algorithm [11] is used for end-to-end training of the model. The main contributions in ours paper are as follows:

(1) DVLT, a high performance end-to-end text spotter, has been innovatively developed for automated CMTS. The proposed DVLT can reduce the cost of identifying container information in fields such as ACT and ILS, and promote the intelligence process in corresponding fields.

(2) A Vision-Language framework is employed to improve text spotting performance. We integrate the SAM to incorporate explicit semantic information, enhancing the model's ability to recognize contextless texts in containers. Additionally, we introduce the CLP generator within the Transformer-based vision model to address vertical text. Notably, this framework introduces no additional computational overhead during inference, ensuring efficient CMTS.

(3) A novel D^2-Dec was designed to fuse content information from SAM and location information from CLP generator, enabling target queries to have clearer semantics. It can significantly improve the performance of DVLT in CMTS.

(4) Extensive experiments were conducted on container dataset and public available natural scene text datasets. The experimental results indicate that DVLT outperformed other text spotters significantly in CMTS tasks, achieving state-of-the-art (SoTA) performance. At the same time, DVLT also has competitive performance in natural scene text datasets such as CTW1500 [12], ICDAR2015 [13] and Total-Text [14]. The excellent performance underscore the promising potential of the DVLT in facilitating comprehensive and intelligent CMTS.

The remainder of this paper is organized as follows: Section 2 details related work on Container Number Detection and Recognition, Text Spotter, and Vision-Language Models. The DVLT for CMTS is introduced in Section 3, with explanations of key components such as the SAM and CLP generator, D^2-Dec, and related elements. Subsequently, Section 4 presents the experimental results and analysis. Finally, the conclusions are provided in Section 5.

2. Related Work

This section first introduces the application of text detection and recognition in the container industry, then reviews the development of text spotter and vision-language models, and finally summarizes the current research gaps in CMTS tasks.

2.1. Container Number Detection and Recognition

Currently, the container marking text detection and recognition in the academic community mostly focus on box numbers. For example, Zhang et al. [15] have proposed an

adaptive fractional aggregation algorithm to remove background noise from text, but the detection and recognition module need to be optimized separately and cannot achieve true end-to-end training. Zhang et al. [16] uses Cascade R-CNN as the foundation to propose a RoIAlign method for vertical text and a parallel text recognition framework to optimize vertically arranged container numbers. This method achieved an F1-score of 96.5% in end-to-end container number recognition tasks. In addition to Deep Learning based methods, Chen et al. [17] and He et al. [18] also applied traditional image processing techniques for container number detection and recognition. However, these methods exhibit poor robustness and necessitate extensive manual design processes. Furthermore, they were constrained by the research progress of their time and lacked scientific evaluation indicators.

2.2. Text Spotter

The cascade model for text detection and recognition typically requires an independently trained detector to locate text regions, followed by a separately trained recognizer that processes the cropped image of the detected region. DBNet [9], as an independent text detection model, demonstrates outstanding instance detection performance through differentiable binarization. Shi et al. [10] integrates feature extraction, sequence modeling, and transcription into a unified framework, establishing a novel text recognition pipeline. While both models excel in their respective tasks, limitations persist when they are utilized in an end-to-end manner. These approaches not only results in a cumbersome data processing workflow but also faces performance constraints due to the lack of mutual adaptability between the two models. In contrast, text spotters enable the joint training of detection and recognition modules, mitigating the sub-optimization issues of cascaded models while offering a simpler data processing pipeline and improved performance.

MTSv1 [19] is the pioneering end-to-end text spotter for texts with arbitrary shapes. This method is built upon Mask R-CNN [20] and employs its segmentation branches to segment feature regions at the instance and character levels after Regions of Interest (RoI) alignment. The introduction of segmentation branches can not only be used for text character recognition, but also achieve more accurate feature alignment and text localization. MTSv3 [21] proposed an anchor-free Segmentation Proposal Network instead of Region Proposal Networks, and combined with Hard RoI Masking, was able to achieve feature alignment for arbitrary shape proposals. The MTS series model is a highly representative approach based on Convolutional Neural Network (CNN) in text spotting. However, with the emergence of Transformer [22], an increasing number of researchers have shifted their focus toward attention mechanisms, which offer superior performance.

Mango [23] introduced the position-aware mask attention module, which generates spatial attention on text regions using both instance-level mask attention (IMA) and character-level mask attention (CMA). IMA and CMA are answerable for perceiving the position of text and characters within the image, respectively. The positional-aware attention spectrum facilitates direct extraction of textual instance features, obviating the need for explicit cropping operations. SPTS [24] regards text detection and text recognition tasks as sequence prediction tasks. However, due to the auto-regressive characteristics of the Transformer, this method needs extensive data for long-term training, which extremely consumes computational resources and is difficult to promote. TESTR [25] regards the text spotting task as a set prediction task and proposes a single encoder and double decoder structure, separating the regression of bounding boxes and character recognition. It also proposes using bounding box to guide the polygon detection, achieving SoTA performance at the time. Raisi et al. [26] proposed two occlusion text datasets for occluded text, Occlusion Scene Text (OST) and Occluded Character level using the Total Text (OCTT) dataset. Secondly, they used MAE [27] as the backbone network, combined with Transformer-based spotting network, to achieve end-to-end training of occluded text. The Transformer-based method, with its stronger feature extraction and long-range dependency capture capabilities, greatly enhances the model's perception ability for complex natural scene text, becoming an important foundation for text spotlighting.

2.3. Vision-Language Models

Vision-Language Models (VLMs) have made significant progress in open set problems and zero-shot problems in recent years, and have received high attention. CLIP [28] proposes to encode text features corresponding to images through a language model, and then maximize the similarity between the language features of object descriptions and the visual features of object images, achieving large-scale unsupervised pre-training. This method has amazing performance in zero-shot image classification. Similar to the idea of CLIP, MDETR [29] applied maximizing the cosine similarity between text caption and image pairs in the field of object detection, which unifies object detection and phrase grounding tasks. Both utilize large-scale pre-training, greatly promoting the development of zero sample object detection field. In the field of open vocabulary object detection, ViLD [30] proposes using pre-trained CLIP to distill knowledge from object detection models. Specifically, the Mask R-CNN is first modified as a category independent detector and splitter to locate and crop the targets in the image. Next, the detection model's proposals are used as pre-trained CLIP model's input as images for maximizing the similarity with the target vocabulary embedding.

2.4. Research Gaps

Despite notable advancements in container number recognition and related research, existing methods still fail to meet industrial performance standards required for intelligent automation. The primary challenges are as follows:

(1) Limited coverage of marking elements. The majority of existing studies concentrate exclusively on

container numbers, often overlooking essential attributes such as type, load, and owner details.

(2) Sub-optimal performance resulting from non-end-to-end training. Although separately optimized text detection and recognition models achieve faster convergence and offer greater deployment flexibility, they depend on cascaded architectures that lack global optimization, which currently prevail in industrial practice.

(3) Insufficient semantic integration in end-to-end methods. While Detection Transformer (DETR) [31] have opened new avenues for end-to-end text spotting, effectively embedding explicit semantic features of container text into the model's feature learning process remains a significant challenge.

To fill the gap in current CMTS research, this paper initiates from semantic features and proposes a high-performance end-to-end text spotter for comprehensive container marking, with the goal of advancing intelligent and automated container information acquisition in domains such as ACT and ILS.

3. Methodology

This section begins by introducing the foundational method of this study, DETR. Next, Section 3.2 provides a detailed explanation of the overall DVLT's pipeline, from input to final output. Finally, Sections 3.4 through 3.6 elaborate on the composition of each module and the objective function for training.

3.1. Preliminary: DETR

DETR is an end-to-end object detection model based on the Transformer architecture, introducing a novel approach by eliminating the need for complex anchor box generation and non-maximum suppression used in traditional object detection methods. It reformulates object detection as an set prediction task, leveraging self-attention mechanisms to process global image information and directly predict the position and category of target objects. The Hungarian algorithm enables bipartite matching between the predicted and ground truth sets, facilitating fully end-to-end training. This design not only streamlines data processing but also enhances model flexibility.

The data processing pipeline of DETR is as follows. First, DETR employs a CNN to extract image features. Then, the Transformer encoder performs self-attention computations on these features, capturing their dependency relationships. Finally, the Transformer decoder, along with a task-specific head, outputs the detection results and class probabilities. This process is both efficient and straightforward. The introduction of DETR has provided novel insights and methodologies for object detection, offering broad application prospects. Inspired by DETR, we propose an innovative end-to-end text spotter that integrates language models.

3.2. Overview of DVLT

The structure of DVLT is shown in Fig. 2. Firstly, the production standards need to be encoded through a language model to obtain a content queries, while the spatial queries is output from the container image through a vision model. Then, the content queries and spatial queries jointly replace the randomly initialized target queries and enter the D^2-Dec for self-attention and cross-attention. Finally, four parallel prediction heads are used to predict the output features of the encoder, obtaining instance categories, text characters, instance bounding box coordinates, and instance center point coordinates. The prediction heads for categories and characters is a linear projection layer, with output dimensions of 2 (text or background) and 96 (character categories), respectively. The prediction heads for the bounding box and the center point are both three-layer perceptrons, with output dimensions of 4 (the offset from the predicted bounding box point to the upper and lower edges of the ground truth) and 8 (the offset from the predicted center point to the ground truth center point), respectively. A bipartite matching loss was used as the DVLT's objective function, ultimately realizing end-to-end optimization for text locating and text recognition.

3.3. Semantic Augmentation Module

The detection and recognition of contextless texts in industrial applications is a major research focus [32]. Unlike contextual texts, contextless texts lacks semantic information and prevents character prediction based on preceding characters. Therefore, relying solely on spatial queries for location information is insufficient to bridge the semantic gap between contextless and contextual texts. From a feature representation perspective, incorporating features specific to contextless texts directly into the model can enhance semantic richness. This process is analogous to adding memory of contextless texts to the human brain, where memory enables faster and more accurate reading. Similarly, guided by such "memories," neural networks can theoretically improve their recognition of contextless texts.

Due to the strict adherence to production standards established by International Organization for Standardization (ISO) during the manufacturing process of containers, these production standards specify in detail the types and contents of markings. Therefore, textual production standards can be seen as a prior knowledge, and by using the SAM to transform it into a content queries with clear semantic information. Subsequently, content queries serve as "memory" to guide the model in spotting contextless texts.

Specifically, we concatenate each text instance in the production standards to form a string of text sequences $T = \{t_1, t_2, ..., tn\}$, with blank space used as delimiter between each instance (e.g."SUPER Tare 81 ic 34,000kgs..."). Then we tokenize the text sequence with a maximum sequence length of L. In order to accommodate the large amount of text information in the container images, we set L to 512. Sequences exceeding L tokens are truncated, while those below L are padded. The tokenized sequences will be encoded

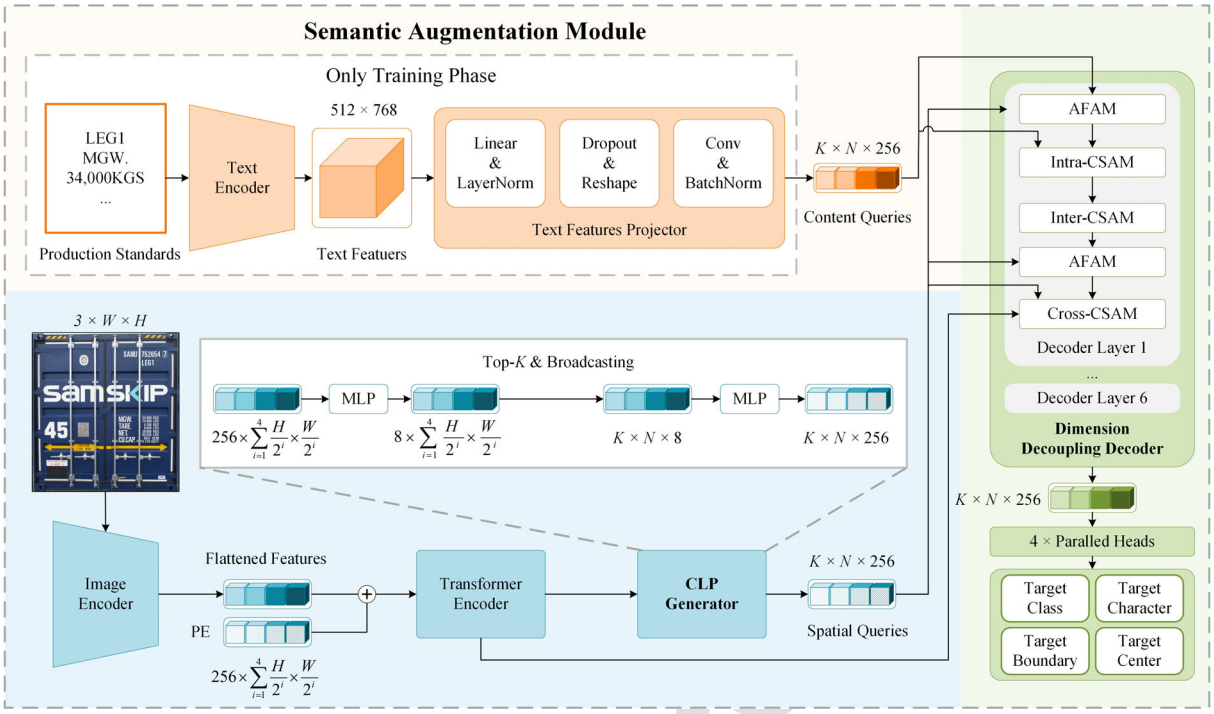

Figure 2: The architecture of DVLT.

through the text encoder to obtain text feature sequences $F_{text} \in \mathbb{R}^{L \times 768}$. Finally, the Text Features Projector (TFP) converts the text sequence $F_{text} \in \mathbb{R}^{L \times 768}$ onto the same feature space as visual embedding to obtain the content queries, denoted as $Q_c \in \mathbb{R}^{K \times N \times 256}$. The definitions and values of K and N are the same as in Section 3.4.

In application scenarios, the inference speed of the model is crucial. During training, production standards are input as textual data into the text encoder to update the content queries. Subsequently, the text encoder performs a one-time pre-encoding on all common production standards to obtain content queries and retains them in vector form as initial feature values. In order to improve the speed of the model, we use zero vectors as content queries during the inference, which can avoid additional computation caused by the language model.

3.4. Center-Line Proposals

In recent years, the prediction approach based on two-stage DETR has also been applied to text spotter [33]. The use of randomly initialized target queries in DETR to predict targets greatly increases the difficulty of learning target positions. Therefore, employing a two-stage prediction approach to optimize the initialization of randomly initialized vectors has become a key factor in enhancing the performance of DETR-based text spotters. We conducted a comprehensive study of publicly available benchmarks for natural scene text. On one hand, following general annotation conventions, text is typically located at the center of the bounding box. On the other hand, compared to rectangular bounding boxes, center-lines are more adaptable to multi-directional and vertical text. To this end, DVLT's vision model adopts a two-stage approach, utilizing the center-line of the bounding box instead of its vertices to represent text position. In this two-stage process, the Transformer encoder first generates a rough prediction of the center-line, which is then refined by the Transformer decoder. The spatial queries that used in place of randomly initialized target queries are derived from the coarse center-line proposals. Next, we provide a detailed introduction to DVLT's vision model.

The vision model consists of an image encoder, a Transformer encoder and a CLP generator. The container image is firstly processed by the image encoder for obtaining general features and four layers of feature maps with different scales are extracted. The feature maps output from the image encoder network are flattened, combined with position embeddings, and fed into the Transformer encoder to obtain the feature sequence F_{enc}. Finally, the CLP generator follows the subsequent process to generate CLP and spatial queries.

For i-th layer F_{enc}, a 3-layer Multi Layer Perceptrons (MLP) will predict 4 control points CP for each pixel: $CP_i = \{cp_{i_0}, cp_{i_1}, cp_{i_2}, cp_{i_3}\}$. In addition, each point goes through a linear layer to predict the score that belonging to the text category. We sample the (Top-$K \times 4$) points that are most likely to represent the position of the text based on the category score input from the linear layer. Then K CLP are generated by those (Top-$K \times 4$) control points according to Eq. 1. Here, each text instance in the image corresponds to at least one CLP. In order to further increase

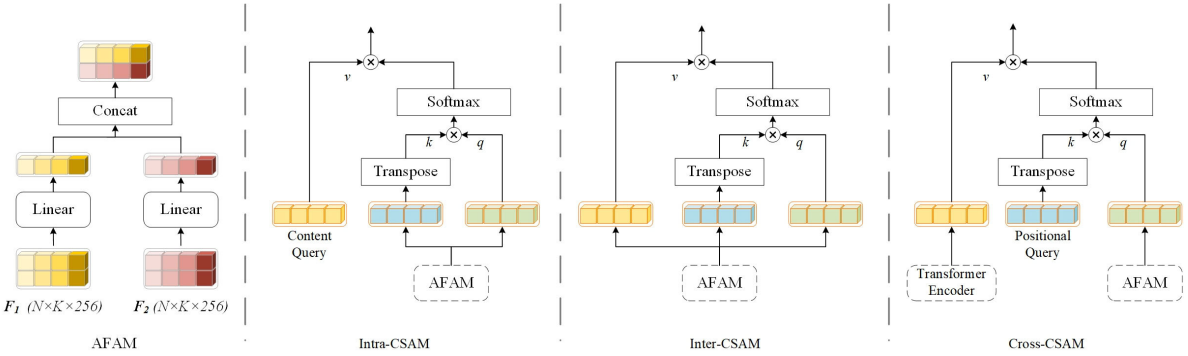

Figure 3: The detail of Dimension Decoupling Decoder.

the number of CLP, we broadcast the CLP features N times. So each text instance corresponds to at least N CLP. For all the CLP, a 2-layer MLP are applied to generate the final spatial queries $Q_s \in \mathbb{R}^{K\times N\times 256}$. K represents the number of target queries, while N denotes the number of CLP. In all experiments conducted in this study, K follows the DETR setting, where the number of target queries is fixed at 100, and N is set to 50.

$$CLP = cp_{i_0}(1-\alpha)^3 + 3cp_{i_1}(1-\alpha)^2 + 3cp_{i_2}(1-\alpha) + cp_{i_3}\alpha^3. \quad (1)$$

In Eq. 1, α is the relative position of four control points along the entire center-line, with a value range of [0,1]. For each pixel with position index (w, h) in feature map with resolution (W, H), we calculate α following $\frac{w}{W}$, which is refers to [34].

3.5. Dimension Decoupling Decoder

The randomly initialized target queries in the Transformer decoder is highly coupled in position and category information, resulting in slow convergence speed during network training. Inspired by above phenomenon, we propose the D^2-Dec, which consists of an Attention Feature Aggregation Module (AFAM), Intra-Content Spatial Attention Module (Intra-CSAM), Inter-Content Spatial Attention Module (Inter-CSAM), and Cross-Content Spatial Attention Module (Cross-CSAM).

The AFAM integrates spatial and content queries to generate target queries with precise location and semantic information. Since the new target queries originate from different models, the Intra-CSAM and Inter-CSAM modules reduce the distance between tokens in the feature space. Finally, Cross-CSAM computes the attention scores of target queries and image features to extract the final text sequence features.

The CSAM follows the calculation method of Multi-Head Attention, but in order to maintain the dimensionality separation of Q_c and Q_s, it is necessary to adjust their inputs separately. Firstly, in AFAM, linear layer is used to reduce the dimension of Q_c and Q_s to $\mathbb{R}^{K\times N\times 128}$, then Q_c and Q_s concatenated according to the last dimension to obtain the output $\in \mathbb{R}^{K\times N\times 256}$. Subsequently, the output of AFAM serves as q and k, while Q_c replaces randomly initialized v for Intra-CSAM. Here, q, k, and v denote the *query*, *key*, and *value* within the attention layer, respectively. The content queries and spatial queries become a feature sequence in AFAM that contains both semantic and positional information. Therefore, the dependency relationship between semantic information and positional information within feature sequences can be fully learned in Intra-CSAM. Different with Intra-CSAM, Inter-CSAM is primarily responsible for learning dependency information between feature sequences and ensuring discriminability between the sequences. In Cross-CSAM, the output of Inter-CSAM F_{inter} and Q_s also need to pass through AFAM first, but this time the output of AFAM is only used as q of Cross-CSAM, while k and v are the spatial queries Q_s and the encoder's output F_{enc}, respectively. The detailed process of D^2-Dec is shown in Fig. 3.

3.6. Optimization

3.6.1. Prediction and Ground Truth Matching

In general, the number of prediction significantly exceeds the actual number of targets present in the image. Therefore, the Hungarian algorithm is employed to obtain the optimal injective function f, which establishes a one-to-one correspondence between the ground truth (GT) Y and the prediction set $\hat{Y}$, thereby minimizing the matching cost C as follows:

$$\underset{f}{argmin}\sum_{g=0}^{G-1} C_{match}\left(Y^{(g)}, \hat{Y}^{(f(g))}\right), \quad (2)$$

where G is the number of GT instances in each image, g represent g-th GT and its matched instances.

$$C_{match}\left(Y^{(g)}, \hat{Y}^{(f(g))}\right) = FL\left(p^{(f(g))}\right) + CTC\left(t^{(g)}, \hat{t}^{(f(g))}\right) + \lambda_{pos} MAE\left({p_n}^{(g)}, \hat{p}_n^{(f(g))}\right), \quad (3)$$

where $p^{(f(g))}$ is the probability that the instance category is text, and $FL(p^{(f(g))})$ [35] is a classification loss used to determine whether the target is text. $CTC\left(t^{(g)}, \hat{t}^{(f(g))}\right)$ represents the Connectionist Temporal Classification (CTC) loss [36] between the GT text sequence $t^{(g)}$ and the predicted sequence $\hat{t}^{(f(g))}$. The last item represents the L1 distance between the GT point p_n and the predicted point $\hat{p}_n$.

3.6.2. Objective Function

For the GT-prediction pairs that has successfully undergone one-to-one matching according to Eqs. 2 and 3, we use the loss of instance categories, character sequences, center-line, and boundaries as the main optimization losses. The overall loss function of DVLT is:

$$L_{total} = L_{main} + L_{aux}. \tag{4}$$

L_{main} is the weighted summary of the following three parts, where k is the k-th target queries.

$$L_{main} = \sum_{k} L_{cls}^{(k)} + L_{char}^{(k)} + \lambda_{pos} L_{pos}^{(k)}. \tag{5}$$

Specifically, L_{cls}, $L_{char.}$, $L_{pos.}$ are as follow:

$$L_{cls}^{(k)} = FL\left(p^{(k)}\right), \tag{6}$$

L_{cls} is mainly used for optimizing the classification of textual and non textual objects. Subsequently, CTC loss was employed to calculate the loss between GT text sequence t and predicted text sequence $\hat{t}$:

$$L_{char}^{(k)} = CTC\left(t^{f^{-1}(k)}, \hat{t}^{(k)}\right). \tag{7}$$

Regarding the position of the instance, the loss of N points on the center-line (p) and the boundary points of top (top) or bottom (bot) is determined based on the L1 distance between the GT and the predicted point:

$$\begin{aligned} L_{pos}^{(k)} = & \sum_{n=0}^{N-1} \left\| {p_n}^{(k)} - \hat{p}_n^{(k)} \right\| \\ & + \left\| {top_n}^{(k)} - \hat{top_n}^{(k)} \right\| + \left\| {bot_n}^{(k)} - \hat{bot_n}^{(k)} \right\|. \end{aligned} \tag{8}$$

In addition to L_{main}, we also use the encoder's output to calculate the instance category loss and center point coordinate loss as auxiliary losses L_{aux}, for i-th feature sequence of encoder:

$$L_{aux} = \sum_{i} L_{cls}^{(i)} + \lambda_{pos} L_{pos}^{(i)}, \tag{9}$$

where L_{cls} and L_{pos} are defined similarly to Eqs. 6 and 8, respectively, except that their inputs are derived from the encoder output rather than the predicted head output. According to [33], λ_{pos} was set to 0.5.

Figure 4: Samples of the container marking dataset.

4. Experiment

This section begins with an introduction to the dataset, evaluation and experimental setup employed in this study. Subsequently, we assess the effectiveness of the proposed method through ablation experiments, comparative analyses, and visualizations. Lastly, we explore the integration of DVLT with existing technologies to enhance freight logistics efficiency.

4.1. Dataset

4.1.1. Container Text Dataset

To ensure data quality and maintain the objectivity of experimental results, we selected a subset, referred to as the Container Text Dataset, from the publicly available container marking dataset [37]. The primary objective of selecting a subset was to exclude low-quality images and incorrect annotations. The Container Text Dataset consists of 1,500 images and 12,857 instances. All images were captured in both day and night scenes, with a resolution of 640 × 480 to 6,420 × 4,160. The contextless text instances constitute exceeds 50% of the total text instances. The vertical text accounts for over 7%, which is relatively high compared to most public available datasets.

As for the dataset division, 1,000 images are selected as training set randomly, while the leftover 500 images constituted the testing set. Fig. 4 shows some samples of the dataset. Figs. 5 and 6 are the distribution of character and instance scale respectively. As shown in Fig. 5, the characters in the container dataset consist of uppercase and lowercase English letters, digits (0–9), and commonly used punctuation marks. In total, there are 96 character categories, corresponding to the dimensions of DVLT's recognition head.

4.1.2. Public Available Datasets

CTW1500 [12] is a comprehensive natural scene text dataset. It consists of 1,000 training images and 500 test images, which contain various curved text instances in natural scenes. Each image in the dataset contains at least one instance of curved text. The images are primarily sourced from

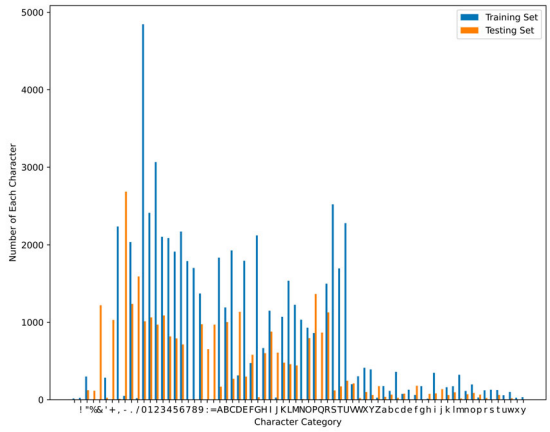

Figure 5: Character distribution map.

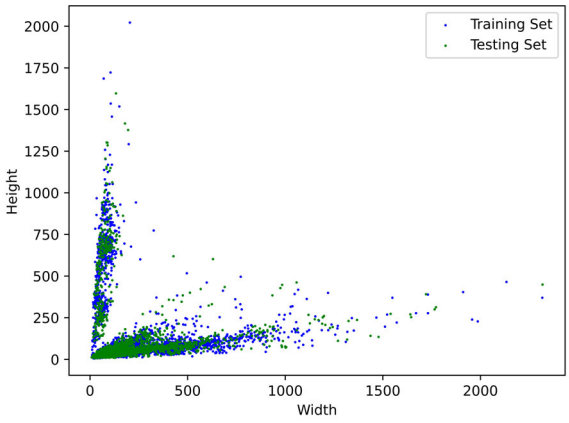

Figure 6: Scale distribution of marking instances.

Google Open Images and phone cameras, encompassing a substantial amount of horizontal and multi-directional text. The dataset exhibits diverse distributions, including indoor and outdoor scenes, as well as text that is blurry, perspective-distorted, or affected by other variations.

ICDAR2015 [13] consists of 1,000 training images and 500 test images captured by Google Glass with a resolution of 720×1280. Most scenes consist of unfocused street views or mall environments, containing quadrilateral text oriented in various directions with significant scale variations. Text instances are labeled at the word level.

Total-Text [14] comprises 1,555 images containing various text types, including horizontal, multi-directional, and curved text. The dataset is divided into a training set of 1,255 images and a testing set of 300 images. Overall, the data collected by Total-Text is similar to that of CTW1500, as both serve as commonly used benchmarks for natural scene text spotting.

4.2. Evaluation

F1-score (F1), Precision (P), and Recall (R) serve as the primary evaluation metrics for end-to-end systems [25, 33, 38, 39]. These metrics, derived from the True Positive (TP), False Positive (FP), and False Negative (FN) counts in the predictions, collectively represent the model's overall performance in text spotting. In contrast, in detection-only scenarios, the focus is solely on the model's ability to locate text. Regardless of whether it's an end-to-end or detection-only approach, F1 remains the most important comprehensive evaluation indicator. The formula for P, R and F1 metrics are defined as follows:

$$P = \frac{TP}{TP + FP}, \tag{10}$$

$$R = \frac{TP}{TP + FN}, \tag{11}$$

$$F1 = 2 \cdot \frac{P \cdot R}{P + R}, \tag{12}$$

4.3. Implemented Details

Multi-scale inputs are used in both training and testing. The minimum size of each batch of input images during training is randomly selected from [512, 640, 768, 896], while the maximum size is 1,200. During testing, the minimum input size is 1,000 and the maximum size is 1,200. All models are optimized by AdamW, with weight decay equal to 1e-4. Cosine decline learning rate update strategy is used for better convergence. Data augmentation is the composition of pixel normalization, resize, and crop.

All experiments are conducted on Intel Core I5-12600KF CPU; NVIDIA TITAN RTX 24GB GPU; Ubuntu 20.04; Python 3.8; CUDA 11.1.

4.4. Ablation Studies

Extensive experiments are conducted on the container dataset based on the experimental setup reported in Section 4.3. The initial learning rate is set to 5e-5, with a batch size of 1, and the training lasts for 400 epochs.

CLP, SAM, and D^2-Dec. We conducted ablation experiments about the corresponding modules, mainly comparing the end-to-end F1 performance before and after using the proposed modules. The combination use of CLP, SAM and D^2-Dec has made varying degrees of positive impact on the model.

The experimental results on the container dataset are presented in Table 1. Overall, the proposed modules have a positive impact on performance. Furthermore, adding more modules yields greater improvements to the model. For instance, using a single module typically results in a performance improvement of less than 1%. However, when any two modules are combined, the performance gain exceeds 1.5%.

Table 1
CLP, SAM, and D^2-Dec Ablation Experiments on Container Dataset. Red font denotes the highest value.

CLP	SAM	D^2-Dec	End-to-End			Detection Only		
			F1	P	R	F1	P	R
×	×	×	88.25	96.29	81.45	90.87	97.86	84.82
✓	×	×	88.99 (+0.74)	96.09	82.87	91.42 (+0.55)	97.83	85.81
×	✓	×	88.46 (+0.21)	96.40	81.73	91.05 (+0.18)	98.13	84.92
×	×	✓	89.45 (+1.20)	95.83	83.86	92.55 (+1.68)	97.95	87.71
✓	✓	×	89.86 (+1.61)	95.82	84.59	92.40 (+1.53)	97.80	87.57
✓	×	✓	90.94 (+2.69)	96.12	86.29	93.39 (+2.52)	97.91	89.27
×	✓	✓	90.53 (+2.28)	96.18	85.51	93.40 (+2.53)	98.13	89.11
✓	✓	✓	**91.69 (+3.44)**	96.22	87.57	**93.94 (+3.07)**	97.89	90.30

Table 2
CLP, SAM, and D^2-Dec Ablation Experiments on CTW1500.

CLP	SAM	D^2-Dec	End-to-End			Detection Only		
			F1	P	R	F1	P	R
×	×	×	79.99	91.93	70.80	87.53	91.28	84.08
✓	✓	×	80.47 (+0.48)	91.38	71.88	88.37 (+0.84)	92.28	85.64
✓	×	✓	80.68 (+0.69)	93.40	71.02	88.56 (+1.03)	92.50	84.93
×	✓	✓	80.56 (+0.57)	91.69	71.84	88.94 (+1.41)	92.04	86.05
✓	✓	✓	**80.96 (+0.97)**	92.92	71.73	**89.08 (+1.55)**	93.03	85.45

When all modules are utilized together, the model achieves optimal end-to-end and detection performance, with F1-score improvements of 3.44% and 3.07%, respectively. This is due to the strong complementary relationship between the proposed modules at the feature level. The CLP provides spatial information, while the SAM captures semantic information, both of which are essential for representing a text instance. The D^2-Dec further integrates and refines these spatial and semantic features, enhancing overall performance.

In addition to the ablation experiment conducted on the container dataset, we further validated the effectiveness of the proposed module using the CTW1500 dataset. The corresponding ablation experimental results are presented in Table 2. Although the prevalence of vertical and contextless text instances has significantly declined in common natural scene text datasets, the proposed module continues to

Table 3
Performance impact of different image encoders on Container Dataset.

Image Encoder	End-to-End			Detection Only		
	F1	P	R	F1	P	R
ResNet-50 [40]	91.69	96.22	87.57	93.94	97.89	90.30
ResNet-101 [40]	**93.40**	96.30	90.67	**95.59**	97.86	93.42
SwinTransformer-Tiny [41]	92.67	96.11	89.47	94.96	97.77	92.32
SwinTransformer-Small [41]	92.59	96.49	88.99	94.62	97.89	91.56
VisionTransformer-S [42]	92.69	96.45	89.20	94.70	98.11	91.52

Table 4
Performance impact of different text encoders on Container Dataset. "Memory" represents the GPU memory required for DVLT training.

Text Encoder	Param.	End-to-End			Detection Only			Memory*
		F1	P	R	F1	P	R	
GPT-2	FT	90.63	95.90	85.90	93.06	97.66	88.88	15.7GB
GPT-2	Frozen	90.29	96.60	84.75	92.89	96.88	89.22	10.6GB
BERT [43]	FT	91.26	96.07	86.91	93.86	97.96	90.09	14.8GB
BERT [43]	Frozen	90.27	95.95	85.23	93.28	97.31	89.58	10.6GB
RoBERTa [44]	FT	**91.69**	96.22	87.57	**93.94**	97.89	90.30	15.3GB
RoBERTa [44]	Frozen	91.09	96.06	86.61	93.01	96.07	90.14	10.7GB

enhance performance. The most notable improvement is in detection performance, which increases by 1.55%. Additionally, the end-to-end performance improves by 0.97%. The ablation experiment on natural scene text data demonstrates that DVLT is effective not only for domain-specific datasets, such as container text, but also exhibits strong generalization across other scene text datasets.

Image Encoder. We employ different vision models as the image encoder for ablation experiments, and the results are presented in Table 3. When the ResNet-101 used as the image encoder, the end-to-end F1 of the model even surpasses the Swin series and ViT-S, reaching a maximum of 93.40%. Observing the loss curves (Fig. 7(a)) of models with different image encoders, it was found that the overall loss of models using ResNet series was lower than that of Swin series and ViT-S, and the decrease was smoother. In addition, there was a significant oscillation in the loss values of the Swin series at the end of the training phase, while ViT is still a downward trend. We believe that the model's complexity is higher when using the Swin and ViT, and the same hyper-parameters as the ResNet series cannot fully converge the Swin series and ViT-S, resulting in abnormal performance of the model with different image encoders.

Text Encoder. DVLT also exhibits performance differences when using different text encoders. When using RoBERTa, the end-to-end F1 of DVLT is the highest, reaching 91.69%, while the performance is the lowest when using GPT-2, with an end-to-end F1 of only 90.63%. The results of differences text encoders are displayed in Table 4. However, according to Fig. 7(b), different text encoders have little effect on the convergence of DVLT.

Additionally, it is worth noting whether the parameters of the text encoder can be fine-tuned (FT). If the they are learnable, all text in the production standards can be encoded normally. While the parameters are frozen, only commonly used vocabulary can be encoded normally and other unordered combinations of letters and numbers are considered uncoded. Due to the fact that fully encoded production standards can provide more detailed semantic information, the text encoder participate in training is beneficial to improve performance.

CSAM of the D^2-Dec. Due to the additional introduction of semantic features generated from text, the attention module in Transformer decoder also needs to be improved to ensure the full utilization of features. Table 5 shows the ablation results of the CSAM module in the D^2-Dec. "✕" means that the semantic features in the attention module are replaced by zero vectors to ensure that the feature dimension can complete the attention calculation. The results showed that the proposed Intra-CSAM, Inter-CSAM, and Cross-CSAM have an increasing positive impact on model performance.

4.5. Container Text Comparative Studies

We selected SoTA models in the field of text spotting for comparison. The experimental settings followed Section 4.3 and ResNet-50 was used as the image encoder. The specific results are shown in Table 6. The DVLT outperformed all comparative models in the most important end-to-end indicators. The end-to-end F1, Precision, and Recall indicators exceeded the second place by 2.7%, 0.13%, and 4.7%, respectively.

It is worth noting that the cascaded model experiment of DBNet++ and SVTR-L. Both models successfully converged in their respective tasks (detection F1 = 91.07%, recognition accuracy = 96.74%). However, when the two models were cascaded, the final end-to-end F1 was only 84.52%. This difference confirms the impact of sub-optimization in non-end-to-end models on the final performance. As for the PGNet [25], adjusting hyper-parameters multiple times cannot make the model perform better. From the analysis of text direction and text type, we speculate that the bad performance of PGNet is caused by the punctuation and vertical text.

4.6. Natural Scene Text Comparative Studies

To verify the generalization of the proposed DVLT on natural text datasets, we selected the well-known CTW1500,

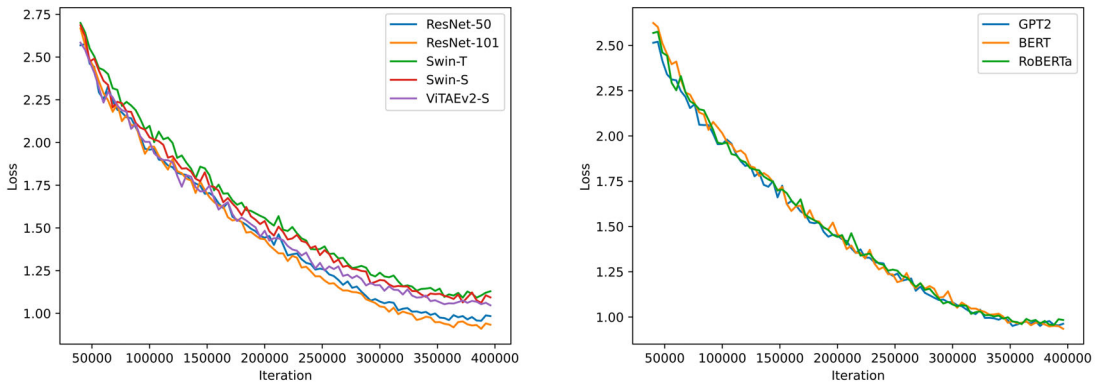

Figure 7: Loss curves for different encoders. (a) Different image encoders. (b) Different text encoders.

Table 5
Intra-CSAM, Inter-CSAM, Cross-CSAM Ablation Experiments on container Dataset.

Intra-CSAM	Inter-CSAM	Cross-CSAM	End-to-End			Detection Only		
			F1	P	R	F1	P	R
×	×	×	91.04	95.72	86.80	92.78	96.61	89.24
✓	×	×	91.17 (+0.13)	95.53	87.19	93.29 (+0.42)	96.59	90.22
✓	✓	×	91.34 (+0.30)	95.89	87.20	93.54 (+0.76)	97.01	90.31
✓	✓	✓	**91.69 (+0.65)**	96.22	87.57	**93.94 (+1.16)**	97.89	90.30

Table 6
Comparative Experiments on Container Dataset. "Inference Speed" is measured in Frames Per Second (FPS), with a higher FPS indicating faster speed. "Training Speed" represent the cost required for each epoch training (minutes / epoch), which lower is better. The performance is color-coded to highlight: red font denotes the highest value, and blue font denotes the second highest.

Method	Epoch	End-to-End			Detection Only			Training	Inference
		F1	P	R	F1	P	R	Speed ↓	Speed ↑
TESTR [25]	400	88.56	89.78	87.37	**95.71**	97.03	**94.43**	6.03	7.7
DeepSolo [33]	400	88.99	96.09	82.87	91.42	97.83	85.81	3.87	**10.1**
DBNet [9] & SVTR-L [45]	1,600	79.63	83.32	76.24	85.29	85.66	84.92	0.62 & 0.53	9.2
DBNet++ [46] & SVTR-L [45]	1,600	84.52	90.66	79.17	91.07	92.08	90.07	0.95 & 0.53	5.6
SPTSv2 [38]	800	82.77	89.32	77.12	-	-	-	12.50	6.9
ABINet++ [39]	400	82.79	88.19	78.02	92.67	**98.70**	87.33	9.37	9.4
PGNet [47]	1,200	24.82	25.89	23.83	65.89	70.95	61.49	1.40	8.2
ABCNetv2 [34]	400	88.35	96.17	87.71	94.21	97.92	90.76	5.35	6.7
Fast-TextSpotter [48]	400	76.84	79.14	74.67	95.12	**97.96**	92.44	7.2	5.9
DG-Bridge Spotter [49]	400	87.48	88.45	86.54	94.20	95.24	93.19	4.1	3.9
DVLT (ResNet-50)	400	**91.69**	**96.22**	**87.57**	93.94	97.89	90.30	5.18	**10.0**
DVLT (ResNet-101)	400	**93.40**	**96.30**	**90.67**	**95.59**	97.86	**93.42**	5.90	8.8

Table 7
Comparative Experiments on CTW1500, ICDAR2015 and Total-Text. "Full", "None", "Strong", "Weak", and "Generic" are End-to-End F1 with different lexicon, while "Detection" is the detection F1.

Method	CTW1500			ICDAR2015				Total-Text		
	Full	None	Detection	Strong	Weak	Generic	Detection	Full	None	Detection
TESTR [25]	81.5	56.0	87.1	85.2	79.4	73.6	**90.0**	83.9	73.3	86.9
SPTS [24]	83.8	63.6	-	77.5	70.2	65.8	-	82.4	74.2	-
SPTSv2 [38]	**84.3**	**63.6**	-	82.3	77.7	72.6	-	84.0	75.5	-
ABCNetv2 [34]	77.2	57.5	84.7	82.7	78.5	73.0	88.1	78.1	70.4	87.0
Swin-TextSpotter [50]	77.0	51.8	88.0	83.9	77.9	70.5	-	84.1	74.3	**88.0**
ABINet++ [39]	80.3	60.2	-	84.1	80.4	75.4	-	**84.5**	**77.6**	-
GLASS [51]	-	-	-	84.7	80.1	76.3	-	83.0	76.6	-
DVLT	81.0	63.1	**89.3**	**85.6**	**81.1**	**76.7**	**90.0**	84.2	74.2	87.4

ICDAR2015 and Total-Text datasets as performance benchmarks, and the experimental results are shown in Table 7.

In the experiment of CTW1500, DVLT's performance can rival SoTA text spotters such as SPTSv2 without using Lexicon. It is worth mentioning that the training data and duration of SPTS series models are much larger than DVLT, which also means that DVLT can achieve almost the same effect with less consumption.

In regard to ICDAR2015, the proposed DVLT outperforms most of the latest text spotters regardless of the type of lexicon used. The text instances in ICDAR2015 are generally small and often affected by blurring and backlighting. With the introduction of the CLP generator, the model effectively enhances proposal regions through two-stage prediction, improving detection accuracy and reducing the likelihood of missed detections.

Finally, in Total-Text, Similar to the results on the CTW1500 dataset, DVLT did not achieve the highest performance on the Total-Text dataset. However, its performance remains comparable to that of ABINet++ when using the Full Lexicon.

The experiments on the above three benchmarks are sufficient to prove that the proposed DVLT not only performs well in CMTS tasks, but also has good generalization in natural scene text data.

4.7. Visualization

The visualization results using ResNet-50 as image encoder are shown in Fig. 8. Characters with mixed patterns such as "CW" or characters that cannot be distinguished by the human's eye are not within the spotting range of the model, while commonly used punctuation are recognized. Our spotting results indicate that the DVLT performs well in facing challenges such as dim light environments, multi-scale text, multi-directional text, and dense text.

Figs. 9 and 10 present the spotting results of DVLT on the CTW1500 and Total-Text datasets, respectively. The comparison method shown in the visualizations corresponds to the highest-performing model for each dataset, as indicated in Table 7. The first two columns primarily compare the recognition of multi-directional and vertical text, while the last two columns focus on contextless texts recognition. Notably, SPTSv2 represents text locations using dots instead of rectangular bounding boxes. Consequently, the visualization of SPTSv2 does not display bounding boxes for the detected text.

Specifically, in Fig. 9, SPTSv2 failed to detect the vertical text "south" and "FAT." Similarly, in Fig. 10, ABINet++ was unable to recognize vertical text such as "TSINGHUA" and "THROWDOWN" in the first two columns of samples. In contrast, our DVLT model demonstrates superior adaptability to nearly vertical text.

On the other hand, DVLT achieves more accurate recognition of contextless texts in the last two columns of Figs. 9 and 10. In contrast, both SPTSv2 and ABINet++ exhibit errors in character classification and character count.

4.8. Efficiency Improvements for ACT and ILS

Timeliness and cost are critical pillars of the logistics and transportation industry. To optimize these factors, researchers have extensively investigated various stages of the logistics process, aiming to improve overall system efficiency. For instance, Hamidi et al. [52] proposed a digital maturity model to assess whether maritime logistics enterprises are ready for digital transformation, thereby enabling time and cost advantages. Xu et al. [53] developed a digital twin-based information integration platform to enhance collaborative logistics, significantly reducing the travel distance of logistics vehicles and improving equipment utilization. This study stands as one of the earliest initiatives to apply digital twin technology in the logistics domain, promoting the convergence of logistics and digital ecosystems. Lee et al. [54] takes the delivery method of goods as the starting point to study the drone delivery route in intelligent logistics, and attempts to use drones instead of manual delivery to save time. In addition, research on logistics efficiency has also

Figure 8: Visualization of DVLT's prediction results using ResNet-50 as an image encoder.

delved into the working environment of logistics workers. Molka-Danielsen et al. [55] monitored the air quality in industrial workplaces such as shipping bases to ensure the health of workers and improve the safety of their operations, thereby enhancing the operational management and production quality of shipping companies. Moreover, Sarkar et al. [56] proposed the integration of intelligent production systems with multi-objective reverse logistics in complex retail environments to minimize costs, shorten delivery times, and reduce environmental impacts, while simultaneously enhancing consumer satisfaction. This work was pioneering in combining intelligent production with reverse logistics networks and introducing a consumer-centric approach to closed-loop supply chains.

Figure 9: The spotting results of DVLT (ResNet-50) on CTW1500.

Figure 10: The spotting results of DVLT (ResNet-50) on Total-Text.

The above-mentioned studies collectively highlight the critical importance of logistics efficiency and cost effectiveness. Building on these insights, this section first evaluates the efficiency of DVLT in extracting container information through experimental analysis. It then discusses DVLT's contributions to ACT and ILS based on experimental data. Finally, it outlines practical approaches for implementing DVLT within ACT and ILS frameworks.

4.8.1. Efficiency Enhancement

To assess the efficiency enhancement of the proposed method for ACT and ILS, we compared the time required for several methods, aiming to quantify the time savings

Table 8
Efficiency comparison for ACT and ILS in spotting task. "Time", calculated according to Eqs. 13 and 14, represents the average duration required to correctly spot **one** container door image, encompassing the time taken to re-spot incorrect results.

Method	Requirement	Acc.	Time*	Efficiency
Manual	Code	100%	16.2s	1×
Manual	Code + Type	100%	22.1s	1×
Manual	All Information	94%	115.3s	1×
DeepSolo	Code	100%	0.096s	168.8×
DeepSolo	Code + Type	99%	0.318s	69.5×
DeepSolo	All Information	94%	7.0s	16.5×
DVLT	Code	100%	0.098s	165.3×
DVLT	Code + Type	100%	0.099s	223.2×
DVLT	All Information	96%	5.6s	20.6×

facilitated by the new approach. Experimental results are shown in Table 8. Specifically, we additionally captured 100 images of container doors in the container yard for this experiment. Given the significant impact of observation methods, weather conditions, and stack positioning on manual spotting in practical applications, human testers were instructed to observe and record marking information directly from container images displayed on a computer screen, thus minimizing the influence of external factors on manual spotting. The time cost calculation for all methods are as follows:

$$T_{manual} = \frac{T_m \times N_t + T_m \times N_e}{N_t}, \tag{13}$$

$$T_{auto} = \frac{T_d \times N_t + T_{manual} \times N_e}{N_t}, \tag{14}$$

$$Acc = \frac{N_t - N_e}{N_t} \times 100, \tag{15}$$

where T_{manual} and T_{auto} denote the average time required for manual and automatic methods, respectively, to correctly spot each image. T_m and T_d denote the average time taken by human testers and the DVLT / DeepSolo, respectively, to spot each image correctly on the first attempt. N_t indicates the total number of images that need spotting, while N_e represents the number of images with more than one spotting error after review. Acc represents the spotting accuracy.

4.8.2. Contributions to ACT and ILS

Container marking information plays a crucial role in the loading, route planning, and resource allocation [57]. The DVLT can bring significant transformative contributions in ACT and ILS, as its efficient (223.2× faster than manual) and accurate (96% *Acc.* in all information spotting) container spotting capabilities will reshape the workflow from the bottom. The specific contributions are as follows:

(1) During the loading and unloading, the real-time spotting capability of DVLT significantly enhances operational efficiency. As a container passes beneath the dock crane, the system instantly verifies whether the container number matches the container type code and automatically triggers the corresponding loading or unloading instructions. Specifically, If a mismatch is detected between the refrigerated container's type code and the booking order, the system can immediately halt the operation and issue an alarm. This prevents cold chain disruptions caused by incorrect loading and reduces truck detention time associated with manual verification [58].

(2) In route planning, DVLT provides a robust data foundation for dynamic path optimization. Traditional logistics systems rely on pre-declared container data, whereas DVLT captures all information of container, such as hazardous goods marking and height marking, enabling the system to proactively adjust transportation strategies. For example, when a container is marked with a height limit sign, the routing engine can automatically avoid low-clearance tunnels. If a "hazardous chemical" code is detected, it can dynamically allocate isolation storage zones and designate specialized transportation routes. This physics-based decision-making mechanism significantly reduces the risk of blind spots associated with manual inspection [59].

(3) For the resource allocation, DVLT's efficient data collection enables precise mapping of logistics resources. The yard bridge scheduling system can automatically calculate the utilization rate of yard space through real-time spotting of container numbers and size data [60]. In addition, this significant efficiency difference enables the system to redirect human resources initially used for spotting and confirmation to high value-added positions such as exception handling, customer service, or process optimization. More importantly, the "transience" of the identification process eliminates key bottlenecks on the logistics assembly line, compressing container turnover time from hour level to minute level.

4.8.3. Applied to ACT and ILS

Visual perception devices can be installed by ACT or ILS practitioners at key logistics nodes, such as gates, shore bridges, and yard bridges. These devices upload images via the network to cloud or local servers deployed with DVLT for processing. The results can then be integrated with the terminal operating system to enable basic application deployment. Thanks to extensive training on real-world data and its strong generalization capability, DVLT can meet the requirements of most ACT or ILS scenarios while avoiding

the labor-intensive process of scene-specific customization. When customization is necessary, DVLT can be further optimized using a small amount of target domain data, leveraging the pre-trained weights provided in this paper.

4.9. Managerial Insight

The implementation of ours DVLT for container information acquisition offers substantial practical benefits, aligning with the principles of industrial information integration. This section delineates the key managerial insights derived from our study.

End-to-End Logistics Automation. High performance CMTS serves as a critical "eye" in automated port operations. It seamlessly integrates crane automation, unmanned vehicles, yard management systems, and port operating systems. This technology eliminates the bottlenecks caused by manual container number entry and information verification, significantly reducing container turnover time at gates, yards, and quay cranes. It enables seamless gate passage, precise yard positioning, and automatic confirmation of loading and unloading instructions, thereby elevating the throughput capacity of ports and terminals to a new level.

Cost Reduction. According to the experimental results in Section 4.8.1, DVLT demonstrates higher recognition accuracy than manual methods and operates with an efficiency approximately 20 to 200 times greater. DVLT substantially reduces the costs associated with misshipments, missed deliveries, demurrage fees, and dispute resolution due to spotting errors. It also significantly lowers dependence on highly skilled and labor-intensive manual verification roles, thereby directly reducing labor costs as well as related investments in training, management, and safety.

Tracking Capability Enhancement. By automatically and accurately capturing container information in real time at each key node of the container circulation process, a real-time and precise dynamic container database can be established, enabling end-to-end visual tracking of goods. This significantly reduces the risk of lost or misshipped items, provides more reliable data support for customs and security, and enhances overall compliance and safety.

Optimizing Decision-Making and Resource Allocation. The massive and accurate container number data obtained through automatic recognition serves as a valuable data asset when combined with information such as container type, status, location, and timestamp. Leveraging this real-time data enables the optimization of yard space utilization, prediction of peak workloads, and dynamic adjustment of equipment and human resource deployment, thereby facilitating more accurate analysis of operational efficiency.

5. Conclusion

In the container industry, extracting container marking information is a fundamental task. Although some studies on container number recognition have achieved impressive results, these methods are not well-suited for comprehensive marking spotting. The contextless and vertical text in container marking are primary factors contributing to the poor performance of existing methods. The lack of efficient and low-cost CMTS methods greatly hinders the further development of container related fields. In this paper, DVLT is proposed to solve the task of automated CMTS. The conclusions are as follows:

(1) We consider the contextless and vertical texts in container marking and propose vision-language based DVLT. The proposed DVLT is a high-performance, end-to-end text spotter capable of locating and recognizing all container markings. It serves as an excellent foundational model for information extraction tasks in container-related domains such as ACT and ILS.

(2) In the ours VLMs framework, we employ a language model-based SAM to encode container production standards into content queries, providing additional semantic information. This SAM only participates in the training phase of the model and will not have any impact on the inference speed of the model. Besides, we introduce CLP for two-stage prediction and generate spatial queries for the visual model, which can generate visual proposals in the form of curves.

(3) We design a D^2-Dec to replace the randomly initialized target queries in DETR by integrating spatial and content queries. D^2-Dec enables full alignment and fusion of multimodal features from visual and language models, ensuring a strong correspondence between textual semantics and visual representations. This alignment significantly enhances the text spotting performance of DVLT.

(4) Extensive experiments are conducted to demonstrate the effectiveness of the proposed DVLT. Ours DVLT achieved a End-to-End F1 of 91.69% in the CMTS task, achieving SoTA performance. In addition, we are also validating the generalization of DVLT on three publicly available benchmarks, CTW1500, ICDAR2015 and Total-Text. The results indicate that even in natural scene text datasets, the proposed DVLT still has performance comparable to SoTA.

CRediT authorship contribution statement

Zhangzhao Liang: Data curation, Formal analysis, Investigation, Methodology, Validation, Visualization, Writing – original draft. **Ying Xu**: Formal analysis, Methodology, Writing - Review & Editing. **Yikui Zhai**: Investigation, Data curation, Writing - Review & Editing. **Hufei Zhu**: Investigation, Supervision. **Jiangtao Xi**: Resources, Supervision. **Pasquale Coscia**: Writing - Review & Editing. **Angelo Genovese**: Writing - Review & Editing, Supervision.

Declaration of competing interest

The authors have no conflicts of interest to declare that are relevant to the content of this article.

Data availability

The dataset and source codes for this work are publicly available at: https://github.com/yikuizhai/DVLT.

Acknowledgments

This work was funded by Guangdong Basic and Applied Basic Research Fund (2021A1515011576), Guangdong Higher Education Innovation and Strengthening School Project (2020ZDZX3031, 2022ZDZX1032, 2023ZDZX1029, 2024ZDZX1009), Wuyi University Hong Kong and Macao Joint Research and Development Fund (2022WGALH19), Guangdong Jiangmen Science and Technology Research Project (2220002000246, 2023760300070008390).

References

[1] Hu, Y., Wang, J., Wang, X., Sun, Y., Yu, H., Zhang, J., 2024. Real-time evaluation of the blending uniformity of industrially produced gravelly soil based on cond-yolov8-seg. J. Ind. Inf. Integr. 39, 100603.

[2] Wang, F., Chi, X., Wei, L., Song, Y., Yang, Z., 2024. Multi-position industrial defect inspection using self-training siamese networks with mix strategies. J. Ind. Inf. Integr. 40, 100615.

[3] Guo, F., Liu, J., Qian, Y., Xie, Q., 2024. Rail surface defect detection using a transformer-based network. J. Ind. Inf. Integr. 38, 100584.

[4] Choi, H., Yun, J.P., Kim, B.J., Jang, H., Shin, W., Kim, S.W., 2024. Steel product number recognition framework using semantic mask-conditioned diffusion model with limited data. J. Ind. Inf. Integr. 38, 100559.

[5] Sun, P.Z.H., et al., 2023. Agv-based vehicle transportation in automated container terminals: A survey. IEEE Trans. Intell. Transp. Syst. 24, 341–356.

[6] Zhang, S.X., et al., 2024. Arbitrary shape text detection via boundary transformer. IEEE Trans. Multim. 26, 1747–1760.

[7] Li, J.N., et al., 2024. VOLTER: visual collaboration and dual-stream fusion for scene text recognition. IEEE Trans. Multim. 26, 6437–6448.

[8] Li, H., et al., Oct. 2017. Towards end-to-end text spotting with convolutional recurrent neural networks, in: Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), pp. 5248–5256.

[9] Liao, M., et al., Feb. 2020. Real-time scene text detection with differentiable binarization, in: Proc. AAAI Conf. Artif. Intell., pp. 11474–11481.

[10] Shi, B., et al., 2017. 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. 39, 2298–2304.

[11] Kuhn, H.W., 2010. The hungarian method for the assignment problem, in: 50 Years of Integer Programming 1958-2008 - From the Early Years to the State-of-the-Art, pp. 29–47.

[12] Liu, Y., et al., 2017. Detecting curve text in the wild: New dataset and new solution. arXiv preprint arXiv:1712.02170 .

[13] Karatzas, D., et al., Aug. 2015. ICDAR 2015 competition on robust reading, in: Proc. Int. Conf. Document Anal. Recognit.(ICDAR), pp. 1156–1160.

[14] Chng, C.K., Chan, C.S., Nov. 2017. Total-text: A comprehensive dataset for scene text detection and recognition, in: Proc. Int. Conf. Document Anal. Recognit.(ICDAR), pp. 935–942.

[15] Zhang, R., et al., 2020. An adaptive deep learning framework for shipping container code localization and recognition. IEEE Trans. Instrum. Meas. 70, 1–13.

[16] Zhang, R., Bahrami, Z., Wang, T., Liu, Z., 2021. A vertical text spotting model for trailer and container codes. IEEE Trans. Instrum. Meas. 70, 1–13.

[17] Chen, M., et al., 2011. Hidden-markov-model-based segmentation confidence applied to container code character extraction. IEEE Trans. Intell. Transp. Syst. 12, 1147–1156.

[18] He, Z., et al., 2005. A new automatic extraction method of container identity codes. IEEE Trans. Intell. Transp. Syst. 6, 72–78.

[19] Liao, M., Lyu, P., He, M., Yao, C., Wu, W., Bai, X., 2021. Mask textspotter: An end-to-end trainable neural network for spotting text with arbitrary shapes. IEEE Trans. Pattern Anal. Mach. Intell. 43, 532–548.

[20] He, K., Gkioxari, G., Dollar, P., Girshick, R., Oct. 2017. Mask r-cnn, in: Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), pp. 2961–2969.

[21] Liao, M., et al., Aug. 2020. Mask textspotter v3: Segmentation proposal network for robust scene text spotting, in: Proc. Eur. Conf. Comput. Vis. (ECCV), pp. 706–722.

[22] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I., Dec. 2017. Attention is all you need, in: Proc. Neural Inf. Process. Syst. (NIPS), pp. 5998–6008.

[23] Liao, M., Shi, B., Bai, X., Wang, X., Liu, W., Feb. 2021. Mango: A mask attention guided one-stage scene text spotter, in: Proc. AAAI Conf. Artif. Intell., pp. 2467–2476.

[24] Peng, D., et al., Oct. 2022. SPTS: single-point text spotting, in: Proc. ACM Int. Conf. Multimedia, pp. 4272–4281.

[25] Zhang, X., et al., Jun. 2022. Text spotting transformers, in: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 9509–9518.

[26] Raisi, Z., Zelek, J.S., Jun. 2022. Occluded Text Detection and Recognition in the Wild, in: Proc. Conf. Robots Vis., pp. 140–150.

[27] He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.B., Jun. 2022. Masked autoencoders are scalable vision learners, in: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 15979–15988.

[28] Radford, A., et al., Jul. 2021. Learning transferable visual models from natural language supervision, in: Proc. Int. Conf. Mach. Learn. (ICML), pp. 8748–8763.

[29] Kamath, A., et al., Oct. 2021. MDETR - modulated detection for end-to-end multi-modal understanding, in: Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), pp. 1760–1770.

[30] Gu, X., et al., 2021. Open-vocabulary object detection via vision and language knowledge distillation. arXiv preprint arXiv:2104.13921 .

[31] Carion, N., et al., Aug. 2020. End-to-end object detection with transformers, in: Proc. Eur. Conf. Comput. Vis. (ECCV), pp. 213–229.

[32] Guan, T., et al., 2022. Industrial scene text detection with refined feature-attentive network. IEEE Trans. Circuits Syst. Video Technol. 32, 6073–6085.

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

[34] Liu, Y., Shen, C., Jin, L., He, T., Chen, P., Liu, C., Chen, H., 2022. Abcnet v2: Adaptive bezier-curve network for real-time end-to-end text spotting. IEEE Trans. Pattern Anal. Mach. Intell. 44, 8048–8064.

[35] Li, H., et al., Oct. 2017. Focal loss for dense object detection, in: Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), pp. 2999–3007.

[36] Graves, A., et al., Jun. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks, in: Proc. Int. Conf. Mach. Learn. (ICML), pp. 369–376.

[37] Xu, Y., Liang, Z., Liang, Y., Li, X., Pan, W., You, J., Long, Z., Zhai, Y., Genovese, A., Piuri, V., Scotti, F., 2024. Data-driven container marking detection and recognition system with an open large-scale scene text dataset. IEEE Trans. Emerg. Top. Comput. Intell. 8, 3368–3381.

[38] Liu, Y., et al., 2023. SPTS v2: Single-point scene text spotting. IEEE Trans. Pattern Anal. Mach. Intell. 45, 15665–15679.

[39] Fang, S., et al., 2023. Abinet++: Autonomous, bidirectional and iterative language modeling for scene text spotting. IEEE Trans. Pattern Anal. Mach. Intell. 45, 7123–7141.

[40] He, K., Zhang, X., Ren, S., Sun, J., Jun. 2016. Deep residual learning for image recognition, in: Proc. IEEE/CVF Conf. Comput.

Vis. Pattern Recognit. (CVPR), pp. 770–778.
[41] Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B., Oct. 2021. Swin transformer: Hierarchical vision transformer using shifted windows, in: Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), pp. 9992–10002.
[42] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N., May. 2021. An image is worth 16x16 words: Transformers for image recognition at scale, in: Proc. Int. Conf. Learn. Represent., (ICLR).
[43] Devlin, J., Chang, M., Lee, K., Toutanova, K., Jun. 2019. BERT: pre-training of deep bidirectional transformers for language understanding, in: Proc. NAACL-HLT, pp. 4171–4186.
[44] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V., 2019. Roberta: A robustly optimized BERT pretraining approach. arXiv preprint arXiv:1907.11692 .
[45] Du, Y., et al., Jul. 2022. SVTR: scene text recognition with a single visual model, in: Proc. Int. Joint Conf. Artif. Intell. IJCAI, pp. 884–890.
[46] Liao, M., et al., 2023. Real-time scene text detection with differentiable binarization and adaptive scale fusion. IEEE Trans. Pattern Anal. Mach. Intell. 45, 919–931.
[47] Wang, P., et al., Feb. 2021. Pgnet: Real-time arbitrarily-shaped text spotting with point gathering network, in: Proc. AAAI Conf. Artif. Intell., pp. 2782–2790.
[48] Das, A., Biswas, S., Pal, U., Lladós, J., Bhattacharya, S., Dec. 2024. Fasttextspotter: A high-efficiency transformer for multilingual scene text spotting, in: Proc. Pattern Recog. Int. Conf. (ICPR), pp. 135–150.
[49] Huang, M., Li, H., Liu, Y., Bai, X., Jin, L., Jun. 2024. Bridging the gap between end-to-end and two-step text spotting, in: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 15608–15618.
[50] Huang, M., et al., Jun. 2022. Swintextspotter: Scene text spotting via better synergy between text detection and text recognition, in: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 4583–4593.
[51] Ronen, R., Tsiper, S., Anschel, O., Lavi, I., Markovitz, A., Manmatha, R., Oct. 2022. GLASS: global to local attention for scene-text spotting, in: Proc. Eur. Conf. Comput. Vis. (ECCV), pp. 249–266.
[52] Hamidi, S.M.M., Hoseini, S.F., Gholami, H., Kananizadeh-Bahmani, M., 2024. A three-stage digital maturity model to assess readiness for blockchain implementation in the maritime logistics industry. J. Ind. Inf. Integr. 41, 100643.
[53] Xu, L., Mak, S., Schoepf, S., Ostroumov, M., Brintrup, A., 2025. Multi-agent digital twinning for collaborative logistics: Framework and implementation. J. Ind. Inf. Integr. 45, 100799.
[54] Lee, H., Lee, C., 2023. Research on logistics of intelligent unmanned aerial vehicle integration system. J. Ind. Inf. Integr. 36, 100534.
[55] Molka-Danielsen, J., Engelseth, P., Wang, H., 2018. Large scale integration of wireless sensor network technologies for air quality monitoring at a logistics shipping base. J. Ind. Inf. Integr. 10, 20–28.
[56] Sarkar, B., Bhattacharya, S., Sarkar, M., 2025. Integrating smart production and multi-objective reverse logistics for the optimum consumer-centric complex retail strategy towards a smart factory's solution. J. Ind. Inf. Integr. 46, 100856.
[57] Golpîra, H., Khan, S.A.R., Safaeipour, S., 2021. A review of logistics internet-of-things: Current trends and scope for future research. J. Ind. Inf. Integr. 22, 100194.
[58] Vilas-Boas, J.L., Rodrigues, J.J., Alberti, A.M., 2023. Convergence of distributed ledger technologies with digital twins, iot, and ai for fresh food logistics: Challenges and opportunities. J. Ind. Inf. Integr. 31, 100393.
[59] Rahmanifar, G., Mohammadi, M., Golabian, M., Sherafat, A., Hajiaghaei-Keshteli, M., Fusco, G., Colombaroni, C., 2024. Integrated location and routing for cold chain logistics networks with heterogeneous customer demand. J. Ind. Inf. Integr. 38, 100573.
[60] Wu, W., Shen, L., Zhao, Z., Harish, A.R., Zhong, R.Y., Huang, G.Q., 2023. Internet of everything and digital twin enabled service platform for cold chain logistics. J. Ind. Inf. Integr. 33, 100443.