Robust Scene Text Detection for Delivery Robots: Augmented Training for Rainy Day

Seunghoon Kang[1], Jonghyun Song[2], Yujin Jeon[2], Myungjoo Lee[2]
[1]Department of Mechanical Engineering, Seoul National University
[2]Graduate School of Data Science, Seoul National University
{alaska97, hyeongoon11, jyj950309 , ckgn0316}@snu.ac.kr

Abstract

Robotic systems that operate in diverse weather conditions require robust scene text recognition (STR) models under adverse weather conditions. Current STR models often exhibit suboptimal performance in real-world scenarios as they rely on ineffective synthetic datasets for training. In this paper, we propose a novel approach to enhance STR model robust specifically for rainy days. Our approach trains De-rain and Scene text detection models with augmented data simulating rainy weather conditions, incorporating manipulated images with raindrops. By introducing raining effects on real-world images, our method enables accurate text recognition in challenging rainy environments. We will conduct experiments to demonstrate its effectiveness in improving STR model performance, specifically in rainy day scenarios.

1. Introduction

For robotic systems to effectively understand and interact with their surroundings, comprehending texts and signs is crucial. Scene Text Recognition (STR) technologies are indispensable in this regard, serving as core components in a variety of modern applications. These applications include autonomous vehicles such as self-driving cars, drones, and unmanned aerial vehicles (UAVs) [7, 14, 20, 24, 26]; robotics, particularly in delivery robots, service robots, and humanoid robots [9, 12, 21]; as well as augmented reality (AR) and virtual reality (VR) smart assistants. [10, 17]

Even though current Scene Text Recognition (STR) models [15] show significant improvement, they often fall short when deployed in real-world scenarios, particularly in adverse weather conditions such as rain and snow. Raindrops and snow can distort the image, posing challenges for STR models in reading the text correctly.

One of the main reasons why STR models underperform in adverse weather scenarios is that the training data primarily consists of synthetic datasets, which do not reflect the characteristics of adverse weather conditions. [1] While these datasets offer scalability and diversity, they often fail to capture the complexity and variability present in real-world environments. As a result, STR models trained solely on synthetic data may struggle to generalize to unseen real-world data, leading to decreased performance and reliability.

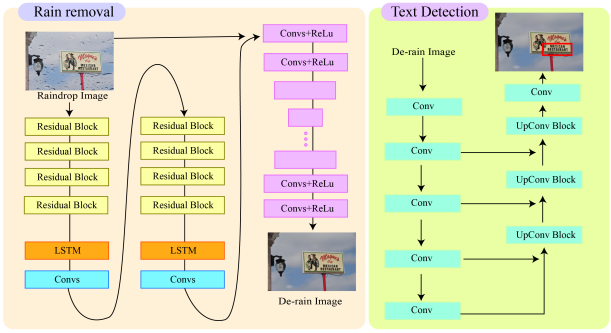


Figure 1. The architecture of our proposed work flow using Rain removal model and Scene text detection model

To address the inherent shortcomings of the training dataset, we propose a novel approach to enhance STR model's robustness by incorporating augmented data into the training process. By introducing raining effects on manipulated real-world images, we aim to simulate various weather conditions encountered by robots during their operations. This augmentation strategy enables the model to learn robust features that are invariant to changes in weather, thereby improving its performance and generalization capabilities.

Our expected contributions from this project are as follows:

- We propose novel data augmentation methods to simulate various weather conditions, vividly representing real-world scenarios.

- We will demonstrate that conventional scene text recognition (STR) models underperform in adverse weather conditions and that our image augmentation methods significantly enhance model performance.

- We propose a robust architecture for processing images with raindrops. Specifically, our model consists of a raindrop network that denoises the rain traces and a text detection network that detects the coordinates of the text.

2. Related Work

2.1. Optical Character Recognition

Optical Character Recognition (OCR) systems extract text from images, relying on traditional methods for pattern recognition and feature extraction. While effective in controlled environments, these techniques struggle with complex scenes due to varying lighting, occlusions, and text orientations. Deep learning advancements have significantly improved OCR accuracy, particularly with large datasets. Despite these improvements, challenges persist in adapting OCR systems to diverse real-world scenarios with varying backgrounds and contexts.

2.2. Scene Text Recognition

Scene Text Recognition (STR) is a subset of OCR focused on identifying text within natural scenes like signs, storefronts, and labels. Unlike traditional OCR, which works in controlled environments, STR deals with cluttered backgrounds, varying fonts, sizes, orientations, and lighting. Early methods used handcrafted features and heuristic algorithms but struggled with generalization. Deep learning introduced end-to-end trainable models, enabling direct learning from raw image data, revolutionizing STR research.

For example, MGP-STR [23] is built upon the original Vision Transformer (ViT) [5] model by adding a special module called Adaptive Addressing and Aggregation (A^3). This module selects important token combinations from ViT and combines them into one output token for each character. It also uses different modules for predicting subwords, which helps understand language better. Finally, all these predictions are merged together using a simple method.

PARSeq [3] utilizes an ensemble of internally linked autoregressive(AR) language models(LM)s with shared weights using Permutation Language Modeling, thereby integrating context-free non-AR and context-aware AR inference. Through training on synthetic data and leveraging attention mechanisms, it ensures robustness in handling arbitrarily-oriented text commonly encountered in real-world images.

CLIP4STR [30] incorporates the permuted sequence modeling technique proposed by PARSeq, and additionally introduces a novel STR method built upon image and text encoders of CLIP [19]. The model incorporates two branches, one for visuals and one for cross-modal refinement, utilizing a dual predict-and-refine decoding strategy.

Despite significant progress, STR models trained on synthetic datasets often face challenges when deployed in real-world settings due to domain gaps and the inability to generalize to diverse environmental conditions. CRAFT [2] identifies text regions by analyzing the area and connection between characters. It excels in pinpointing text areas on a character-by-character basis, making it highly effective for recognizing text in diverse shapes and orientations. DBNet [11] proposes the approach designed to improve both the precision and efficiency of text detection within intricate real-world settings using Differentiable binarization and adaptive scale fusion. DOTS [13] presents the methods to handle oriented text by integrating a unique RoIRotate module, enabling it to recognize rotated texts directly.

2.3. Data Augmentation for Raindrop

Data augmentation techniques play a crucial role in enhancing the robustness and generalization capabilities of OCR and STR models. By artificially increasing the diversity of training data through various transformations, augmentation methods aim to mitigate overfitting and improve model performance on unseen data. Common augmentation strategies include geometric transformations (such as rotation, scaling, and perspective transformation), color augmentation (adjusting brightness, contrast, and saturation), and noise injection (adding Gaussian noise, blurring, or dropout) [16, 22]. Previous attempts have been made to apply weather conditions such as rain or snow as a part of data augmentation techniques [1]. However, Conventional techniques for simulating weather conditions such as rain or snow as a part of data augmentation have often produced unrealistic images. In addition, novel approaches have emerged recently that a dataset consists of manipulated images using professional image editing software such as Adobe Photoshop [25]. However, there have been no previous attempts to apply weather conditions such as rain as part of data augmentation using the software. In this study, we aim to fill this gap by exploring the feasibility of applying raindrop conditions using this image editing program. We anticipate that our approach will produce more realistic images, enhancing data augmentation effectiveness in scene text recognition.

2.4. Raindrop Removal Methods

Numerous methods have been proposed to address the challenge of removing raindrops from single images, primarily leveraging convolutional neural networks (CNNs).

Eigen et al. [6] introduced a CNN-based approach trained on paired raindrop-degraded and raindrop-free images. While effective for sparse and small raindrops, this method struggles with larger and denser raindrops, often resulting in blurred outputs due to network limitations.
AttentiveGAN [18] employs an adversarial training strategy coupled with visual attention mechanisms for raindrop removal. Despite its success, the presence of inexplicable noise points in output images remains a limitation. Expanding upon these advancements, Haiying Xia introduced a novel two-step generative adversarial network (GAN) approach for raindrop removal [27]. By incorporating hierarchical supervision and attention mechanisms, this method addresses the deficiencies of previous techniques and achieves superior performance in raindrop removal tasks.

Notably, prior research primarily focused on raindrop removal without considering the presence of text in images. In this study, we aim to bridge this gap by creating a synthesis dataset containing text, leveraging existing scene text recognition (STR) datasets such as ICDAR2015 [8] and TotalText [4]. Our new synthesis dataset will be utilized to train and evaluate both raindrop removal and STR models, providing insights into their performance in real-world scenarios.

3. Method

3.1. Model Architecture

Our text detection pipeline consists of two modules: rain removal and text detection. For the rain removal module, we adopt the pre-trained AttentiveGAN model [18], which effectively removes raindrops from images. For the text detection module, we utilize either the CRAFT [2] or DB-Net [11] models. When an image is taken as the input, the raindrops are first removed using the rain removal module. The cleaned images are then processed through the text detection module to detect the text. (Figure 1)

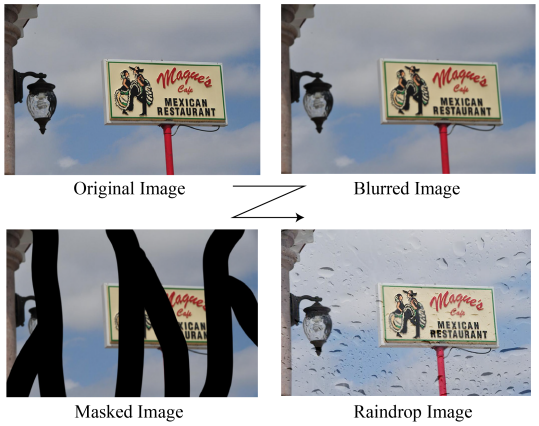


Figure 2. Generating manipulated raindrop image

3.2. Optimization

For the rain removal module $\mathcal{M}_{rem}$ and the text detection module $\mathcal{M}_{det}$, a raindrop-affected image $\mathbf{x}$ is processed by $\mathcal{M}_{rem}$ to produce a de-rained image $\hat{\mathbf{y}}$. This de-rained image is then processed by $\mathcal{M}_{det}$ to obtain $\hat{\mathbf{z}}$, the coordinates for the text in the image. In this process, loss function for de-rain net is defined as a generative adversarial loss:

$$\mathcal{L}_{rem} = \min_{\mathcal{M}_{rem}} \max_{D} \mathbf{E}_{\mathbf{R}\sim p_{clean}} log(D(\mathbf{R})) + \mathbf{E}_{\mathbf{x}\sim p_{raindrop}} log(1 - D(\mathcal{M}_{rem}(\mathbf{x}))) \quad (1)$$

where D represents the discriminative network for adversarial training and $\mathbf{R}$ is a sample from a pool of clean natural images. The loss function for the text detection module is defined using two scores: the affinity score and the region score. The affinity score represents the probability that a pixel is in the center of the space between adjacent characters, while the region score indicates the probability that a given pixel is the center of a character. The loss function for the text detection module is calculated as the sum of two Mean Squared Error (MSE) losses—one for the region score and one for the affinity score.

$$\mathcal{L}_{det} = \sum_{p} (||S_r(p) - S_r^*(p)||_2^2 + ||S_a(p) - S_a^*(p)||_2^2) \quad (2)$$

where p denotes the pixel in the bounding box region, and $S_r(p)$ and $S_a(p)$ is the predcited region score and affinity score.

As our model consists of two separate modules, how to train the entire model using two losses is a design choice. We conduct experiments for two training strategies: training rain removal and text detection *separately* and *simultaneously*.

For *separate* training, we first train the de-rain module independently, without supervision from the text detection module. Since the de-rain module is pre-trained on images without text, we further fine-tune it using rainy images that contain text. Concurrently, we train the text detection module without any assistance from the de-rain module. This approach enhances the robustness of the text detection module under adverse conditions, ensuring it performs well even without prior rain removal.

The only difference in the *simultaneous* training approach is that the de-rain module receives training signals from both $\mathcal{L}_{rem}$ and $\mathcal{L}_{det}$. In other words, the objective function for training the de-rain module is $\mathcal{L} = \mathcal{L}_{rem} + \mathcal{L}_{det}$, while the text detection module maintains the same loss function as in the separate training approach.

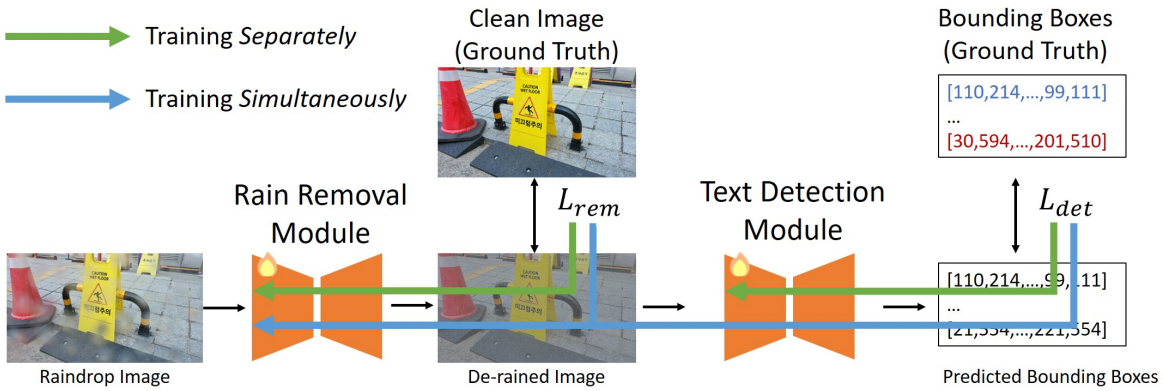


Figure 3. Optimization strategy for our framework. In training *separately*, the rain removal module and text detection module are trained separately. In training *simultaneously*, training signal from text detection loss is back-propagated to the rain removal module.

3.3. Data Augmentation

In this section, the process for creating a new dataset of manipulated images is detailed, as depicted in Figure 2. The process begins with datasets such as ICDAR2015, Totaltext, and CTW 1500. Gaussian blur with a standard deviation of 0.6 is applied to simulate raindrops on the images. Next, the effect of rain lines is applied by adding clipping masks with arbitrary vertical lines, resembling rain flowing down a wet window. Lastly, random wet window images are synthesized to diversify the dataset. This approach enables the generation of a sizable and diverse collection of manipulated images, crucial for training robust scene text detection models. Moving forward, efforts will be made to further refine this dataset and explore additional augmentation techniques to enhance its effectiveness in STR tasks.

4. Experiments

4.1. Datasets

As mentioned in Section 2.3, we utilized Adobe Photoshop to overlay raindrop images on the dataset required for model training. Specifically, the datasets used are as follows:

4.1.1 Datasets for Text Detection

ICDAR 2015 (IC15) [8] is the fourth challenge of the ICDAR 2015 Robust Reading Competition, serving as a popular benchmark for oriented scene text detection and recognition tasks. The dataset features 1,000 training images and 500 testing images, all captured through Google Glass without specific attention to the positioning of the device. Consequently, the texts within these images may appear in various orientations. The dataset primarily contains English text, with annotations provided at the word level utilizing quadrilateral boxes.

CTW 1500 [28] is a recently introduced dataset designed for curved text detection. It comprises 1000 images for training and 500 for testing. The dataset specifically emphasizes curved text instances, each labeled with a 14-polygon annotation.

TotalText [4] contains 1,555 images exhibiting various text styles, such as horizontal, multi-angled, and curved text examples. It aims to tackle the insufficient diversity in text alignment observed in current scene text datasets by highlighting three distinct text alignments: horizontal, multi-angled, and curved. This dataset represents a pioneering effort in presenting a broad spectrum of text alignments on a relatively large scale.

4.1.2 Datasets for Rain Removal

Raindrop dataset [18] contains 861 pairs of images for the training set of raindrop images. The test set is divided into two subsets: TestA and TestB. TestA consists of 58 pairs, which are a subset of TestB containing 249 pairs. Notably, this dataset is not generated through data synthesis. Instead, it involves the use of two identical glasses—one sprayed with water and the other kept clean—followed by image acquisition using the Sony A6000 and Canon EOS 60 cameras.

MLVU Raindrop dataset comprises 150 image pairs for the training set, showcasing raindrop images, and 76 pairs for the test set. Remarkably, this dataset is not synthesized; rather, it is created by employing two identical glasses—one coated with water and the other left pristine—capturing images with Sony Samsung SM-F926N and Canon EOS 200D cameras. Our image dataset not only addresses raindrop removal but also incorporates a crucial element: text. Unlike other datasets focused solely on removing raindrops from images, our dataset includes text and provides quadrilateral-type bounding box annotation, making it invaluable for

	IC15 (ICDAR 2015)					
	w/o raindrops			w/ raindrops		
	Recall	Precision	H-mean	Recall	Precision	H-mean
CRAFT	86.4	90.5	88.4	84.2 (-2.2)	89.3 (-1.2)	86.7 (-1.7)
DBNet-18	56.8	87.9	69	55.8 (-1.0)	83.0 (-4.9)	66.7 (-1.3)
FOTS	75.9	79.8	77.8	71.2 (-4.7)	80.1 (+0.3)	75.4 (-2.4)

Table 1. Performance of baselines on images without and with raindrops. The number inside the parenthesis is the performance gap between the two settings.

Dataset	Raindrop		IC15 w/rain		MLVU raindrop	
Method	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
AttentiveGAN [18]	31.57	0.91	**15.82**	0.82	**19.68**	**0.56**

Table 2. PSNR and SSIM generalization comparisons on Raindrop dataset and our dataset

Dataset	Method	Recall	Precision	H-mean
CTW1500 w/o rain		84.8	90.3	87.5
CTW1500 w/ rain	base	80.6	91.5	85.7
	derain	**80.8**	91.9	**86.0**
	fine-tuned derain	77.8	**92.5**	84.5
Total-text w/o rain		88.2	92.9	90.5
Total-text w/ rain	base	**85.6**	92.1	**88.7**
	derain	80.7	91.1	85.6
	fine-tuned derain	81.5	**92.7**	86.7
ICDAR 2015 w/o rain		86.4	73.1	75
ICDAR 2015 w/ rain	base	**84.2**	89.3	**86.7**
	derain	72.8	89.0	80.1
	fine-tuned derain	78.6	**92.1**	84.8
MLVU w/o rain		64.2	33.8	44.3
MLVU w/ rain	base	40.8	34.6	37.4
	derain	**53.2**	30.1	38.4
	fine-tuned derain	37.0	**40.0**	**38.5**

Table 3. OCR performance of baseline models on multiple datasets each with and without raindrops. Raindrop-applied images are directly used as the input(base), raindrops are removed with pretrained de-raining model(derain), and raindrops are removed with de-raining model fine-tuned for each dataset(fine-tuned derain). For the CTW1500 and Total-Text datasets, we employ MixNet for inference. For other datasets, we utilize CRAFT for text detection during inference.

tasks such as text detection. This unique feature sets our dataset apart and underscores its significance for advancing research in this domain.

4.2. Training Details and Evaluation Setup

We experimented with MixNet and CRAFT across a variety of datasets. All models were initialized from pretrained versions on datasets without raindrop effects. Subsequently, the models were further fine-tuned on raindrop-affected datasets. All models were implemented in PyTorch and optimized using the Adam optimizer. For evaluation metrics, we used recall, precision, and harmonic mean (H-mean) following prior works.

4.3. Experimental Results

Model Adaptation Results To demonstrate the impact of the raindrop effect on scene text detection, we conduct the preliminary experiment which compares the performance of major baselines [2, 11, 13] on images with and without raindrops.

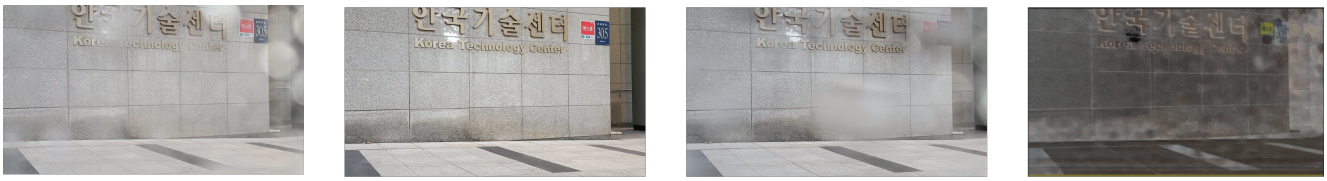
In Table 1, we illustrate the performance gap observed when the same model, which has not been fine-tuned on images with raindrops, evaluates the ICDAR '15 images both with and without raindrops. Each model exhibits a performance decline ranging from 1.3 to 2.4 points for images with raindrops. This indicates that state-of-the-art (SOTA) scene text recognition (STR) models may lack robustness to raindrop-affected images, highlighting the importance of dataset augmentation techniques for these conditions.

For pretrained Attentive GAN model [18] with a limited amount of raindrop data, we observe moderate performance in terms of PSNR and SSIM on the Raindrop dataset. However, when applying raindrop effects, which we specifically manipulated, the model demonstrated significantly reduced PSNR score (reaching 19.68) and SSIM score (reaching 0.56) (Table 2). Also the result of the image from the pretrained model has color illusion and character distortion or MLVU raindrop data. This stark difference suggests that the Attentive GAN model could benefit from a more extensive and diverse training dataset, such as the one we have developed. Therefore, future iterations of the model should focus on incorporating larger datasets, including our dataset, to further enhance its performance and generalization capabilities.

We conduct experiments with two state-of-the-art baselines using different training approaches and optimization strategies to validate our claim. Across all datasets and models, we find the tendency that inference with pre-trained or fine-tuned derain module enhances the performance.

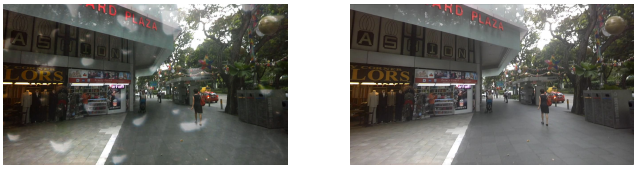
MixNet MixNet [29] is a hybrid architecture combining CNNs and Transformers to accurately detect small scene text instances under challenging conditions, such as irregular positions and nonideal lighting. Using MixNet, we conducted text detection predictions on two datasets, CTW1500 [28] and Total-text [4]. As illustrated in Table 3, the model demonstrates strong performance on the original datasets without raindrops. However, its performance significantly deteriorates when raindrops are added to the images, supporting our hypothesis. When the raindrops are removed using a pre-trained DeRain [27] model, there is a slight improvement in performance, though it still falls short compared to the original dataset. Notably, retraining the DeRain model with our specific dataset leads to a performance gain, highlighting the importance of appropriate preprocessing and training on rain-affected datasets for optimal text detection results in real-world scenarios.

CRAFT CRAFT [2] is a U-Net architecture proposed for multi-lingual text detection. As CRAFT is pre-trained on



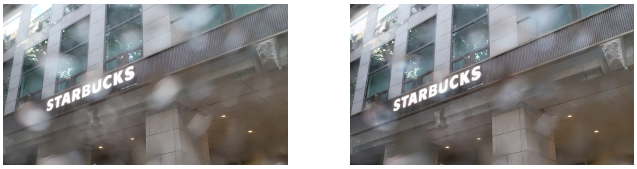
(a) Raindrop Image (b) Clean Image (c) De-rained Image from Training *Separately* (d) De-rained Image from Training *Simultaneously*

Figure 4. Comparison of de-rained images from training separately (c) and training simultaneously



(a) rain-photoshop (b) derained

Figure 5. Images from ICDAR2015 with synthesized raindrops and raindrops removed with DeRain



(a) rain-real (b) derained

Figure 6. Images from our dataset with real raindrops and raindrops removed with DeRain

the ICDAR 2015 dataset, we conduct fine-tuning for the ICDAR 2015 rain datasets and perform out-of-distribution inference on the MLVU dataset. In Table 3, the model demonstrates robust performance when there is no influence of raindrops.

For the ICDAR 2015 dataset without rain, the baseline model achieves H-mean of 75. When de-rain module is presented, the performance of the baseline model shows a comparable result H-mean of 84.8. For the out-of-distribution MLVU dataset, the baseline model performs poorly with H-mean of 37.4. Incorporating the pre-trained de-rain module improves the performance to a H-mean of 38.5.

The Effect of Training *Simultaneously* In Table 4, we only presented the results with the training *Separately*. In this section, we will discuss the effect of the training *Simultaneously*. When trained *simultaneously*, the performance dropped across all datasets. Additionally, the de-rained images indicate that supervision from the text detection loss, i.e., training simultaneously, degrades the de-raining performance (Fig 4).

Unlike the de-rained images from separate training, which effectively removes raindrops while maintaining the overall color of the entire image (Fig 4.c), simultaneous training results in distorted image colors. This implies that the text detection loss adversely affects the de-raining capability when both modules are trained *simultaneously* (Fig 4.d).

Dataset	Training	Recall	Precision	H-mean
ICDAR 2015 w/ rain	Separately	78.6	92.1	84.8
	Simultaneously	63.5	86	63.5
MLVU w/ rain	Separately	37	40	38.5
	Simultaneously	36.6	35.4	35.9

Table 4. The effect of training CRAFT *separately* and *simultaneously* for ICDAR 2015 and MLVU dataset

Overall, these results highlight the importance of the de-rain module, particularly for out-of-distribution datasets like MLVU raindrop. When raindrop effects are present, the baseline model without the de-rain module does not show robust performance, especially in out-of-distribution scenarios. Fine-tuning the de-rain module for specific datasets significantly improves the text detection performance under adverse weather conditions.

4.4. Discussion

Our current work focuses on the individual training and evaluation of models for raindrop removal and scene text detection. However, there are several ways for future research to enhance the robustness and applicability of this approach.

Challenges in Real-World Application Despite our efforts to train the model using Photoshop-generated raindrop images, we found that applying the model to real-world data proved to be challenging. The synthetic data, while useful for controlled training, did not entirely prepare the model for the variability and complexity of actual raindrop patterns on real-world images. This honest acknowledgment of our model's current limitations highlights the need for more realistic and diverse datasets.

Particularly in the case of the derain model, we observed a significant drop in performance when experimenting with real-world images. As shown in Fig 5 and Fig 6, while the derain process was successfully carried out with the syn-

thesized raindrop image, the real-world image did not experience the same level of success. This indicates that our current model is not yet fully capable of handling the complexities found in actual raindrop patterns.

Future work will involve collecting and utilizing real-world data to bridge this gap and improve the model's performance in practical applications. This will ensure that our models are not only effective in controlled environments but also robust and reliable in real-world scenarios.

Generalization to Different Weather Conditions Another limitation of our approach is the current focus on raindrop effects. While this is a significant challenge, other adverse weather conditions such as snow, fog, and heavy rain can also severely impact the performance of scene text recognition systems. The robustness of the model under varying weather conditions has not yet been thoroughly tested. Future research should aim to extend the current approach to handle a broader range of weather conditions. This could involve creating and utilizing diverse weather-specific datasets and developing adaptive algorithms that can generalize well across different environmental challenges.

5. Conclusion

In this paper, we addressed the critical issue of Scene Text Recognition (STR) performance degradation under adverse weather conditions, particularly focusing on rainy days. Our proposed approach involves the use of novel data augmentation techniques to simulate various weather conditions, specifically by manipulating real-world images to include raining effects. This methodology aims to create a more robust training dataset that better represents the challenges faced in real-world scenarios.

Through our experiments, we demonstrated that conventional STR models exhibit significant performance drops when exposed to adverse weather conditions due to their reliance on synthetic datasets that fail to capture real-world complexities. Our enhanced training process, incorporating augmented data with realistic raining effects, significantly improves the robustness and generalization capabilities of STR models.

In conclusion, our approach of augmenting training data with realistic weather conditions and integrating a rain removal model provides a promising solution to improve the robustness of scene text detection systems. This advancement holds significant potential for various applications in robotics, autonomous vehicles, and augmented reality, ensuring reliable text recognition even in challenging environments.

References

[1] Rowel Atienza. Data augmentation for scene text recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1561–1570, 2021. 1, 2

[2] Youngmin Baek, Bado Lee, Dongyoon Han, Sangdoo Yun, and Hwalsuk Lee. Character region awareness for text detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9365–9374, 2019. 2, 3, 5

[3] Darwin Bautista and Rowel Atienza. Scene text recognition with permuted autoregressive sequence models. In *European conference on computer vision*, pages 178–196. Springer, 2022. 2

[4] Chee Kheng Ch'ng, Chee Seng Chan, and Chenglin Liu. Total-text: Towards orientation robustness in scene text detection. *International Journal on Document Analysis and Recognition (IJDAR)*, 23:31–52, 2020. 3, 4, 5

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 2

[6] David Eigen, Dilip Krishnan, and Rob Fergus. Restoring an image taken through a window covered with dirt or rain. In *Proceedings of the IEEE international conference on computer vision*, pages 633–640, 2013. 3

[7] Abhishek Gupta, Alagan Anpalagan, Ling Guan, and Ahmed Shaharyar Khwaja. Deep learning for object detection and scene perception in self-driving cars: Survey, challenges, and open issues. *Array*, 10:100057, 2021. 1

[8] Dimosthenis Karatzas, Lluis Gomez-Bigorda, Anguelos Nicolaou, Suman Ghosh, Andrew Bagdanov, Masakazu Iwamura, Jiri Matas, Lukas Neumann, Vijay Ramaseshan Chandrasekhar, Shijian Lu, et al. Icdar 2015 competition on robust reading. In *2015 13th international conference on document analysis and recognition (ICDAR)*, pages 1156–1160. IEEE, 2015. 3, 4

[9] Obadiah Lam, Feras Dayoub, Ruth Schulz, and Peter Corke. Text recognition approaches for indoor robotics: a comparison. In *Proceedings of the 16th Australasian Conference on Robotics and Automation 2014*, pages 1–7. Australian Robotics and Automation Association (ARAA), 2014. 1

[10] Minghui Liao, Boyu Song, Shangbang Long, Minghang He, Cong Yao, and Xiang Bai. Synthtext3d: synthesizing scene text images from 3d virtual worlds. *Science China Information Sciences*, 63:1–14, 2020. 1

[11] Minghui Liao, Zhisheng Zou, Zhaoyi Wan, Cong Yao, and Xiang Bai. Real-time scene text detection with differentiable binarization and adaptive scale fusion. *IEEE transactions on pattern analysis and machine intelligence*, 45(1):919–931, 2022. 2, 3, 5

[12] Shuhua Liu, Huixin Xu, Qi Li, Fei Zhang, and Kun Hou. A robot object recognition method based on scene text reading in home environments. *Sensors*, 21(5):1919, 2021. 1

[13] Xuebo Liu, Ding Liang, Shi Yan, Dagui Chen, Yu Qiao, and Junjie Yan. Fots: Fast oriented text spotting with a uni-

fied network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5676–5685, 2018. 2, 5

[14] Yang Liu, Zhaowen Wang, Hailin Jin, and Ian Wassell. Synthetically supervised feature learning for scene text recognition. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 435–451, 2018. 1

[15] Canjie Luo, Lianwen Jin, and Zenghui Sun. Moran: A multi-object rectified attention network for scene text recognition. *Pattern Recognition*, 90:109–118, 2019. 1

[16] Francisco J Moreno-Barea, Fiammetta Strazzera, José M Jerez, Daniel Urda, and Leonardo Franco. Forward noise adjustment scheme for data augmentation. In *2018 IEEE symposium series on computational intelligence (SSCI)*, pages 728–734. IEEE, 2018. 2

[17] Imene Ouali, Mohamed Ben Halima, and WALI Ali. Augmented reality for scene text recognition, visualization and reading to assist visually impaired people. *Procedia Computer Science*, 207:158–167, 2022. 1

[18] Rui Qian, Robby T Tan, Wenhan Yang, Jiajun Su, and Jiaying Liu. Attentive generative adversarial network for raindrop removal from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2482–2491, 2018. 3, 4, 5

[19] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2

[20] Sangeeth Reddy, Minesh Mathew, Lluis Gomez, Marçal Rusinol, Dimosthenis Karatzas, and CV Jawahar. Roadtext-1k: Text detection & recognition dataset for driving videos. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11074–11080. IEEE, 2020. 1

[21] José Antonio Álvarez Ruiz, Paul Plöger, and Gerhard K Kraetzschmar. Active scene text recognition for a domestic service robot. In *RoboCup 2012: Robot Soccer World Cup XVI 16*, pages 249–260. Springer, 2013. 1

[22] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48, 2019. 2

[23] Peng Wang, Cheng Da, and Cong Yao. Multi-granularity prediction for scene text recognition. In *European Conference on Computer Vision*, pages 339–355. Springer, 2022. 2

[24] Shu Wang, Dianwei Wang, Pengfei Han, Xincheng Ren, and ZHIJIE XU. Text recognition in uav aerial images. In *Proceedings of the 2021 4th International Conference on Artificial Intelligence and Pattern Recognition*, AIPR '21, page 232–238, New York, NY, USA, 2022. Association for Computing Machinery. 1

[25] Sheng-Yu Wang, Oliver Wang, Andrew Owens, Richard Zhang, and Alexei A Efros. Detecting photoshopped faces by scripting photoshop. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10072–10081, 2019. 2

[26] Qingtian Wu, Yimin Zhou, and Guoyuan Liang. A text detection and recognition system based on an end-to-end trainable framework from uav imagery. In *2018 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 736–741. IEEE, 2018. 1

[27] Haiying Xia, Yang Lan, Shuxiang Song, and Haisheng Li. Raindrop removal from a single image using a two-step generative adversarial network. *Signal, Image and Video Processing*, 16(3):677–684, 2022. 3, 5

[28] Liu Yuliang, Jin Lianwen, Zhang Shuaitao, and Zhang Sheng. Detecting curve text in the wild: New dataset and new solution. *arXiv preprint arXiv:1712.02170*, 2017. 4, 5

[29] Yu-Xiang Zeng, Jun-Wei Hsieh, Xin Li, and Ming-Ching Chang. Mixnet: toward accurate detection of challenging scene text in the wild. *arXiv preprint arXiv:2308.12817*, 2023. 5

[30] Shuai Zhao, Xiaohan Wang, Linchao Zhu, and Yi Yang. Clip4str: A simple baseline for scene text recognition with pre-trained vision-language model. *arXiv preprint arXiv:2305.14014*, 2023. 2