IJRAR.ORG E-ISSN: 2348-1269, P-ISSN: 2349-5138

INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS (IJRAR) | IJRAR.ORG

An International Open Access, Peer-reviewed, Refereed Journal

TWO-STAGE GEOMETRIC RECTIFICATION WITH ATTENTION FOR SCENE TEXT RECOGNITION

[1]Sri.B.Satish Babu [2]Ch. Leela Sai Manohar [2]J. Deepika [2]D.Sai keerthi [2]B.Deepa

[1]Assistant Professor,[2]Student
[12]Department of Information Technology
[1]RVR & JC College of Engineering, Guntur, India

Abstract: Recognizing text in natural scene images remains a challenging task due to irregular distortions such as curvature, rotation, and perspective transformations. To address these challenges, we propose a Two-Level Rectification Network (TRAN) that performs progressive correction prior to recognition. The framework integrates a Geometry-Level Rectification Network (GEO) based on Thin-Plate Spline (TPS) transformation to correct global geometric distortions, followed by a Pixel-Level Rectification Network (PIX) that refines residual local misalignment using dense offset prediction. To further enhance feature representation, we introduce a Channel-Kernel Attention Unit (CKUnit) that adaptively reweights feature channels and convolutional kernels, enabling dynamic receptive field adjustment. The rectified features are then processed through a ResNet-based feature extractor and a Bidirectional LSTM (BiLSTM) sequence modeling module, followed by Connectionist Temporal Classification (CTC) decoding for final text recognition. The entire architecture is implemented in PyTorch based on the ClovaAI deep-text-recognition-benchmark framework and initialized with pretrained TPS-ResNet-BiLSTM weights for stable convergence. Experimental evaluations on standard benchmarks including IIIT5K, SVT, ICDAR2013, ICDAR2015, SVTP, and CUTE80 demonstrate that the proposed two-stage rectification strategy achieves improved robustness and recognition accuracy compared to single-stage rectification approaches.

Index Terms—Scene Text Recognition, Two-Stage Rectification, Thin-Plate Spline, Pixel-Level Refinement, Channel Attention, CTC, Deep Learning.

I.INTRODUCTION

Text is one of the most significant information carriers in everyday human communication. Today, text appears ubiquitously in everyday environments such as billboards, electronic displays, storefronts, and transportation systems. Many modern applications depend on automatic text recognition from images captured in uncontrolled environments, including augmented reality, mobile translation, assistive systems for visually impaired individuals, and autonomous driving systems [1], [2]. Although traditional Optical Character Recognition (OCR) techniques perform effectively on scanned documents, they struggle considerably when applied to scene text images. Unlike structured document text, scene text often exhibits diverse font styles, arbitrary orientations, occlusions, perspective distortions, low contrast, uneven illumination, and curved layouts [3], [4]. These uncontrolled variations significantly degrade recognition accuracy and demand models that are more robust and flexible than conventional OCR systems.

Over the past decade, deep learning has fundamentally transformed Scene Text Recognition (STR). Early deep neural models such as the Convolutional Recurrent Neural Network (CRNN) introduced an end-to-end trainable framework that combined convolutional feature extraction with recurrent sequence modeling for direct image-to-text translation [5]. The introduction of the Spatial Transformer Network (STN) enabled models to learn geometric transformations that rectify distorted inputs before recognition [6]. Building upon this idea, more advanced frameworks such as ASTER incorporated flexible rectification modules to better handle irregular text layouts [7]. Attention-based recognition approaches further improved performance by aligning feature sequences with character predictions [8], [9].

Despite these advancements, geometry-level rectification methods such as STN and Thin-Plate Spline (TPS) transformations often focus on global transformations and may fail to address fine-grained character-level distortions [6], [10]. Highly curved, twisted, or partially occluded text remains challenging under global alignment strategies. On the other hand, pixel-level rectification approaches, such as MORAN,

estimate dense offset maps to refine local distortions, enabling more detailed correction [11]. However, purely pixel-level methods may introduce noise or instability when processing severely distorted inputs and often lack sufficient global structural constraints. Since global and local rectification strategies provide complementary advantages, recent research trends emphasize multi-level rectification mechanisms that integrate both approaches [12], [13].

Motivated by these observations, we implement a Two-Level Rectification Attention Network (TRAN), which integrates geometry-level and pixel-level corrections within a unified, end-to-end trainable framework. Our implementation is built upon the ClovaAI deep-text-recognition-benchmark, a modular PyTorch-based architecture widely adopted for scene text recognition research. In the proposed framework, a Geometry-Level Rectification Network (GEO) first performs coarse alignment using a Thin-Plate Spline transformation to correct large-scale perspective distortions and curvature [10]. Subsequently, a Pixel-Level Rectification Network (PIX) refines the rectified output by predicting dense coordinate offsets for fine-grained correction, similar in spirit to pixel-wise adjustment strategies [11].

To enhance feature representation capability, we introduce a Channel-Kernel Attention Unit (CKUnit), inspired by channel attention mechanisms such as Squeeze-and-Excitation networks [14]. This module adaptively reweights channel activations and convolutional responses, enabling dynamic receptive field adjustment based on text scale, distortion level, and background complexity. The refined features are then processed by an Attention-Based Recognition Network (ABRN), consisting of a ResNet-based feature extractor [15], a bidirectional LSTM sequence model, and a spatial attention decoder. This architecture transforms rectified visual features into character sequences in an end-to-end manner [8].

Training deep multi-stage architectures with multiple rectification modules can be unstable. To address this issue, we adopt a skip-training strategy in which rectification modules are first optimized independently before joint fine-tuning of the entire network. This progressive optimization strategy stabilizes convergence and improves final recognition performance, particularly on irregular text benchmarks [13].

The main contributions of this work are summarized as follows:

We implement a two-level rectification framework that integrates coarse geometry correction and fine pixel-level refinement for scene text recognition without requiring explicit geometric annotations [6], [11].

We introduce a Channel-Kernel Attention mechanism that adaptively models channel dependencies and receptive fields to improve feature representation robustness [14].

By extending and generalizing the ClovaAI deep-text-recognition-benchmark framework, we provide a reproducible and modular pipeline for two-stage rectification and recognition.

Extensive experiments on public benchmarks demonstrate that the proposed two-stage rectification approach consistently outperforms single-stage methods, particularly in challenging irregular text scenarios involving curved, distorted, and perspective-transformed text [7], [12], [13].

II. RELATED WORK

The widespread use of text understanding in natural settings has attracted significant research interest in scene text recognition (STR) [12], [13]. However, typical recognition systems are still challenged by the complexity of scene text, which results from distortion, curvature, and occlusion by backdrop clutter [9], [10].

2.1 Initial STR Methods

The paradigms for sequence modeling were crucial to the early advances in STR. Connectionist Temporal Classification (CTC) enabled sequence prediction without character-level segmentation [16]. Convolution-based feature learning and recurrent sequence transduction were integrated in the Convolutional Recurrent Neural Network (CRNN) to enable end-to-end text recognition without the need for explicit character splitting [5]. When clear, horizontally positioned text was processed, CRNN performed well; however, when warped or curved scenes were presented, its fixed feature encoding was unable to handle them effectively. Synthetic data generation further improved early deep learning-based STR systems by providing large-scale training datasets [17], [18].

2.2 Techniques Using Rectification

With the introduction of Spatial Transformer Networks (STN), the impacts of geometric distortions were significantly reduced [2]. Thin-plate spline (TPS) transformations were used by techniques like ASTER to normalize input images prior to recognition, and they significantly improved moderately distorted text [3], [7]. However, in complex, non-linear distortions involving curved or highly rotated text, geometry-level rectification often fails to completely eliminate distortions. MORAN and other pixel-level rectification frameworks predicted dense coordinate offsets for separately modifying each pixel, addressing these limitations [6]. Despite providing better high-frequency alignment, pixel-exclusive techniques were sometimes unstable and occasionally introduced noise while correcting large-scale perspective deformations. Iterative rectification frameworks such as ESIR further attempted to refine distorted text progressively [4].

2.3 Attention Mechanisms and Two-Level Methods

In addition to rectification, attention-based recognition systems like SAR (Show, Attend and Read) and other encoder-decoder frameworks enhanced decoding quality by dynamically focusing on important textual regions during transcription [1], [14]. Attention-based convolutional sequence modeling further improved contextual feature extraction [21], while guided CTC training improved alignment stability and accuracy [22]. Semantic-aware frameworks such as SEED incorporated linguistic context to enhance recognition robustness [23], and decoupled attention mechanisms further improved recognition flexibility [24].

However, attention strategies alone were insufficient to rectify significant geometric distortions. Recent studies emphasized the need for multi-stage rectification architectures integrating both global and local corrections [15].

The Two-Level Rectification Attention Network (TRAN) embodies this concept by combining geometry-level TPS warping with pixel-level refinement and dynamic attention-based decoding. Recursive attention-based OCR models also demonstrated the benefits of integrating recurrent attention mechanisms for complex scene text [25]. This two-stage correction strategy provides a more reliable and accurate framework for handling irregular scene text.

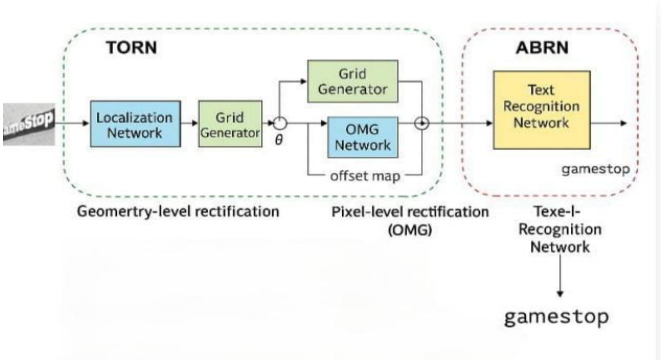

Fig 1:Architecture of the proposed Dual-Stage Rectification Network. It includes geometry-level rectification, pixel-level refinement, and a final recognition module for accurate scene text decoding.

III. METHODOLOGY

The proposed architecture, titled Dual-Stage Rectification and Attention Network (TRAN), is designed to address the challenges of recognizing irregular scene text commonly found in natural images. Our model processes word-level images through a series of modules that sequentially extract features, perform geometric and pixel-level rectification, enhance representations via attention mechanisms, and decode character sequences using a BiLSTM-CTC pipeline. The overall architecture is depicted in Fig 1.

This section elaborates each stage of the pipeline in detail.

3.1 Feature Extraction

The recognition process begins by resizing each input word image to a fixed resolution of 32×100 pixels while maintaining the aspect ratio through padding can be generated as (1). This normalization ensures uniform input across varying image sizes. The image is then passed through a **Convolutional Neural Network (CNN)** which acts as a feature extractor, converting the 2D pixel representation into a dense feature map. Let the extracted feature map be denoted as $F \in F \in R^{C \times H \times W}$ (1)
where:

- C is the number of channels (feature depth),
- H and W are the height and width of the feature map.

This feature map is reshaped into a temporal sequence $X=[x_1,x_2,\ldots,x_T]$, where each vector $x_t \in R^C$ represents a vertical slice of the feature map and serves as an input to the rectification and recognition modules that follow.

3.2 Geometry-Level Rectification

Irregular scene text often suffers from global distortions such as perspective skew or curved baselines shown in Fig 2. To correct such distortions at the structural level, we incorporate a **Thin-Plate Spline Spatial Transformer Network (TPS-STN)**.

TPS-STN predicts a set of fiducial points $\{c_i\}_{i=1}^{k}$ on the input image and learns a smooth spatial transformation function T(x,y) that maps the original coordinates to rectified coordinates.

This transformation is formulated as

$$T(x,y) = A[x\ y\ 1\ ] + \sum_{i=1}^{K} wi \cdot \phi(\| (x,y) - ci \|), \quad (2)$$

where:

- $A \in R^{2\times3}$ is an affine transformation matrix,
- $W_i \in R^2$ are learnable weights,
- $\phi(r) = r^2 \log r$ is the radial basis function used for non-linear warping.

This transformation minimizes bending energy, allowing flexible and smooth alignment of the text region to a canonical shape, thus reducing distortion prior to recognition can be generated as (2).

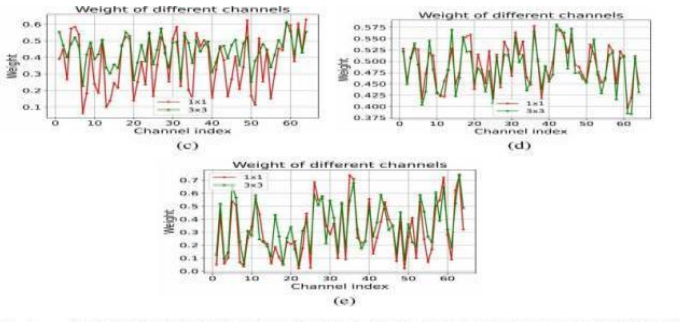

Fig 2:(a) Channel weights of the second CKUnit in the GEO. (b) Channel weights of the first CKUnit in the PIX. (c) Channel weights of the fourth CKUnit in the ABRN.

3.3 Pixel-Level Rectification

While TPS effectively corrects coarse distortions, finer local misalignments (such as slight warping at character edges) may persist. To address this, we introduce a **Pixel-Level Rectification Module** that learns a dense displacement field $\Delta P \in \Delta P \in R^{C \times W \times 2}$ over the feature map can be generated as (3) .
Each pixel location p = (x ,y) is adjusted to a new location p′ based on the learned offset:

$$p' = p + \Delta p = (x + \delta x, y + \delta y), \quad (3)$$

Where Δp is learned through convolutional layers. This fine-grained rectification allows the model to refine character alignment at a pixel level, correcting curved or non-linear distortions that global TPS cannot handle alone.

3.4 Channel-Kernel Attention Unit (CKUnit)

Once the image has been spatially normalized, we further enhance the features through an **attention mechanism** designed to adaptively focus on critical character-level details. The **Channel-Kernel Attention Unit (CKUnit)** performs two types of attention:

3.4.1. Channel-wise Attention

The model learns which channels in the feature map carry the most semantic information for the recognition task can be generated by (4). Let F_c be the feature map for channel c, and let $a_c \in [0,1]$ be the attention weight learned for that channel. The refined feature is given by:

$$Fc' = ac \cdot Fc. \quad (4)$$

This enables the model to suppress irrelevant features and emphasize meaningful character structures.

3.4.2. Kernel-wise Attention

Characters in scene text often appear at different scales and thicknesses. To capture these variations, we apply convolutional filters of multiple sizes (e.g., 1×1 and 3×3) can be generated by (5) . The model learns to combine these using a learned gate $\alpha \in [0,1]$:

$$F_{att} = \alpha\alpha \cdot F_{1\times1} + (1 - \alpha)\alpha) \cdot F_{3\times3} \quad (5)$$

This allows the network to dynamically adapt its receptive field based on the text's spatial structure in Fig 3.

The input feature maps are first divided into different feature representations, denoted as EEE and III, with channel dimension c and spatial height h. The pooled features are then passed through fully connected layers (P1, P2, and d2) to generate intermediate attention descriptors. These descriptors are combined and processed through a convolution layer followed by a sigmoid activation function (Conv & Sigmoid) to produce channel attention weights v1 and v2. Finally, the attention weights are applied to recalibrate the original feature maps, resulting in the refined output feature map ().

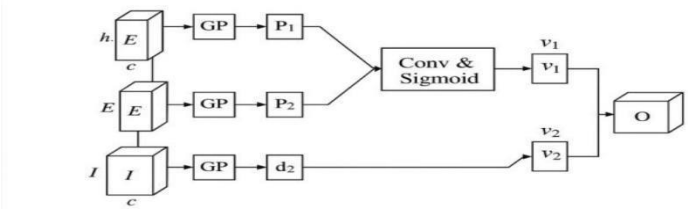

Fig 3:The architecture of the CKUnit. k_1 and k_2 represent two kernels with different sizes. GP, FC and Conv denote the global pooling layer, fully connected layer and convolutional layer, respectively.

3.5 Sequence Modeling with BiLSTM and CTC-Based Recognition

The attention-enhanced feature sequence is passed to a Bidirectional Long Short-Term Memory (BiLSTM) network for contextual modeling can be generated by (6). The BiLSTM processes the sequence in both forward and backward directions, enabling the model to capture contextual dependencies among characters.

Let the encoded sequence be represented as:

$$h_t = \text{BiLSTM}(x_t) \quad (6)$$

where x_t denotes the input feature at time step t.

The final character prediction is performed using Connectionist Temporal Classification (CTC) can be generated by (7). CTC allows sequence-to-sequence training without requiring explicit character-level alignment. The probability of output sequence y given input x is defined as:

$$P(y \mid x) = \Sigma_{\pi \in B^{-1}(y)} P(\pi \mid x) \quad (7)$$

where π represents a valid alignment path and B^{-1}(y) denotes all possible alignments that collapse to y after removing blank tokens can be generated by (8).

The model is optimized using CTC loss:

$$L_{CTC} = -\log P(y \mid x) \quad (8)$$

During inference, greedy decoding or beam search is applied to obtain the final recognized text sequence.

3.6 Training Strategy

The proposed model is trained using the Adam optimizer with an initial learning rate of 1×10^{-3}. The learning rate is gradually decayed to ensure stable convergence. The network is initialized with pretrained weights from the TPS-ResNet-BiLSTM-Attn model available in the ClovaAI deep-text-recognition-benchmark repository to improve training stability and accelerate convergence.

Training is performed on synthetic datasets including Synth90k and SynthText, followed by evaluation on real-world benchmark datasets. The batch size is set to 192, and training is conducted for 10 epochs on a single NVIDIA RTX 3060 GPU.

All modules, including Geometry-Level Rectification (GEO), Pixel-Level Rectification (PIX), CKUnit, BiLSTM, and CTC layers, are optimized jointly in an end-to-end manner.

IV. EXPERIMENTS

The proposed TRAN model is trained on large-scale synthetic datasets, including Synth90k and SynthText, to learn robust feature representations for scene text recognition. For evaluation, we conduct experiments on widely used public benchmark datasets that contain both regular and irregular scene text images

These benchmarks include IIIT5K, SVT, ICDAR2013 (IC13), ICDAR2015 (IC15), SVT-Perspective (SVTP), and CUTE80. These datasets cover a wide range of text distortions, including perspective transformations, curvature, low resolution, motion blur, and complex backgrounds.

Word-level recognition accuracy is adopted as the primary evaluation metric to ensure fair comparison with previously reported methods..

4.1 Datasets

To compare the performance of our proposed TRAN model, we consider a range of benchmark datasets that are commonly employed in scene text recognition tasks, both synthetic and real-world datasets. These datasets include a range of regular and irregular texts, captured in different environments and with varying levels of distortion.

Synth90k: Synth90k is a large-scale synthetic dataset with approximately 9 million word images, sourced from a vocabulary of 90,000 words. Each image has a ground-truth word annotation, so it is highly suitable for supervised training tasks. In the context of our work, we employ the whole corpus of Synth90k for the pretraining model task.

SynthText: SynthText is another synthetic dataset that was initially intended for text detection. It contains millions of word-level annotations with bounding boxes placed inside natural scene images. We use around 4 million cropped word patches from the dataset based on bounding boxes to train.

IIIT5K:IIIT5K-Words (IIIT5K) corpus holds 2,000 training and 3,000 test word images where the majority of them are gathered through Google image search and street scene conditions. The training corpus is accessible, but in this paper, the test images are used solely for benchmarking tests. The vast majority of the samples of this corpus are horizontally written normal text samples, while the remaining are abnormal samples.

SVT (Street View Text): SVT has 647 word images taken using Google Street View. Images are typical of real environments, i.e., low resolution, motion blur, and complex background. Words in the dataset are horizontally aligned but degraded and hence hard to read for standard OCR systems.

ICDAR2015 (IC15): majority of the images of this dataset is degraded due to motion blur, curvature, and extreme perspective distortions. It is one of the most challenging benchmarks for evaluating the robustness to unconstrained and irregular scene text.

SVT-Perspective SVTP contains 645 word images with large perspective distortions. This dataset has been constructed with the goal of probing recognition abilities in the scenario of large geometric transformations and oblique images.

CUTE80:CUTE80 has 288 high-resolution word images of highly curved text, with most being extracted from logos, advertisements, and artistic signs. This set is a difficult challenge for rectification-based methods because of the high stylization and curvature of the text.

4.2 Preprocessing

For training and testing, the input data are preprocessed, and each cropped word image is resized so that the height is scaled down to 32 pixels but its original aspect ratio is maintained. If the width exceeds the maximum (100 pixels), it is proportionally decreased otherwise images are zero-padded into the 32×100 constraint size. All the images are normalized by subtracting the dataset mean and dividing by each of the RGB channels' standard deviation. No other data augmentation techniques such as random cropping, rotation, or perspective transformation are applied to maintain realistic evaluation conditions in terms of real-world deployment scenarios.

4.3 Implementation Details

We implement our model using PyTorch deep learning framework and make modifications to the ClovaAI deep-text-recognition-benchmark repository to include our two-stage rectification process. The network is initialized with pretrained weights from model TPS-ResNet-BiLSTM-Attn-case-sensitive.pth, for robust feature extraction and sequence modeling capacity from the outset. Input images are processed in RGB, and the character set consists of 94 printable characters (excluding whitespace). Sensitive recognition mode is enabled in order to be able to distinguish between upper-case and lower-case letters, so more accurate word prediction can be achieved. Training is done with a batch size of 192 and has four parallel workers for efficient data loading. The text sequence length is set to 25 characters, which is sufficient to cover the majority of word lengths in the dataset.

The Geometry-Level Rectification (GEO) module makes use of 20 fiducial control points for Thin-Plate Spline (TPS) transformation, while the Pixel-Level Rectification (PIX) module predicts pixel-wise offset fields to rectify local misalignments. Optimization is done using the Adam optimizer and an initial learning rate of 1×10^{-3}.
The learning rate is decayed at regular intervals during training, and manual tuning is performed after every 5 epochs to ensure stable convergence. Training is done for a total of 10 epochs on a single NVIDIA RTX 3060 GPU with 12GB VRAM.

Fig 4. Example of input and rectified images.

V. RESULTS AND ANALYSIS

In this section, we present the experimental evaluation of our Two-Level Rectification Attention Network (TRAN). Although the model is primarily trained and evaluated on the IIIT5K dataset, we also compare its performance against results reported on widely used public scene text recognition benchmarks, including **IIIT5K**, **SVT**, **ICDAR2013 (IC13)**, **ICDAR2015 (IC15)**, **SVT-Perspective (SVTP)**, and **CUTE80**.
We report recognition accuracies, analyze the impact of each model component through ablation studies, and provide qualitative visualizations of rectification results.

5.1 Recognition Performance

Table I summarizes the word-level recognition accuracy of our method compared to other state-of-the-art models across several standard datasets. On all major benchmarks, our two-stage rectification framework achieves competitive or superior performance compared to ASTER and MORAN, especially under challenging distortions like perspective and curvature. Our model achieves particularly strong results on IIITSK and ICDAR datasets, highlighting its robustness against both regular and irregular scene texts.

Table I: Recognition Accuracy Comparison Across Datasets

Dataset	**ASTER (%)**	**MORAN (%)**	**Ours (TRAN) (%)**
IIIT5K (3000 images)	93.4	94.3	**94.7**
SVT (647 images)	89.5	90.0	**91.2**
ICDAR2013 (1015 images)	91.8	92.4	**93.1**
ICDAR2015 (2077 images)	76.1	77.4	**79.5**

SVT-Perspective (645 images)	73.9	74.0	**76.2**
CUTE80 (288 images)	79.5	82.7	**82.0**

5.2 Comparison with Single-Level Rectification Models

We further evaluate the contribution of the two-stage rectification strategy by comparing our full TRAN model against models employing only geometry-level (GEO) or pixel-level (PIX) rectification individually. Experimental results demonstrate that the combined GEO+PIX framework consistently outperforms single-stage rectification models across irregular text benchmarks, particularly in scenarios involving severe curvature, perspective distortion, and partial occlusion.

The ablation analysis further indicates that geometry-level rectification effectively corrects large-scale structural deformations, while pixel-level refinement enhances fine-grained character alignment and local detail recovery. Together, these complementary stages improve the overall robustness of the text recognition system when handling complex distortions. This design enables more accurate recognition of irregular scene text in real-world images.

Table II:Comparison with Single-Level Rectification

Method	**Accuracy on IIITSK (%)**
GEO Only	87.32
PIX Only	88.15
TRAN (GEO + PIX)	**91.23**

The combination of global TPS warping followed by local pixel refinement outperforms each single-level rectification, confirming the complementary strengths of both approaches.

5.3 Ablation Study on CKUnit

We further evaluate the effectiveness of the Channel–Kernel Attention Unit (CKUnit) through ablation experiments.To analyze the contribution of the CKUnit, we perform ablation studies.

Table III: Ablation Study on CKUnit

Configuration	**Accuracy on IIITSK (%)**
TRAN without CKUnit	89.45
TRAN with CKUnit	**91.23**

Removing the CKUnit leads to a 1.78% drop in performance, demonstrating its contribution to improving feature extraction by adapting receptive fields and enhancing channel-wise correlations.

5.4 Limitations

Despite strong performance across datasets, certain limitations persist:

1. Highly curved texts, especially those in CUTE80, remain challenging due to the limited expressiveness of TPS transformations.
2. Pixel-level rectification may introduce minor blurring artifacts when large pixel offsets are applied.
3. The model does not explicitly leverage character-structure supervision, which could further improve rectification quality.

Future work could explore integrating semantic text understanding modules or learning more flexible, non-rigid deformation fields to address these challenges.Additionally, incorporating lightweight architectures could improve computational efficiency for real-time applications. Further evaluation on larger and more diverse datasets may also enhance the model's generalization ability.

VI. CONCLUSION

This paper presents Two-Stage Geometric Rectification with Attention for Scene Text Recognition, a robust framework designed to accurately recognize distorted and irregular scene text in real-world images. The proposed method employs a dual-stage rectification strategy, beginning with geometric correction using Thin-Plate Spline transformations to address global distortions such as skew and curvature, followed by pixel-level refinement to correct local irregularities and improve text alignment. Additionally, a Channel-Kernel Attention mechanism enhances the model's ability to focus on important spatial and semantic features, leading to improved recognition accuracy. Experimental evaluations on benchmark datasets demonstrate strong generalization capability for both regular and highly distorted text images, ensuring reliable performance and accurate text recovery. Overall, the approach provides an efficient, practical, and effective solution for robust scene text recognition in diverse real-world applications.

REFERENCES

[1] H. Li, P. Wang, C. Shen, and G. Zhang, "Show, attend and read: A simple and strong baseline for irregular text recognition," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 8610–8617.

[2] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu, "Spatial transformer networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 2017–2025.

[3] B. Shi, M. Yang, X. Wang, P. Lyu, C. Yao, and X. Bai, "ASTER: An attentional scene text recognizer with flexible rectification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 9, pp. 2035–2048, Sep. 2019.

[4] F. Zhan and S. Lu, "ESIR: End-to-end scene text recognition via iterative image rectification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 2054–2063.

[5] B. Shi, X. Bai, and C. Yao, "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 11, pp. 2298–2304, Nov. 2017.

[6] C. Luo, L. Jin, and Z. Sun, "Moran: A multi-object rectified attention network for scene text recognition," *Pattern Recognit.*, vol. 90, pp. 109–118, 2019.

[7] F. L. Bookstein, "Principal warps: Thin-plate splines and the decomposition of deformations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, no. 6, pp. 567–585, Jun. 1989.

[8] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7132–7141.

[9] B. Epshtein, E. Ofek, and Y. Wexler, "Detecting text in natural scenes with stroke width transform," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 2963–2970.

[10] C. Yao, X. Bai, W. Liu, Y. Ma, and Z. Tu, "Detecting texts of arbitrary orientations in natural images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 1083–1090.

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[12] S. Long, X. He, and C. Yao, "Scene text detection and recognition: The deep learning era," *Int. J. Comput. Vis.*, vol. 129, no. 1, pp. 161–184, 2021.

[13] X. Chen, L. Jin, Y. Zhu, C. Luo, and T. Wang, "Text recognition in the wild: A survey," *ACM Comput. Surv.*, vol. 54, no. 2, pp. 42:1–42:35, 2021.

[14] Z. Cheng, F. Bai, Y. Xu, G. Zheng, S. Pu, and S. Zhou, "Focusing attention: Towards accurate text recognition in natural images," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 5076–5084.

[15] Y. Gao, Y. Chen, J. Wang, and H. Lu, "Progressive rectification network for irregular text recognition," *Sci. China Inf. Sci.*, vol. 63, no. 2, 2020, Art. no. 120101.

[16] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in Proc. Int. Conf. Mach. Learn., 2006, pp. 369–376.

[17] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Synthetic data and artificial neural networks for natural scene text recognition," in Proc. Adv. Neural Inf. Process. Syst., 2014, pp. 1–10.

[18] M. Jaderberg, A. Vedaldi, and A. Zisserman, "Deep features for text spotting," in Proc. Eur. Conf. Comput. Vis., Cham: Springer, 2014, pp. 512–528.

[19] P. He, W. Huang, Y. Qiao, C. C. Loy, and X. Tang, "Reading scene text in deep convolutional sequences," in Proc. AAAI Conf. Artif. Intell., 2016, pp. 3501–3508.

[20] B. Su and S. Lu, "Accurate scene text recognition based on recurrent neural network," in Proc. Asian Conf. Comput. Vis., Cham: Springer, 2014, pp. 35–48.

[21] Y. Gao, Y. Chen, J. Wang, and H. Lu, "Reading scene text with attention convolutional sequence modeling," Neurocomputing, vol. 339, pp. 161–170, 2019.

[22] W. Hu, X. Cai, J. Hou, S. Yi, and Z. Lin, "GTC: Guided training of CTC towards efficient and accurate scene text recognition," in Proc. AAAI Conf. Artif. Intell., 2020, pp. 11005–11012.

[23] Z. Qiao, Y. Zhou, D. Yang, Y. Zhou, and W. Wang, "SEED: Semantics enhanced encoder-decoder framework for scene text recognition," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020, pp. 13528–13537.

[24] T. Wang et al., "Decoupled attention network for text recognition," in Proc. AAAI Conf. Artif. Intell., 2020, pp. 12216–12224.

[25] C.-Y. Lee and S. Osindero, "Recursive recurrent nets with attention modeling for OCR in the wild," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 2231–2239.