Linguistic More: Taking a Further Step toward Efficient and Accurate Scene Text Recognition

Boqiang Zhang , Hongtao Xie* , Yuxin Wang , Jianjun Xu , Yongdong Zhang
University of Science and Technology of China, Hefei, China
{cyril, wangyx58, xujj1998}@mail.ustc.edu.cn, {htxie, zhyd73}@ustc.edu.cn

Abstract

Vision model have gained increasing attention due to their simplicity and efficiency in Scene Text Recognition (STR) task. However, due to lacking the perception of linguistic knowledge and information, recent vision models suffer from two problems: (1) the pure vision-based query results in attention drift, which usually causes poor recognition and is summarized as linguistic insensitive drift (LID) problem in this paper. (2) the visual feature is suboptimal for the recognition in some vision-missing cases (e.g. occlusion, etc.). To address these issues, we propose a **L**inguistic **P**erception **V**ision model (LPV), which explores the linguistic capability of vision model for accurate text recognition. To alleviate the LID problem, we introduce a Cascade Position Attention (CPA) mechanism that obtains high-quality and accurate attention maps through step-wise optimization and linguistic information mining. Furthermore, a Global Linguistic Reconstruction Module (GLRM) is proposed to improve the representation of visual features by perceiving the linguistic information in the visual space, which gradually converts visual features into semantically rich ones during the cascade process. Different from previous methods, our method obtains SOTA results while keeping low complexity (92.4% accuracy with only 8.11M parameters). Code is available at https://github.com/CyrilSterling/LPV.

1 Introduction

Scene Text Recognition (STR) is a meaningful task in computer vision that aims to understand the textual information from the cropped image of natural scenes [Long *et al.*, 2021; Shi *et al.*, 2016a; Fang *et al.*, 2021; Xu *et al.*, 2022; Sheng *et al.*, 2019]. Due to the lack of language modal information of other perception tasks, STR is widely used in Visual Questions and Answers (VQA), automatic pilots, *etc.*

Early work has generally treated STR as a visual task, using an encoder to get visual features and, after sequence modeling, a CTC-based [Graves *et al.*, 2006] or attention-based decoder to obtain the predicted characters. The attention-based decoder uses a visual query to decode the position of each character, so it is accurate for arbitrary shape text recognition. In addition, due to its simplicity and effectiveness, the attention-based decoder is currently the mainstream solution for vision model [Wang *et al.*, 2021; Fang *et al.*, 2021; Wang *et al.*, 2022a; Zheng *et al.*, 2021; Wang *et al.*, 2022b]. Such methods have a simple structure that can be efficient in most application scenarios. Though these vision-only methods have achieved promising results, there are still two problems.

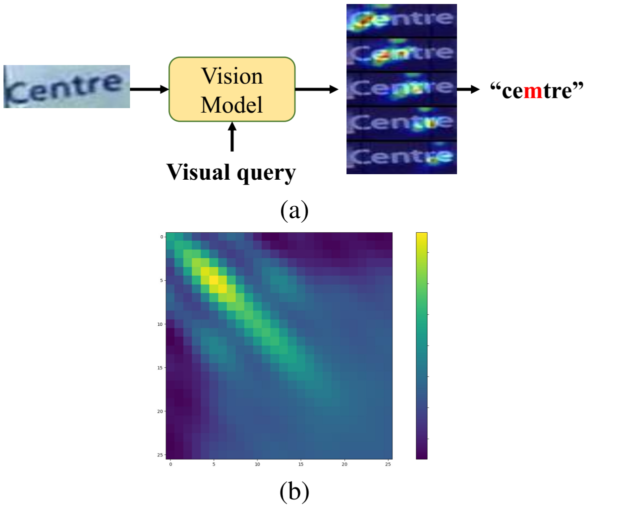

Figure 1: Our motivation. (a) The basic structure of previous vision-only models with an attention-based decoder and a visual query to decode the characters, which has the problem of attention drift. (b) The visualization of the dot similarity between the query vectors at each position of ABINet [Fang *et al.*, 2021].

The first problem is the attention drift in the attention-based decoders, which is not received enough attention in recent researches. Attention drift is when the area of attention region is not aligned with the target character (Figure. 1 (a)). RobustScanner [Yue *et al.*, 2020] deeply analyzed attention drift and proved that the query vectors in the decoder encode not only context but also positional information. However, this positional information is easily drowned out by the in-

[*]Corresponding Author

troduction of other information. Note that recent methods use a pure vision-based query, which is fixed when inputting different images. We further visualize the dot similarity between the query vectors at each position in ABINet [Fang *et al.*, 2021]. As shown in Figure. 1 (b), it is observed that the query at each position is similar to that at neighboring positions. This will lead to similar features when decoding the attention map of neighboring characters, thereby causing attention drift. Therefore, we indicate that the attention drift comes from decoding different images with the fixed vision-based query, which is linguistic insensitive. We summarize this issue as the Linguistic Insensitive Drift (LID) problem. Thus, how to eliminate the LID issue and obtain an accurate attention map is the key for robust text recognition.

Another problem is that visual feature is suboptimal for recognition in some vision-missing cases. To solve this problem, recent methods introduce the linguistic knowledge to assist the vision model. However, it is hard for vision model to obtain linguistic information efficiently and accurately. VisionLAN [Wang *et al.*, 2021] designed a masked language-aware module to randomly occlude a character in the training stage which guides the vision model to utilize the linguistic information in text images. But the model introduces additional modules and requires separate pre-training. MGP-STR [Wang *et al.*, 2022a] proposed a multi-granularity prediction strategy to inject information from the language modality into the model in an implicit way. However, this network requires a huge number of parameters. Thus, how to perceive linguistic information with an efficient structure and a simple training strategy is a great challenge for text recognition.

To enhance the linguistic perception of both query and feature in a simple way, we propose a concise Linguistic Perception Vision model (LPV). The pipeline of our LPV is shown in Figure. 2. The pipeline mainly consists of two parts: the GLRM branch and the CPA branch. The GLRM branch continuously enhances the features of the input image. In this branch, the visual features $\mathbf{F}^0$ are firstly extracted from the backbone. Then, Global Linguistic Reconstruction Module (GLRM) enhances the features of the previous stage $\mathbf{F}^{i-1}$ into the features of the current stage $\mathbf{F}^i$ using the mask generated by the attention map $\mathbf{A}^{i-1}$. In this way, linguistic information is aggregated in GLRM and the visual features can be gradually transformed into semantic-rich features. Meanwhile, the GLRM ensures the simplification of the pipeline and there are no redundant modules. The CPA branch hierarchically optimizes the attention map and the query using the cascade position attention mechanism. Each Position Attention Module (PAM) takes the visual features $\mathbf{F}^i$ as input, and obtains the attention map $\mathbf{A}^i$ and the features $\mathbf{R}^i$ of each character. Note that the prior query of the first PAM is initialized to $\mathbf{0}$, which means we have no prior to each character. PAM at i^{th} stage uses $\mathbf{R}^{i-1}$ as the prior query. Such an operation can take the recognition result of the previous stage as a priori and re-perform the similarity calculation for the enhanced features to obtain more accurate attention positions. Meanwhile, positional and linguistic information is constantly introduced in the PAM, so as to alleviate the linguistic insensitive drift problem. Compared with previous methods, we have a more concise structure and training strategy, while achieving better performance.

The main contributions of our work are as follows:

- We are the first to point out the attention drift due to lack of linguistic information, which is called Linguistic Insensitive Drift (LID) problem, and propose a Cascade Position Attention mechanism to effectively handle the LID problem.
- We propose a Global Linguistic Reconstruction Module to reconstruct the features of each character by aggregating global linguistic information during the process of sequence modeling. The method does not introduce extra parameters.
- Our method achieves state-of-the-art performance while keeping a very low parameter quantity with a simple end-to-end training strategy.

2 Related Work

2.1 Scene Text Recognition

Scene Text Recognition (STR) has been a significant research term in computer vision. Early methods use a backbone and a sequence modeling network for feature extraction and use a Connectionist Temporal Classification (CTC) [Graves *et al.*, 2006] decoder or attention decoder for prediction [Qiao *et al.*, 2020; Lyu *et al.*, 2019a]. CTC-based decoder aims to maximize the probability of all the paths for final prediction, while attention-based decoder aims to localize the position of each character by attention mechanism. To further extract linguistic information of the visual predictions, SRN [Yu *et al.*, 2020] proposed a language model to learn the relationship between each character. ABINet [Fang *et al.*, 2021] further proposed a stronger bi-directional language model for autonomous linguistic modeling. We believe that a powerful recognizer must have the ability of contextual linguistic modeling, but explicit language models have a large number of parameters, which severely limits recognition efficiency.

Recently, the simplicity of model reasoning has been emphasized. Considering the CTC-based decoder has an advantage in speed while the attention-based decoder has an advantage in precision, GTC [Hu *et al.*, 2020] used a powerful attention-based decoder to guide the training of a CTC-based decoder. MGP-STR [Wang *et al.*, 2022a] used ViT as the backbone to achieve high performance, which proved that the structure of ViT is applicable to STR. Further, SVTR [Du *et al.*, 2022] proposed a faster and more lightweight backbone for STR task. We think that in order to design a more powerful recognizer, the model must have linguistic modeling capability while keeping the structure simple. Thus, we propose the Global Linguistic Reconstruction Module, which can aggregate the contextual linguistic information during the process of sequence modeling. Such a design ensures the simplicity of the model.

2.2 Attention Drift

The visual attention drift problem in STR refers to the fact that when an attention-based decoder is used, the attention region of the decoder cannot be accurately aligned with the

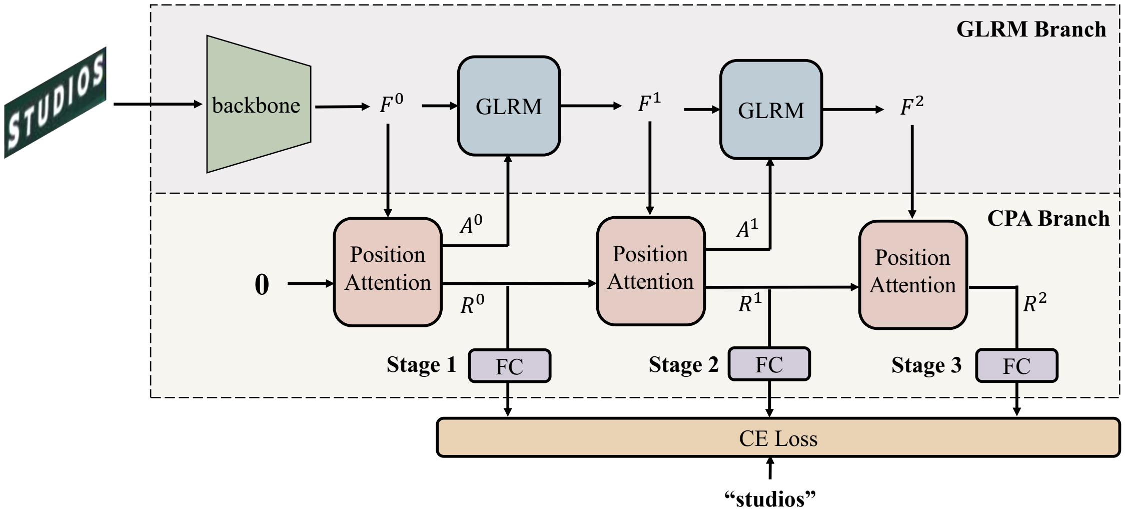

Figure 2: The pipeline of our LPV. The pipeline mainly contains two branches: GLRM branch and CPA branch. GLRM branch continuously enhances the feature using proposed GLRM. CPA branch takes the feature as input and hierarchically decodes the attention map and feature of each character. CE Loss means the cross-entropy loss.

target character. [Cheng *et al.*, 2017] first identified this problem and proposed a Focusing Attention Network (FAN) that is composed of an attention network for character recognition and a focusing network to adjust the attention drift. RobustScanner [Yue *et al.*, 2020] deeply investigated the decoding process of the attention-based decoder and empirically find that a character-level sequence decoder utilizes not only context information but also positional information. They further suggested that the drowning of position information leads to attention drift problems. Using the above analysis, they solve this problem by a position enhancement branch to introduce position information. We further point out that attention drift comes from decoding different images with a linguistic insensitive query, which also lacks positional information. Based on this, we propose a Cascade Position Attention mechanism to solve this problem, which has a concise framework and does not introduce extra modules.

3 Proposed Method

In this section, we first detail the pipeline of proposed method in Sec. 3.1, and then we introduce Cascade Position Attention and Global Linguistic Reconstruction Module in Sec. 3.2 and Sec. 3.3 respectively.

3.1 Pipeline

The pipeline of our LPV is shown in Figure.2. We can view the pipeline as two branches. Given an input image of size $H \times W \times 3$, the features $\mathbf{F}^i \in \mathbb{R}^{\frac{H}{4} \times \frac{W}{4} \times E}$ are obtained by the GLRM branch, which continuously enhances the features to obtain long-distance contextual linguistic information using proposed GLRM. Meanwhile, in the CPA branch, $\mathbf{F}^i$ are fed into the i^{th} Position Attention Module (PAM) to get the attention map $\mathbf{A}^i$ and the feature $\mathbf{R}^i$ of each character. The CPA branch constantly rectifies the recognition results to alleviate linguistic insensitive drift using a linguistic-sensitive query. Note that the parameters in each PAM are *NOT* shared.

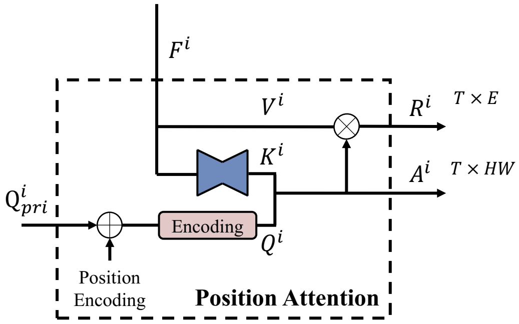

Figure 3: The structure of Position Attention Module in the CPA. $\mathbf{Q}_{pri}$ is the prior query. 'Encoding' can be one FC layer.

3.2 Cascade Position Attention

The Cascade Position Attention mechanism hierarchically optimizes the recognition result using the enhanced feature $\mathbf{F}^i$ in each stage and outputs the attention map of each character.

As shown in Figure. 3, a cross-attention mechanism is utilized to transcribe visual features into character sequences. Specifically, the attention map $\mathbf{A}^i \in \mathbb{R}^{T \times \frac{HW}{16}}$ and the features $\mathbf{R}^i \in \mathbb{R}^{T \times E}$ of each character is calculated by the queries, keys, and values as Eq. 1, where T is the maximum length of the character sequence. The prediction results $\mathbf{Y}^i \in \mathbb{R}^{T \times C}$ can be further obtained by a classification head (e.g. FC Layer), where C indicates the number of character classes. $\mathcal{P}(\cdot)$ is the classification head.

$$\begin{aligned} \mathbf{A}^i &= softmax(\mathbf{K}^i \mathbf{Q}^{i\top} / \sqrt{E}) \\ \mathbf{R}^i &= \mathbf{A}^i \mathbf{V}^i \\ \mathbf{Y}^i &= sofmax(\mathcal{P}(\mathbf{R}^i)) \end{aligned} \tag{1}$$

Concretely, $\mathbf{K}^i = \mathcal{G}(\mathbf{F}^i) \in \mathbb{R}^{\frac{HW}{16} \times E}$, where $\mathcal{G}(\cdot)$ is imple-

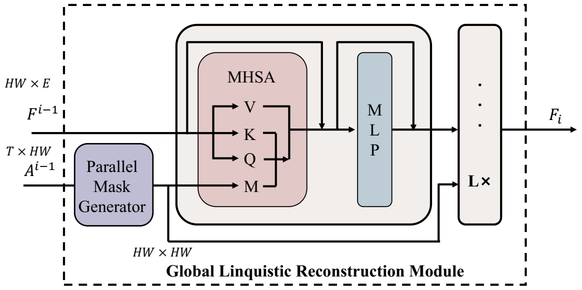

Figure 4: The structure of Global Linguistic Reconstruction Module. GLRM consists of a Parallel Mask Generator and L × Masked Transformer Encoder.

mented by a mini U-Net. $\mathbf{V}^i = \mathcal{H}(\mathbf{F}^i) \in \mathbb{R}^{\frac{HW}{16} \times E}$, where $\mathcal{H}(\cdot)$ is identity mapping. The most important, $\mathbf{Q}^i \in \mathbb{R}^{T \times E}$ is used to decode the position of each character and can be regarded as the priori of each character. Therefore, $\mathbf{Q}^i$ is generated by a given prior $\mathbf{Q}^i_{pri}$ and a position encoding $\mathbf{P}$ through the encoding layer (e.g. one FC layer). At the beginning of decoding, we do not know the specific information about the character so $\mathbf{Q}^0_{pri}$ is initialized to the $\mathbf{0}$ vectors. At the i^{th} stage of decoding, $\mathbf{Q}^i_{pri}$ is set as the features of each character from the previous stage. The generation process of $\mathbf{Q}^i$ can be formalized as follows:

$$\begin{aligned} \mathbf{Q}^i_{pri} &= \begin{cases} 0, & i = 0 \\ \mathbf{R}^{i-1}, & otherwise \end{cases} \\ \mathbf{Q}^i &= \mathcal{F}(\mathbf{Q}^i_{pri} + \mathbf{P}) \end{aligned} \tag{2}$$

Where $\mathcal{F}(\cdot)$ is the encoding layer.

To deal with the problem of linguistic insensitive drift, on the one hand, through the continuous iteration of $\mathbf{Q}^i$, the network gradually gets a linguistic-sensitive query to decode the attention map. On the other hand, the positional information is constantly introduced by position encoding, which can enhance the positional sensitivity of the model.

3.3 Global Linguistic Reconstruction Module

We argue that the input feature $\mathbf{F}^i$ of each stage can not be the same and it needs to be dynamically adjusted, e.g. sequence modeling. We will prove this inference in the ablation study. Therefore, it is necessary to add a sequence modeling network between stages but the simple sequence modeling network has no linguistic awareness, so we propose Global Linguistic Reconstruction Module to aggregate global linguistic information during sequence modeling without introducing extra parameters.

The details of GLRM is shown in Figure. 4, it takes the feature and attention map of the previous stage as inputs and outputs the enhanced feature of the current stage. GLRM contains two parts: Parallel Mask Generator (PMG) and Masked Transformer Encoder. The transformer encoder is proven to be effective for modeling long-range dependencies in recent computer vision tasks [Carion *et al.*, 2020; Lyu *et al.*, 2019b], which can be used well for sequence modeling and contextual information aggregating. To guide the model learning linguistic knowledge, we design a novel way that reconstructs the features of each character by masking each character. To achieve this, PMG transforms the attention map $\mathbf{A}^{i-1}$ into a parallel mask $M \in \mathbb{R}^{\frac{HW}{16} \times \frac{HW}{16}}$ as Eq. 3, where $\mathbf{U}(x)$ is the unit stage function which takes the value of 1 for $x \geq 0$ and 0 for $x < 0$. t is the threshold of foreground and background, which is set to 0.05 in our experiments. $\otimes$ is matrix multiplication.

$$\begin{aligned} M^p &= \mathbf{U}(\mathbf{A}^{i-1} - t)^\top \otimes \mathbf{U}(\mathbf{A}^{i-1} - t) \\ M_{ij} &= \begin{cases} -\infty, & M^p_{ij} > 0 \\ 0, & otherwise \end{cases} \end{aligned} \tag{3}$$

Then, the attention operation inside multi-head self-attention blocks can be formalized as follows:

$$\begin{aligned} [\mathbf{Q}, \mathbf{K}, \mathbf{V}] &= \mathbf{F^{i-1}W} \\ \mathbf{F}^i &= softmax(\frac{\mathbf{QK}^\top}{\sqrt{E}} + M)\mathbf{V} \end{aligned} \tag{4}$$

By using such a mask, the tokens in one character can not see the tokens in the same character during the self-attention operation, which means that features within each character region are reconstructed from features other than that character. Benefiting from such a design, the visual features are gradually transformed into semantic-rich features during the cascade stage.

Compared with BERT [Devlin *et al.*, 2018] and VisionLAN [Wang *et al.*, 2021], though all approaches mask out the information in a certain time step, there are two differences: 1) BERT and VisionLAN mask the tokens of the input features, which leads to loss of origin features. But GLRM masks the tokens in the self-attention operation, which can ensure self-attention to model the global linguistic information. Meanwhile, the origin local features of each character are not lost due to the shortcut of the transformer encoder; 2) BERT and VisionLAN can only mask one character in a forward process, it can only guide the model to learn linguistic knowledge, but GLRM can mask all characters in a parallel way, which can reconstruct the features of each character and obtain an enhanced feature.

3.4 Training Objective

The final objective function of the proposed method is formulated in Eq. 5. N is the number of cascade stages and $\mathbf{Y}^i$ is the prediction at the i^{th} stage. g_t is the ground truth. T is the max length of the character sequence which we set to 25 in our experiments.

$$L = -\frac{1}{NT} \sum_{i=0}^{N} \sum_{j=0}^{T} log(P(\mathbf{Y}^i | g_t)) \tag{5}$$

4 Experiment

4.1 Datasets

For fair comparison, we conduct experiments following the setup of [Wang *et al.*, 2022a; Fang *et al.*, 2021]. We use

Method	Regular			Irregular			AVG	#Params (M)	Speed (ms)
	IC13	SVT	IIIT5k	IC15	SVTP	CUTE			
CRNN [Shi *et al.*, 2016b]	91.9	81.6	82.9	69.4	70.0	65.5	78.6	8.3	-
ASTER [Shi *et al.*, 2018]	91.8	89.5	93.4	76.1	78.5	79.5	86.7	27.2	-
SEED [Qiao *et al.*, 2020]	92.8	89.6	93.8	80.0	81.4	83.6	88.3	-	-
RobustScanner [Yue *et al.*, 2020]	94.8	88.1	95.3	77.1	79.5	90.3	88.4	-	-
TextScanner [Wan *et al.*, 2020]	92.9	90.1	93.9	79.4	84.3	83.3	88.5	-	-
SRN [Yu *et al.*, 2020]	95.5	91.5	94.8	82.7	85.1	87.8	90.4	54.7	-
VisionLAN [Wang *et al.*, 2021]	95.7	91.7	95.8	83.7	86.0	88.5	91.2	32.8	21.73
ABINet [Fang *et al.*, 2021]	97.4	93.5	96.2	86.0	89.3	89.2	92.3	36.7	46.86
MGP-Small [Wang *et al.*, 2022a]	96.4	93.5	95.3	86.1	87.3	87.9	92.0	52.6	-
MGP-Base [Wang *et al.*, 2022a]	97.3	**94.7**	96.4	87.2	**91.0**	90.3	93.3	148.0	-
SVTR-Tiny [Du *et al.*, 2022]	96.3	91.6	94.4	84.1	85.4	88.2	90.8	6.03	4.11
SVTR-Small [Du *et al.*, 2022]	95.7	93.0	95.0	84.7	87.9	92.0	91.6	10.3	4.81
SVTR-Base [Du *et al.*, 2022]	97.1	91.5	96.0	85.2	89.9	91.7	92.3	24.6	5.80
LPV-Tiny (Ours)	96.7	92.9	96.3	86.4	86.7	90.6	92.5	8.11	5.17
LPV-Small (Ours)	96.8	93.7	96.7	87.1	89.8	92.4	93.3	13.99	5.77
LPV-Base (Ours)	**97.6**	94.6	**97.3**	**87.5**	90.9	**94.8**	**94.0**	35.13	7.41

Table 1: Results on IC13, SVT, IIIT5K, IC15, SVTP and CUTE datasets. Following [Fang *et al.*, 2021; Wang *et al.*, 2022a], all the results are under NONE lexicon. The speed is the inference time on one NVIDIA 2080Ti GPU averaged over 1000 English image text.

MJSynth [Jaderberg *et al.*, 2014; Jaderberg *et al.*, 2016] and SynthText [Gupta *et al.*, 2016] as training data and they contain 9M and 7M synthetic text images respectively. The performance is evaluated on 6 benchmarks containing IIIT 5K-Words (IIIT5K) [Mishra *et al.*, 2012], ICDAR2013 (IC13) [Karatzas *et al.*, 2013], ICDAR2015 (IC15) [Karatzas *et al.*, 2015], Street View Text (SVT) [Wang *et al.*, 2011], Street View Text-Perspective (SVTP) [Phan *et al.*, 2013] and CUTE80 (CUTE) [Risnumawan *et al.*, 2014]. Details of the above 6 datasets can be found in previous works [Wang *et al.*, 2022a; Fang *et al.*, 2021].

4.2 Implementation Details

We use the backbone proposed in SVTR [Du *et al.*, 2022] as our backbone due to its impressive performance in STR. Particularly, to use the attention-based decoder, we change the stride of the merging module to 1 and remove the final mixing head to obtain visual features at 1/4 resolution. The image size is set to 100 × 32. Following the most recent works [Wang *et al.*, 2022a; Baek *et al.*, 2019], for fair comparison, we use the same code framework and data augmentation. We conduct the experiments on 4 NVIDIA 3090 GPUs with batch size 384. The vocabulary size C of character classification head is set to 38, including 0 - 9, a - z, [PAD] for padding symbol and [EOS] for ending symbol.

The network is trained end-to-end using Adam [Kingma and Ba, 2014] optimizer of initial learning rate 1e-4 and the learning rate is decayed to 1e-5 after six epochs. We trained a total of 20 epochs. The first 10 epochs do not use the mask we proposed in GLRM so that the position attention can obtain a relatively accurate attention map. The last 10 epochs add the mask for finetune so that the network can learn contextual linguistic knowledge.

4.3 Comparisons with State-of-the-Arts

We compare our method with previous state-of-the-art methods on 6 benchmarks in Table. 1. Our model shows significant performance in both regular (IC13, SVT and IIIT5K) and irregular (IC15, SVTP and CUTE) datasets while keeping a very low parameter quantity. Notably, LPV-Tiny has already outperformed most of the state-of-the-art methods with only 8.11M parameters while LPV-Small and LPV-Base obtain the performance of 93.3% and 94.0% with only 13.99M and 35.13M parameters respectively. For inference time, LPV-Tiny only needs 5.17ms, which is faster than most of the existing methods.

Many previous works, such as SRN [Yu *et al.*, 2020], ABINet [Fang *et al.*, 2021], VisionLAN [Wang *et al.*, 2021], and MGP [Wang *et al.*, 2022a] tried to introduce linguistic knowledge to assist recognition. Compared to them, LPV shows the best performance on all datasets. This result implies that our cascade position attention mechanism and GLRM are effective. Additionally, compared with SVTR, our tiny, small, and base model obtains 1.6%, 1.7%, and 1.7% improvement respectively.

4.4 Ablation Study

The Effectiveness of Cascade Position Attention

We propose the Cascade Position Attention (CPA) mechanism to alleviate the linguistic insensitive drift problem. To prove the effectiveness of CPA, we perform ablation from two aspects.

From the aspect of model performance, we conduct several experiments to evaluate the effect of the number of stages N in Table. 2. Especially, $N = 1$ means no extra stage to optimize the recognition result. The first row of Table 2 is the baseline with a CTC-based decoder instead of attention-based decoder. From the statistics we can conclude: 1) Our

N	IC13 IC15	SVT SVTP	IIIT5k CUTE	AVG	Params (M)
-	96.3 84.1	91.6 85.4	94.4 88.2	90.9	6.03
1	96.616 84.705	91.808 86.512	95.733 90.625	91.336	4.39
2	96.033 85.864	92.581 **87.752**	95.833 90.625	92.150	6.25
3	**96.733** **86.361**	**92.890** 86.667	**96.300** **90.625**	**92.481**	8.11

Table 2: Ablation study of Cascade Position Attention (CPA) mechanism in LPV-Tiny, N is the number of stages in CPA. The first row is the baseline with a CTC-based decoder.

Str	Feat	Mask	IC13 IC15	SVT SVTP	IIIT5k CUTE	AVG
Tiny	D	✗	95.683 85.588	92.736 **87.597**	95.800 90.278	92.012
	D	✓	**96.733** **86.361**	**92.890** 86.667	**96.300** 90.625	**92.481**
	F	✗	95.683 85.919	91.499 86.512	95.433 **90.972**	91.763
Small	D	✗	96.383 86.582	93.045 88.217	96.533 89.583	92.701
	D	✓	**96.849** **87.134**	**93.663** **89.767**	**96.663** **92.361**	**93.240**

Table 3: Ablation study of GLRM, in the column of Feat, 'D' means dynamic feature in the sequence modeling and 'F' means fixed feature. Mask indicates if use the mask we peoposed.

L	IC13 IC15	SVT SVTP	IIIT5k CUTE	AVG	Params (M)
1	96.616 84.705	91.808 86.512	95.733 **90.625**	91.708	6.53
2	96.733 **86.361**	92.890 86.667	96.300 90.625	92.481	8.11
3	**97.083** 86.140	**92.890** **87.907**	**96.500** 89.931	**92.632**	9.69

Table 4: Ablation study of the layer number of GLRM. L is the layer number of GLRM. Experiments were performed on LPV-Tiny.

model with 3 stages outperforms 1.581% more improvement. 2) Due to the concise structure of the hierarchical optimization strategy, the increase of stages will result in great gains in average accuracy while little increase in parameter quantity. 3) As the number of stages increases gradually, performance improvement is limited. For the trade-off between parameter quantity and accuracy, we choose 3 stages in our model.

From the aspect of attention drift, we further visualize it. In the position attention mechanism, the query guides the decoder to find the position of each character. As described in Sec. 1, the high similarity of queries between neighboring locations leads to the problem of attention drift. Based on LPV-Tiny, we visualize the similarity between query vectors at different positions in Figure. 5. Due to stages 2 and 3 having a linguistic-sensitive query that is different when inputting different images, we calculate the average similarity with all images in IC15 [Karatzas *et al.*, 2015] of each sequence length. As shown in Figure 5, the query in the first stage has no position and linguistic prior about the input image so it does not have a centralized similarity. In stages 2 and 3, the queries consist of a linguistic prior query $\mathbf{Q}_{pri}$, and the position encoding is introduced again to enhance position sensitivity. Therefore, the similarity is centered in the diagonal, which means the position of each character is more certain. When decoding characters, the feature similarity of neighboring characters is reduced, so the attention drift is mitigated. Note that the similarity in stage 3 is more concentrated than that in stage 2 due to the stronger prior and more position information. Additionally, the difference is even more pronounced with long text, because attention drift is more likely to occur in the case of long text.

The Effectiveness of GLRM

As described in Sec. 3.3, we argue that the input features $\mathbf{F}^i$ of each stage can not be the same and needs to be dynamically adjusted, so a sequence modeling network is necessary. To prove this inference, we first use a simple transformer encoder as the sequence modeling network to obtain dynamic features. For fair comparison, we place the transformer encoder before the CPA decoder to fix the features and keep the same parameter quantity. As shown in Table. 3, the performance is not good (91.763% vs 92.012%) when we place the transformer layer before the CPA and input the same features into each stage.

Furthermore, to acquire linguistic knowledge, we propose GLRM as the sequence modeling network which uses a parallel mask to enhance the feature and obtain the contextual linguistic information. As shown in Table. 3, for LPV-Tiny and LPV-Small, the proposed mask obtains 0.469% and 0.539% improvement on average accuracy respectively.

The Layer Number of GLRM

Our GLRM consists of a Parallel Mask Generator and $L\times$ Masked Transformer Encoder. To determine the number of layers L, we conduct several experiments. From the results in Table. 4 we can observe: 1) More layers in GLRM can provide stronger contextual modeling capacity and obtain higher performance. 2) We can obtain 0.69% improvement when L increases from 1 to 2, which is greatly larger than 0.234% when L goes from 2 to 3. That is because the masked transformer encoder can only model the area around each character at a shallow layer, and gradually model the global feature as it moves deeper. This conjecture can be verified by the visualization of the attention map in the masked transformer encoder. We calculate the average attention map of the pixels in the area masked and show the visualization in Figure. 6. From the attention map, we can find that the attention in the first layer of the first GLRM is limited around each character

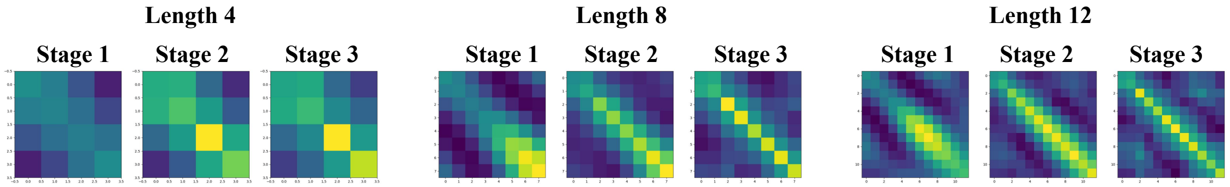

Figure 5: The visualization of the similarity between the queries at different positions in LPV-Tiny.

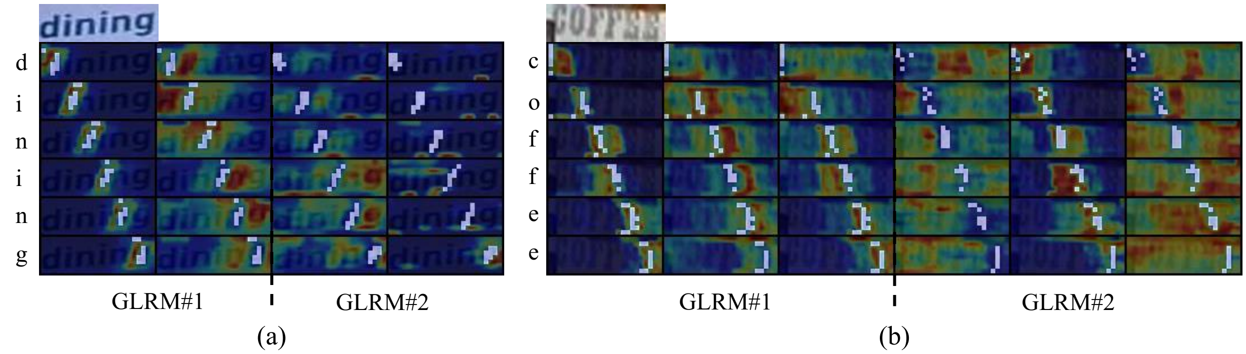

Figure 6: The average attention map of the pixels in the area masked and the area masked is shown in white. (a) LPV-Tiny with 2 layers in each GLRM. (b) LPV-Small with 3 layers in each GLRM.

because there is no global feature in the input. When it goes deep, the attention area goes global.

Finally, considering the total parameter quantity, we set L to 2 in LPV-Tiny and 3 in LPV-Small and LPV-Base.

4.5 The Qualitative Analysis

GLRM in Subword Perception

From the visualization in Figure. 6, we can further analyze the attention area. As we all know, there are some sub-words that occur frequently in words (e.g. 'ing', 'pri', 'mer', 'tion', etc). Such knowledge can assist the model to obtain a more accurate result when the visual clue is confused. Our GLRM guides the model to reconstruct the feature of each character using the feature of other characters so it will be sensitive to the sub-words. As shown in Figure 6 (a), the sub-words 'din' and 'ing' pay attention to themselves individually. This demonstrates the ability of our GLRM to learn contextual linguistic knowledge.

Lguistic Insensitive Drift Problem

Figure. 7 shows some sample cases of attention drift being corrected. For each input image, LPV can get three stages of recognition results: one preliminary result and two correction results. From the attention map, we can observe that if the 1^{st} stage gets a drift result, the remaining stages have the ability to correct benefiting from the linguistic-sensitive query in CPA.

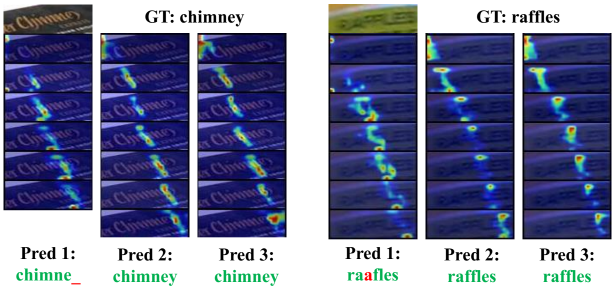

Figure 7: The visualization of attention map in each stage of CPA. Experiments were performed on LPV-Small. The model has three stages in CPA so there are three predict results for each input.

5 Conclusion

This paper first notices the Linguistic Insensitive Drift (LID) problem and analyzes the linguistic perception of the model. To find an efficient and accurate method, LPV is proposed to enhance the linguistic information of both query and feature (Linguistic More). To be specific, LPV introduces CPA to obtain an accurate attention map by using linguistic-sensitive query instead of visual query, and designs GLRM to aggregate the global linguistic information to enhance the visual feature. Compared with previous methods, our LPV is able to take a further step toward efficient and accurate recognition, which obtains dominant recognition performance while maintaining a concise pipeline. We believe that LPV will inspire recent works in simple network design and efficient linguistic perception, and we will further explore its potential in the future.

Acknowledgments

This work is supported by the National Key Research and Development Program of China (2022YFB3104700), the National Nature Science Foundation of China (62121002, 62022076, U1936210, 62232006).

References

[Baek *et al.*, 2019] Jeonghun Baek, Geewook Kim, Junyeop Lee, Sungrae Park, Dongyoon Han, Sangdoo Yun, Seong Joon Oh, and Hwalsuk Lee. What is wrong with scene text recognition model comparisons? dataset and model analysis. In *International Conference on Computer Vision (ICCV)*, 2019.

[Carion *et al.*, 2020] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.

[Cheng *et al.*, 2017] Zhanzhan Cheng, Fan Bai, Yunlu Xu, Gang Zheng, Shiliang Pu, and Shuigeng Zhou. Focusing attention: Towards accurate text recognition in natural images. In *Proceedings of the IEEE international conference on computer vision*, pages 5076–5084, 2017.

[Devlin *et al.*, 2018] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[Du *et al.*, 2022] Yongkun Du, Zhineng Chen, Caiyan Jia, Xiaoting Yin, Tianlun Zheng, Chenxia Li, Yuning Du, and Yu-Gang Jiang. Svtr: Scene text recognition with a single visual model. *arXiv preprint arXiv:2205.00159*, 2022.

[Fang *et al.*, 2021] Shancheng Fang, Hongtao Xie, Yuxin Wang, Zhendong Mao, and Yongdong Zhang. Read like humans: Autonomous, bidirectional and iterative language modeling for scene text recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7098–7107, 2021.

[Graves *et al.*, 2006] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pages 369–376, 2006.

[Gupta *et al.*, 2016] Ankush Gupta, Andrea Vedaldi, and Andrew Zisserman. Synthetic data for text localisation in natural images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2315–2324, 2016.

[Hu *et al.*, 2020] Wenyang Hu, Xiaocong Cai, Jun Hou, Shuai Yi, and Zhiping Lin. Gtc: Guided training of ctc towards efficient and accurate scene text recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11005–11012, 2020.

[Jaderberg *et al.*, 2014] Max Jaderberg, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Synthetic data and artificial neural networks for natural scene text recognition. *arXiv preprint arXiv:1406.2227*, 2014.

[Jaderberg *et al.*, 2016] Max Jaderberg, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Reading text in the wild with convolutional neural networks. *International journal of computer vision*, 116(1):1–20, 2016.

[Karatzas *et al.*, 2013] Dimosthenis Karatzas, Faisal Shafait, Seiichi Uchida, Masakazu Iwamura, Lluis Gomez i Bigorda, Sergi Robles Mestre, Joan Mas, David Fernandez Mota, Jon Almazan Almazan, and Lluis Pere De Las Heras. Icdar 2013 robust reading competition. In *2013 12th international conference on document analysis and recognition*, pages 1484–1493. IEEE, 2013.

[Karatzas *et al.*, 2015] 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.

[Kingma and Ba, 2014] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[Long *et al.*, 2021] Shangbang Long, Xin He, and Cong Yao. Scene text detection and recognition: The deep learning era. *International Journal of Computer Vision*, 129(1):161–184, 2021.

[Lyu *et al.*, 2019a] Pengyuan Lyu, Zhicheng Yang, Xinhang Leng, Xiaojun Wu, Ruiyu Li, and Xiaoyong Shen. 2d attentional irregular scene text recognizer. *arXiv preprint arXiv:1906.05708*, 2019.

[Lyu *et al.*, 2019b] Pengyuan Lyu, Zhicheng Yang, Xinhang Leng, Xiaojun Wu, Ruiyu Li, and Xiaoyong Shen. 2d attentional irregular scene text recognizer. *arXiv preprint arXiv:1906.05708*, 2019.

[Mishra *et al.*, 2012] Anand Mishra, Karteek Alahari, and CV Jawahar. Scene text recognition using higher order language priors. In *BMVC-British machine vision conference*. BMVA, 2012.

[Phan *et al.*, 2013] Trung Quy Phan, Palaiahnakote Shivakumara, Shangxuan Tian, and Chew Lim Tan. Recognizing text with perspective distortion in natural scenes. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 569–576, 2013.

[Qiao *et al.*, 2020] Zhi Qiao, Yu Zhou, Dongbao Yang, Yucan Zhou, and Weiping Wang. Seed: Semantics enhanced encoder-decoder framework for scene text recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13528–13537, 2020.

[Risnumawan *et al.*, 2014] Anhar Risnumawan, Palaiahankote Shivakumara, Chee Seng Chan, and Chew Lim Tan. A robust arbitrary text detection system for natural scene images. *Expert Systems with Applications*, 41(18):8027–8048, 2014.

[Sheng *et al.*, 2019] Fenfen Sheng, Zhineng Chen, and Bo Xu. Nrtr: A no-recurrence sequence-to-sequence model for scene text recognition. In *2019 International conference on document analysis and recognition (ICDAR)*, pages 781–786. IEEE, 2019.

[Shi *et al.*, 2016a] Baoguang Shi, Xiang Bai, and Cong Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE transactions on pattern analysis and machine intelligence*, 39(11):2298–2304, 2016.

[Shi *et al.*, 2016b] Baoguang Shi, Xiang Bai, and Cong Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE transactions on pattern analysis and machine intelligence*, 39(11):2298–2304, 2016.

[Shi *et al.*, 2018] Baoguang Shi, Mingkun Yang, Xinggang Wang, Pengyuan Lyu, Cong Yao, and Xiang Bai. Aster: An attentional scene text recognizer with flexible rectification. *IEEE transactions on pattern analysis and machine intelligence*, 41(9):2035–2048, 2018.

[Wan *et al.*, 2020] Zhaoyi Wan, Minghang He, Haoran Chen, Xiang Bai, and Cong Yao. Textscanner: Reading characters in order for robust scene text recognition. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 12120–12127, 2020.

[Wang *et al.*, 2011] Kai Wang, Boris Babenko, and Serge Belongie. End-to-end scene text recognition. In *2011 International conference on computer vision*, pages 1457–1464. IEEE, 2011.

[Wang *et al.*, 2021] Yuxin Wang, Hongtao Xie, Shancheng Fang, Jing Wang, Shenggao Zhu, and Yongdong Zhang. From two to one: A new scene text recognizer with visual language modeling network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14194–14203, 2021.

[Wang *et al.*, 2022a] 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.

[Wang *et al.*, 2022b] Yuxin Wang, Hongtao Xie, Shancheng Fang, Mengting Xing, Jing Wang, Shenggao Zhu, and Yongdong Zhang. Petr: Rethinking the capability of transformer-based language model in scene text recognition. *IEEE Transactions on Image Processing*, 31:5585–5598, 2022.

[Xu *et al.*, 2022] Jianjun Xu, Hongtao Xie, Hai Xu, Yuxin Wang, Sun-ao Liu, and Yongdong Zhang. Boat in the sky: Background decoupling and object-aware pooling for weakly supervised semantic segmentation. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 5783–5792, 2022.

[Yu *et al.*, 2020] Deli Yu, Xuan Li, Chengquan Zhang, Tao Liu, Junyu Han, Jingtuo Liu, and Errui Ding. Towards accurate scene text recognition with semantic reasoning networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12113–12122, 2020.

[Yue *et al.*, 2020] Xiaoyu Yue, Zhanghui Kuang, Chenhao Lin, Hongbin Sun, and Wayne Zhang. Robustscanner: Dynamically enhancing positional clues for robust text recognition. In *European Conference on Computer Vision*, pages 135–151. Springer, 2020.

[Zheng *et al.*, 2021] Tianlun Zheng, Zhineng Chen, Shancheng Fang, Hongtao Xie, and Yu-Gang Jiang. Cdistnet: Perceiving multi-domain character distance for robust text recognition. *arXiv preprint arXiv:2111.11011*, 2021.