Ensemble Learning for Vietnamese Scene Text Spotting in Urban Environments

Hieu Nguyen[1,2,†], Cong-Hoang Ta[1,2,†], Phuong-Thuy Le-Nguyen[1,2,†],
Minh-Triet Tran[1,2], Trung-Nghia Le[1,2,*]
[1] *Faculty of Information Technology, University of Science, VNU-HCM, Vietnam*
[2] *Vietnam National University, Ho Chi Minh City, Vietnam*

***Abstract*—This paper presents a simple yet efficient ensemble learning framework for Vietnamese scene text spotting. Leveraging the power of ensemble learning, which combines multiple models to yield more accurate predictions, our approach aims to significantly enhance the performance of scene text spotting in challenging urban settings. Through experimental evaluations on the VinText dataset, our proposed method achieves a significant improvement in accuracy compared to existing methods with an impressive accuracy of 5%. These results unequivocally demonstrate the efficacy of ensemble learning in the context of Vietnamese scene text spotting in urban environments, highlighting its potential for real-world applications, such as text detection and recognition in urban signage, advertisements, and various text-rich urban scenes.**

***Index Terms*—Ensemble learning, scene text spotting, Vietnamese scene text**

I. Introduction

The detection and recognition of text in images, namely *scene text spotting*, represent a highly formidable task in computer vision [1], demanding the precise localization and identification of text sequences within real-world contexts [2]. The implications of scene text spotting are far-reaching, spanning crucial applications such as key entities recognition [3], autonomous driving [4], intelligent navigation [5, 6], etc. Scene text spotting has been recently drawing much attention from the community, resulting in rapid investigations [7, 8].

Nonetheless, the intricacies of the Vietnamese language, with its rich set of characters and diacritics, pose significant challenges for scene text spotting. Certain Vietnamese characters, especially when accompanied by diacritics, exhibit visual similarities that can lead to confusion, such as, 'ă' and 'â', 'ô' versus 'ó', 'é' and 'ê'. Consequently, conventional individual models may encounter limitations when it comes to accurately detecting and recognizing Vietnamese scene text.

In addition, the realm of scene text spotting in Vietnamese urban environments is rife with challenges for existing methods [9, 1] In bustling city contexts, scene texts are frequently occluded, either partially or entirely, due to the presence of diverse surrounding objects such as trees, towers, traffic signs, electric poles, etc. The variability in lighting conditions, encompassing weather fluctuations and low contrast scenarios caused by background reflections, further exacerbates the difficulty of scene text spotting.

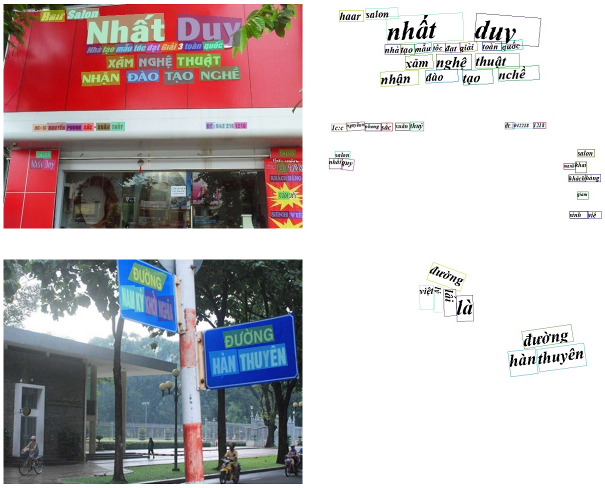


Fig. 1: Scene text spotting in Vietnamese urban environments poses various challenges, such as obscured by trees and perspective-shifted.

Existing approaches predominantly rely on individual models [10, 11, 2, 7], each exhibiting varying degrees of success across different subsets of images. However, when faced with the complexities of Vietnamese urban environments, an intricate domain with large and diverse contexts, such single-model solutions often yield limited results.

To overcome these limitations, we propose an ensemble learning framework, a powerful technique that has shown remarkable efficacy in various computer vision tasks [12, 13]. Our ensemble learning framework combines multiple state-of-the-art methods, leveraging their individual strengths and mitigating their weaknesses. Indeed, each method is tailored to address distinct

†Equal contributors
*Corresponding author. Email address: ltnghia@fit.hcmus.edu.vn

challenges related to text detection, recognition, and the intricacies of the Vietnamese script. By aggregating their predictions, we can significantly enhance the performance of Vietnamese scene text spotting in urban environments. Extensive experiments on the VinText dataset [10] show that our proposed ensemble learning framework outperforms individual models for Vietnamese scene text spotting, demonstrating superior accuracy and robustness in challenging urban environments by boosting the accuracy up to 5%. The experimental findings underscore the potential of ensemble learning as a powerful tool for advancing scene text spotting in dynamic urban environments.

Our paper contributes to the field of Vietnamese scene text spotting in urban environments in the following ways:

- We propose a novel ensemble learning framework specifically designed for Vietnamese scene text spotting, addressing the complexities of urban environments and the unique characteristics of the Vietnamese script.
- We conduct thorough experiments using an extensive dataset encompassing diverse urban scenes in Vietnam. Widely adopted in the research community, this dataset enables a comprehensive evaluation of our proposed approach and facilitates fair comparisons with existing methods. Extensive experimental findings demonstrate the utility and superiority of our ensemble approach, exhibiting higher accuracy and enhanced robustness in challenging urban contexts.

II. RELATED WORK

A. Scene Text Spotting

In recent decades, the growth of deep learning significantly contributes to the advancement of scene text spotting. There are typically approaches: two-stage and end-to-end approaches. The two-stage approach consists of two major stages: scene text detection and scene text recognition. A plethora of effective algorithms have been proposed for each stage, including DB/DB++ [14, 15], SAST [16], EAST [17], etc. for detection and SPIN [18], SRN [19], SVTR [20], ABINet [21], etc. for recognization. Efforts have also been directed towards integrating detection and recognition processes, with works like TextBoxes by Liao et al. [22] combining a single-shot detector and a text recognizer. Furthermore, Nguyen et al. [10] leveraged a dictionary to generate a list of potential results and identify the most visually compatible outcome with the text's appearance to train and improve the recognition stage. Additionally, He et al. contributed VinText dataset [10], a challenging benchmark for Vietnamese scene text spotting. Despite these advancements, existing techniques treat detection and recognition as independent tasks, lacking seamless information exchange between them.

Meanwhile, end-to-end approach focuses on merging detection and recognition into a unified system. Peng et al. [23] presented an end-to-end scene text spotting method that approaches scene text spotting as a sequence prediction task. Wang et al. proposed PGNet [24], which revolves around developing a model that combines a detection unit and a recognition module, allowing for shared CNN features and joint training. Huang et al. [1] introduced a novel end-to-end scene text spotting framework called SwinTextSpotter. By integrating both functionalities into a single algorithm, the resulting end-to-end model becomes more compact and efficient, leading to improved speed and performance recognition.

B. Ensemble Learning

Ensemble learning represents a powerful methodology that amalgamates the strengths and benefits of multiple approaches, culminating in a superior model. As early as 1996, Rosen proposed [25] a technique called decorrelation network training to enhance the accuracy of regression learning in ensemble neural networks. Deng et al. [26] introduced linear and log-linear stacking methods for ensemble learning, focusing on applications to speech class posterior probabilities computed by convolutional, recurrent, and fully-connected deep neural networks. Recently, Casado-García and Heras [27] explored ensemble methods for object detection, addressing the challenges associated with ensembling object detectors, such as the limitations of existing ensemble approaches that depended on specific detection models or frameworks. Leveraging the advantage of different methods, we introduce an ensemble learning framework to integrate results from individual models of both the two-stage and end-to-end approaches to improve the performance of scene text spotting.

III. PROPOSED METHOD

A. Overview

We present an ensemble learning framework designed to effectively combine the outputs of multiple scene text spotting methods. Fig. 2 illustrates the overview of our method, which consists of three main components: data converter, base models, and meta-model.

In the first stage of the workflow, we transform the raw images and labels of the VinText dataset [10] into different formats suitable for training and predicting of each base model. The data used to train the recognition model is images with a single word, while the dataset includes images with many different text lines which contain more than one word. Therefore, we crop the

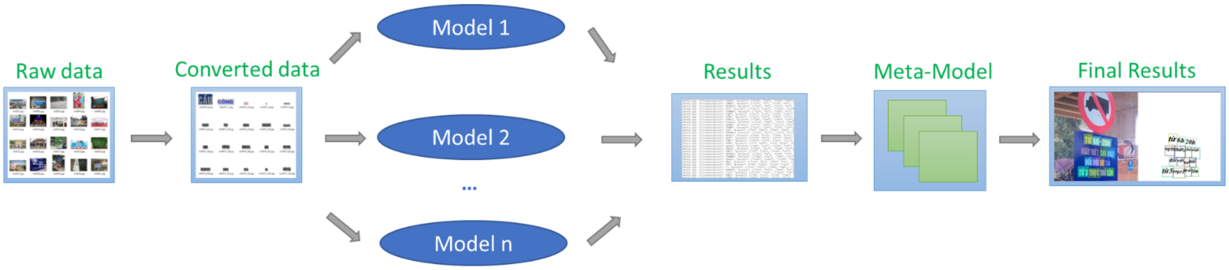


Fig. 2: Workflow of the proposed ensemble learning framework for Vietnamese scene text spotting.

original images to smaller images that contain only one word and also create new labels for the new images.

After that, base models are trained on converted data and used for prediction to obtain initial results. Subsequently, the results of all models are combined with an ensemble technique and are essential data to create a meta-model. The meta-model is then used for prediction and generates final result, which is expected to be better than the initial ones.

B. *Text Box Combination:*

Our posed Meta-model undergoes two processes: merging non-overlapping text boxes and merging overlapping text boxes. Given an image and n base models, the input of our ensemble algorithm is a list of prediction results: $D = [D_1, \dots, D_n]$ where each D_i, with $i \in \{1, \dots, n\}$, is a list of predicted text boxes $D_i = [B_1, B_2, \dots]$. In general, each D_i comes from the predictions of a scene text spotting model using a particular method M_i for a given image.

1) Merging non-overlapping text boxes: Firstly, we combine the non-overlapping text boxes belonging to different D_i results together to form the result D_s:

$$D_s = D_1 \cup D_2 \cup \ldots \cup D_n, \tag{1}$$

where pair of non-overlapping text boxes (B_h, B_k) are defined by:

$$\begin{cases} \forall P_o \in B_h & \implies P_o \notin B_k, \\ \forall P_o \in B_k & \implies P_o \notin B_h, \end{cases} \tag{2}$$

where B_h, B_k are text boxes consisting of the coordinates of four points at the four corners of the text box, starting from the bottom left corner in counterclockwise order. P_o is the coordinates of a point in Oxy, which is one of four points of a text box (B).

2) Merging overlapping text boxes: Subsequently, we combine the overlapping text boxes belonging to different D_i results based on the Intersection over Union (IoU). IoU is utilized to determine whether to merge two text boxes into one or separate them into two distinct text boxes.

Given two bounding boxes, denoted as B_h and B_k, the IoU formula is used to determine the overlapped region between them, which is represented by the equation:

$$IoU_{(B_h, B_k)} = \frac{S(B_h \cap B_k)}{S(B_h \cup B_k)}, \tag{3}$$

where $S(\cdot)$ is the area of the text box. From empirically experimental findings, we notice that when two text boxes contain different words, IoU of them is always less than 0.5. Therefore, we propose the following formula to ensemble two overlapping text boxes:

$$\begin{cases} B_s = B_h \cap B_k & \text{if } IoU \geq 0.5, \\ B_n = B_h - (B_h \cap B_k) & \text{if } IoU < 0.5, \\ B_m = B_k - (B_h \cap B_k) & \text{if } IoU < 0.5, \end{cases} \tag{4}$$

where B_n, B_m, and B_s are new text boxes after merging two text boxes B_h and B_k.

IV. EXPERIMENTS

A. *Implementation Details*

All experiments were performed on Google Colab with the following configuration GPU T4 and 13GB of RAM. Detection methods were pre-trained on the ICDAR 2015 [28] for $1,000$ epochs. The initialized learning rate was 7×10^{-3} and followed a decay learning rate schedule with a decay factor of 0.9. After that, the models were fine-tuned for 30 epochs on VinText dataset [10], and weight decay regularization was applied with a value of 10^{-4}. The dataset underwent decoding in BGR format, and various data augmentation techniques were applied, such as horizontal flipping, affine transformations (rotations between $-10°$ to $10°$), and resizing between 0.5 and 3 times the original size.

Recognition methods were pre-trained on MJSynth [29, 30] and SynthText [31] datasets. After that, they underwent a total of 20 epochs of fine-tuning on the VinText dataset [10]. We utilized the Adam optimizer [32] with a β_1 value of 0.9 and β_2 value of 0.99. Gradient clipping with a norm threshold of 20.0 was applied to prevent exploding gradients. The learning rate followed a piecewise schedule, decaying at epoch

TABLE I: Result of detection methods on the VinText dataset [10]. 1^{st} and 2^{nd} places are shown in **blue** and **red**, respectively.

Method	Backbone	Pre-trained Model	Fine-tuned on VinText [10]	Precision	Recall	Hmean
SAST [16]	ResNet50_vd [33]	Total-Text [34]	✓	87.82	53.40	66.40
				86.09	59.98	70.70
		ICDAR2015 [28]	✓	**89.53**	70.56	**78.92**
				87.45	56.43	68.59
DB++ [15]	ResNet50 [35]	ICDAR2015 [28]	✓	**92.17**	**71.17**	**80.32**
				89.05	65.18	75.26
DB [14]	ResNet50_vd [33]	ICDAR2015 [28]	✓	85.56	65.17	73.97
				84.36	56.43	67.62
	MobileNetV3 [36]		✓	78.73	65.34	71.41
				76.71	50.44	60.86
EAST [17]	ResNet50_vd [33]	ICDAR2015 [28]	✓	67.00	**71.86**	69.35
				62.59	54.73	58.40
	MobileNetV3 [36]		✓	68.13	69.50	68.80
				63.80	51.73	57.13

20 tow 10^{-4} and 10^{-5} for subsequent steps. L_2 regularization with a factor of 0 was employed.

B. Experimental Settings

The VinText dataset [10], a recently proposed Vietnamese text dataset, was used to evaluate methods. This benchmark dataset has a total of 56K text instances and 2,000 images, which consists of 1,200 training images, 500 testing images, and 300 unseen test images. Most text instances have usually too many patterns and chaotic scenes with scene text instances of various types, appearances, sizes, and orientations. We evaluated methods on unseen test images.

In order to evaluate the performance and effectiveness of methods, we used Character Accuracy (CA) metric for recognition task, Precision (P), Recall (R), and F-measure ($F1$) metrics for detection task.

C. Vietnamese Scene Text Spotting Benchmark

1) Scene Text Detection: We conducted experiments on the Vintext dataset [10] using various detection methods, including SAST [16], DB++ [15], DB [14], EAST [17], in different settings.

Table I showcases the results with DB++ [15] outperforming other methods. Specifically, DB++ with the ResNet50 [35] backbone achieved the highest performance (80.32% in terms of Hmean) when pre-trained on the ICDAR2015 dataset [28] and fine-tuned on the VinText dataset. On the other hand, SAST [16] with the ResNet50_vd backbone, pre-trained on the ICDAR2015 dataset, demonstrated competitive results across all metrics. However, it is noteworthy that there exists a trade-off between the accuracy and comprehensiveness of these methods, as reflected by relatively lower recall values. In contrast, EAST [17] exhibited a balanced performance in terms of precision and recall. These findings shed light on the strengths and limitations of these methods, facilitating informed decision-making when selecting an appropriate scene text detection method for our ensemble learning framework.

Backbone evaluation: Our investigation focuses on two scene text detection methods, DB [14] and EAST [17], where we explore the influence of different backbone architectures, ResNet50_vd [33] and MobileNetV3 [36]. The comparative results in Table I highlight the superiority of ResNet50_vd over MobileNetV3 in terms of performance. These findings emphasize the critical importance of selecting an appropriate backbone architecture for each model, as it significantly influences the effectiveness of text detection and recognition tasks.

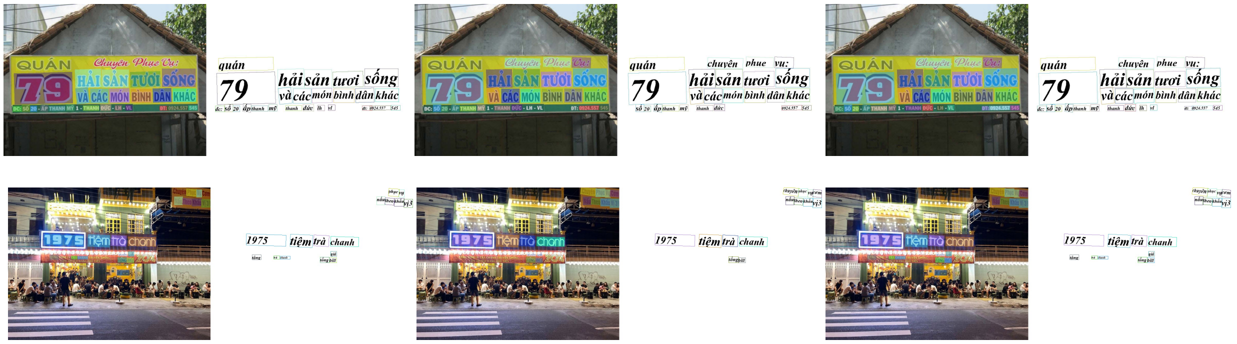
TABLE II: Result of recognition methods on the VinText [10]. 1^{st} and 2^{nd} places are shown in **blue** and **red**, respectively. We remark that methods were pre-trained on MJSynth [29, 30] and SynthText [31] datasets in default.

Method	Backbone	Pre-trained Model	Fine-tuned on VinText [10]	Accuracy
SRN [19]	ResNet50_vd_fpn [33]	✓	✓	70.73
		✓		25.81
			✓	26.01
ABINet [21]	ResNet45 [37]		✓	73.03
		✓		28.57
			✓	36.89
SPIN [18]	ResNet32 [38]	✓	✓	**78.35**
		✓		20.69
RobustScanner [39]	ResNet31 [38]	✓	✓	**80.77**
		✓		28.01
			✓	42.18
SAR [40]	ResNet31 [38]	✓	✓	69.30
		✓		26.10
			✓	49.21
SVTR [20]	SVTR_Tiny [41]	✓	✓	66.04
		✓		28.04
			✓	30.04
NRTR [42]	NRTR_MTB [42]	✓	✓	30.30
		✓		4.44
			✓	28.00
RFL [43]	ResNetRFL [43]	✓	✓	52.18
		✓		26.77
			✓	27.07

Effectiveness of fine-tuning models: Table I shows that the fine-tuned models on the Vintext dataset [10] exhibit a substantial improvement in results compared to their pre-training counterparts. These findings demonstrate the potency of leveraging the Vintext dataset for fine-tuning to significantly enhance the performance of Vietnamese scene text detection models.

2) Scene Text Recognition: We also evaluated the performance of various recognition methods, such as SRN [19], ABINet [21], SPIN [18], RobustScanner [39], SAR [40], SVTR [20], NRTR [42], RFL [43], in different settings. Table II presents a comparative analysis of these methods based on their average accuracy. RobustScanner [39], SPIN [18], ABINet [21], and SRN [19] stand out with impressive results. RobustScanner achieved the highest accuracy of 80.77%, followed by SPIN with 78.35%, ABINet with 73.03%, and SRN with 70.73%. These methods consistently demonstrate strong performance. On the other hand, RFL [43] and NRTR [42] exhibit relatively lower accuracies of 52.18% and 30.30%, respectively.

The experimental results also reveal compelling evidence of the importance of pre-training models on MJSynth [29, 30] and SynthText [31] datasets. Training from scratch on the VinText dataset [10] yields notably inferior performance across all methods. In addition, the results of fine-tuned models exhibit a remarkable improvement, which indicates the necessity of fine-tuning pre-trained models on the VinText dataset [10].



(a) DB++ [15] and SPIN [18]

(b) SAST [16] and ABINet [21]

(c) Our ensemble method

Fig. 3: Visualization of results of our ensemble learning framework.

TABLE III: End-to-end scene text spotting results on the VinText dataset [10].

	Method	Char_acc	F-measure
(1)	DB++ [15] and SPIN [18]	54.08	60.19
(2)	SAST [16] and ABINet [21]	52.59	58.59
(3)	DB [14] and SRN [19]	50.51	56.87
(4)	PGNet (4) [24]	53.73	60.03
	Ensemble learning (Ours):		
	(1), (2)	**58.75**	**65.21**
	(1), (3)	51.09	55.06
	(1), (4)	56.24	64.37
	(1), (2), (3)	49.83	57.33
	(1), (2), (3), (4)	52.91	55.42

D. Ensemble Learning Evaluation

We carefully analyzed the results of detection and recognition methods in Table I and Table II to select appropriate models for our ensemble learning framework. The objective is to identify models that excel in distinct aspects and can synergistically complement each other when combined.

In the evaluation of detection methods from Table I, we meticulously analyze their performance on both the ICDAR2015 [28] and Total-Text datasets [34]. Notably, DB++ [15], EAST [17], and SAST [16] stand out in different metrics. DB++ [15] showcases high precision, while EAST [17] demonstrates good recall, and SAST exhibits a strong F-measure specifically on the Total-Text dataset [34].

Shifting our focus to the recognition methods from Table II, we assess their average accuracy. SPIN [18], ABINet [21], and RobustScanner [39] stand out as top performers, demonstrating remarkable accuracy in text recognition. However, upon merging the RobustScanner [39] with the detection methods, we observe challenges with overlapping text boxes, leading to a decline in overall performance. As a result, we make the decision to exclude the RobustScanner [39] from our final ensemble, striking a balance between detection and recognition performance.

After the individual evaluations, we selected DB++ [15], EAST [17], SAST [16])for the detection task, SPIN [18], ABINet [21], SRN [19] for the recognition task. Additionally, PSNet [24] was chosen as an end-to-end model capable of addressing both detection and recognition aspects. Subsequently, we employed our ensemble algorithm to combine model pairs, harnessing the unique strengths of each end-to-end model. The results of this comprehensive ensemble are presented in Table III, showcasing a powerful and complementary fusion of models for Vietnamese scene text spotting.

While potential of ensemble learning is vast, it is crucial to recognize that mere combination of more models does not always guarantee improved results. In fact, some models, when merged, may yield even worse performance compared to their individual counterparts (see Table III). Our rigorous experimentation has led us to discover that the most optimal ensemble result is achieved by combining pairs of methods DB++ [15] / SPIN [18] and SAST [17]/ ABINet [21]. This carefully curated selection of effective ensemble pairs sheds light on the delicate interplay between methods and underscores the significance of thoughtful model combination for achieving remarkable performance gains in scene text spotting. Indeed, our ensemble framework delivers improved performance in all metrics (See Fig. 3).

V. Conclusion

In this paper, we have presented an ensemble learning framework for Vietnamese scene text spotting in urban environments. The proposed technique demonstrates the effectiveness of combining the strengths of different models. Through meticulous experimentation, we identify models with promising performance, strategically designing an ensemble framework that maximizes individual model strengths. Nonetheless, our method does exhibit certain limitations, notably an increase in computational complexity due to the integration of multiple models and instances where certain model combinations yield suboptimal results. In our future research, we aim to address these challenges by focusing on enhancing spelling accuracy during word recognition and reducing the computational complexity of the ensemble model.

Acknowledgement

This research is supported by research funding from Faculty of Information Technology, University of Science, Vietnam National University - Ho Chi Minh City.

References

[1] M. Huang, Y. Liu, Z. Peng, C. Liu, D. Lin, S. Zhu, N. Yuan, K. Ding, and L. Jin, "Swintextspotter: Scene text spotting via better synergy between text detection and text recognition," *arXiv:2203.10209*, 2022.

[2] Y. Liu, J. Zhang, D. Peng, M. Huang, X. Wang, J. Tang, C. Huang, D. Lin, C. Shen, X. Bai, and L. Jin, "Spts v2: Single-point scene text spotting," *arXiv:2301.01635*, 2023.

[3] J. Wang, C. Liu, L. Jin, G. Tang, J. Zhang, S. Zhang, Q. Wang, Y. Wu, and M. Cai, "Towards robust visual information extraction in real world: New dataset and novel solution," *arXiv:2102.06732*, 2021.

[4] C. Zhang, Y. Tao, K. Du, W. Ding, B. Wang, J. Liu, and W. Wang, "Character-level street view text spotting based on deep multisegmentation network for smarter autonomous driving," *IEEE Transactions on Artificial Intelligence*, vol. 3, no. 2, pp. 297–308, 2022.

[5] X. Rong, B. Li, J. P. Munoz, J. Xiao, A. Arditi, and Y. Tian, "Guided text spotting for assistive blind navigation in unfamiliar indoor environments," in *ISVC*, pp. 11–22, 2016.

[6] H.-C. Wang, C. Finn, L. Paull, M. Kaess, R. Rosenholtz, S. Teller, and J. Leonard, "Bridging text spotting and slam with junction features," in *IROS*, pp. 3701–3708, 2015.

[7] N. T. Pham, V. D. Pham, Q. Nguyen-Van, B. H. Nguyen, D. N. Minh Dang, and S. D. Nguyen, "Vietnamese scene text detection and recognition using deep learning: An empirical study," in *GTSD*, pp. 213–218, 2022.

[8] M.-Q. Ha, V.-H. Phan, B. D. Q. Nguyen, H.-B. Nguyen, T.-H. Do, Q.-D. Pham, and N.-N. Dao, "Intelligent scene text recognition in streaming videos," in *Intelligence of Things: Technologies and Applications*, pp. 356–365, 2022.

[9] C. Tran, K. Nguyen-Trong, C. Pham, D. Tran-Anh, and T. Nguyen-Thi-Tan, "Improving text recognition by combining visual and linguistic features of text," in *SoICT*, pp. 329–335, 2022.

[10] N. Nguyen, T. Nguyen, V. Tran, M.-T. Tran, T. D. Ngo, T. H. Nguyen, and M. Hoai, "Dictionary-guided scene text recognition," in *CVPR*, pp. 7383–7392, 2021.

[11] A. D. Le, H. T. Nguyen, and M. Nakagawa, "End to end recognition system for recognizing offline unconstrained vietnamese handwriting," *arXiv:1905.05381*, 2019.

[12] D. Yu, X. Li, C. Zhang, J. Han, J. Liu, and E. Ding, "Towards accurate scene text recognition with semantic reasoning networks," *arXiv:2003.12294*, 2020.

[13] S. Fang, H. Xie, Y. Wang, Z. Mao, and Y. Zhang, "Read like humans: Autonomous, bidirectional and iterative language modeling for scene text recognition," *arXiv:2103.06495*, 2021.

[14] M. Liao, Z. Wan, C. Yao, K. Chen, and X. Bai, "Real-time scene text detection with differentiable binarization," in *AAAI*, pp. 11474–11481, 2020.

[15] M. Liao, Z. Zou, Z. Wan, C. Yao, and X. Bai, "Real-time scene text detection with differentiable binarization and adaptive scale fusion," *IEEE T-PAMI*, 2022.

[16] P. Wang, C. Zhang, F. Qi, Z. Huang, M. En, J. Han, J. Liu, E. Ding, and G. Shi, "A single-shot arbitrarily-shaped text detector based on context attended multi-task learning," in *ACM MM*, pp. 1277–1285, 2019.

[17] X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, and J. Liang, "East: an efficient and accurate scene text detector," in *CVPR*, pp. 5551–5560, 2017.

[18] Zhang, Chengwei, Xu, Yunlu, Cheng, Zhanzhan, Pu, Shiliang, Niu, Yi, Wu, Fei, Zou, and Futai, "Spin: structure-preserving inner offset network for scene text recognition," in *AAAI*, vol. 35, pp. 3305–3314, 2021.

[19] D. Yu, X. Li, C. Zhang, T. Liu, J. Han, J. Liu, and E. Ding, "Towards accurate scene text recognition with semantic reasoning networks," in *CVPR*, pp. 12113–12122, 2020.

[20] Y. Du, Z. Chen, C. Jia, X. Yin, T. Zheng, C. Li, Y. Du, and Y.-G. Jiang, "Svtr: Scene text recognition with a single visual model," in *IJCAI*, 2022.

[21] S. Fang, H. Xie, Y. Wang, Z. Mao, and Y. Zhang, "Abinet: Read like humans: Autonomous, bidirectional and iterative language modeling for scene text recognition," in *CVPR*, pp. 7098–7107, 2021.

[22] M. Liao, B. Shi, X. Bai, X. Wang, and W. Liu, "Textboxes: A fast text detector with a single deep neural network," in *AAAI*, 2017.

[23] D. Peng, X. Wang, Y. Liu, J. Zhang, M. Huang, S. Lai, S. Zhu, J. Li, D. Lin, C. Shen, X. Bai, and L. Jin, "Spts: Single-point text spotting," *arXiv preprint arXiv:2112.07917*, 2022.

[24] P. Wang, C. Zhang, F. Qi, S. Liu, X. Zhang, P. Lyu, J. Han, J. Liu, E. Ding, and G. Shi, "Pgnet: Real-time arbitrarily-shaped text spotting with point gathering network," *arXiv preprint arXiv:2104.05458*, 2021.

[25] B. E. Rosen, "Ensemble learning using decorrelated neural networks," No. 3-4 in 8, pp. 373–384, Taylor & Francis, 1996.

[26] L. Deng and J. Platt, "Ensemble deep learning for speech recognition," in *Interspeech*, 2014.

[27] Á. Casado-García and J. Heras, "Ensemble methods for object detection," in *ECAI*, pp. 2688–2695, IOS Press, 2020.

[28] D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, S. Ghosh, A. Bagdanov, M. Iwamura, J. Matas, L. Neumann, V. R. Chandrasekhar, S. Lu, F. Shafait, S. Uchida, and E. Valveny, "Icdar 2015 competition on robust reading," in *ICDAR*, pp. 1156–1160, 2015.

[29] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Synthetic data and artificial neural networks for natural scene text recognition," in *Workshop on Deep Learning, NIPS*, 2014.

[30] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Reading text in the wild with convolutional neural networks," No. 1 in 116, pp. 1–20, Springer, jan 2016.

[31] A. Gupta, A. Vedaldi, and A. Zisserman, "Synthetic data for text localisation in natural images," in *CVPR*, 2016.

[32] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv:1412.6980*, 2014.

[33] S. Wang, X. Xia, L. Ye, and B. Yang, "Automatic detection and classification of steel surface defect using deep convolutional neural networks," No. 3 in 11, p. 388, MDPI, 2021.

[34] C. K. Ch'ng and C. S. Chan, "Total-text: A comprehensive dataset for scene text detection and recognition," in *ICDAR*, vol. 1, pp. 935–942, 2017.

[35] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, pp. 770–778, 2016.

[36] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, *et al.*, "Searching for mobilenetv3," in *ICCV*, pp. 1314–1324, 2019.

[37] S. Wu, S. Zhong, and Y. Liu, "Deep residual learning for image steganalysis," *Multimedia tools and applications*, vol. 77, pp. 10437–10453, 2018.

[38] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *arXiv:1512.03385*, 2015.

[39] X. Yue, Z. Kuang, C. Lin, H. Sun, and W. Zhang, "Robustscanner: Dynamically enhancing positional clues for robust text recognition," in *ECCV*, pp. 135–151, 2020.

[40] H. Li, P. Wang, C. Shen, and G. Zhang, "Show, attend and read: A simple and strong baseline for irregular text recognition," in *AAAI*, no. 01 in 33, pp. 8610–8617, 2019.

[41] C. Li, W. Liu, R. Guo, X. Yin, K. Jiang, Y. Du, Y. Du, L. Zhu, B. Lai, X. Hu, *et al.*, "Pp-ocrv3: More attempts for the improvement of ultra lightweight ocr system," *arXiv preprint arXiv:2206.03001*, 2022.

[42] F. Sheng, Z. Chen, and B. Xu, "Nrtr: A no-recurrence sequence-to-sequence model for scene text recognition," in *ICDAR*, pp. 781–786, 2019.

[43] H. Jiang, Y. Xu, Z. Cheng, S. Pu, Y. Niu, W. Ren, F. Wu, and W. Tan, "Reciprocal feature learning via explicit and implicit tasks in scene text recognition," in *ICDAR*, 2021.