Parallel and Robust Text Rectifier for Scene Text Recognition

Bingcong Li[1,†], Xin Tang[1,†], Jun Wang[2,†], Liang Diao[1], Rui Fang[1], Guotong Xie[2], Weifu Chen[3,*]

[1] Visual Computing Group, Ping An Property & Casualty Insurance Company, Shenzhen, China
[1] Ping An Technology (Shenzhen) Co. Ltd., Shenzhen, China
[3] School of Information and Telecommunication Engineering, Guangzhou Maritime University, Guangzhou, China

Introduction

Problems: Scene text recognition (STR) is to recognize text appearing in images. Current state-of-the-art STR methods usually adopt a rectifier to iteratively rectify errors from previous stage, which are not proficient in addressing misalignment problems, see Fig.1.

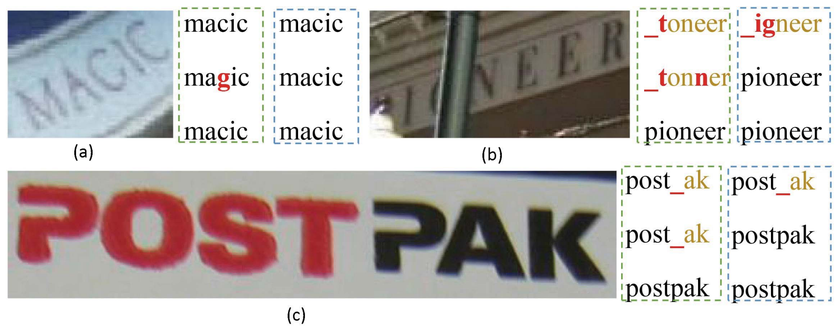

Figure 1: Three examples with intermediate predictions (a)(b)(c). Texts in green dash boxes and blue dash boxes represent the results of ABINet [**?**] and PRTR (ours). In each box, the first row is the initial prediction, the second row is the prediction of the rectifier, and the last one is the ground-truth. "_" is a placeholder, characters in red represent aligned errors, and characters in yellow correspond to misaligned errors.

Motivation: Given an input image **I**, we aim at predicting the character sequence $Y = \{y_1, y_2, \cdots, y_T\}$ in the image by maximizing $p(Y|\mathbf{I})$. Motivated by the observation in physiology that humans progressively improve prediction confidence by iteratively correcting the recognition results, we sequentially predict $S(S \geq 1)$ character sequences $\{Y^{(s)}\}_{s=0}^{S}$ to progressively approximate the ground-truth sequence Y^* by

$$p(Y|\mathbf{I}) = p(Y^{(S)}|Y^{(S-1)}, \mathbf{I}) \cdots p(Y^{(1)}|Y^{(0)}, \mathbf{I}) p(Y^{(0)}|\mathbf{I}), \tag{1}$$

Methods

Architecture of the proposed framework

PRTR consists of three parts: a visual feature extraction, an initial decoder and a robust visual-semantic rectifier (RVSR).

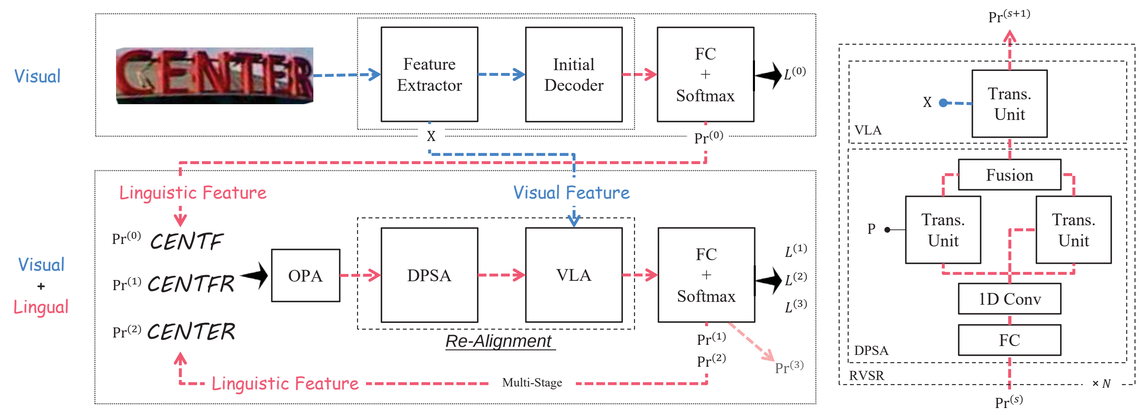

Figure 2: Overall architecture of PRTR. OPA, DPSA and VLA denote online probability augmentation, dual-path semantic alignment and visual-linguistic alignment respectively. The right part demonstrates the structure of DPSA and VLA.

Visual Feature Extraction

The visual features **X** of image **I** can be obtained as:

$$\mathbf{X} = \mathcal{R}(\mathcal{T}(\mathcal{C}(\mathbf{I}))) \in \mathbb{R}^{N \times D}, \tag{2}$$

where $\mathcal{C}$ is a ResNet31 with multi-aspect global context attention, $\mathcal{T}$ is an transformer spacial encoder, $\mathcal{R}$ represents the reshape operator. D denotes the dimension of the features, $N = \frac{H}{8} \times \frac{W}{4}$, H and W are the height and the width of **I**.

Initial Decoder

The initial decoder in Eq. (1) can be written as $p(Y^{(0)}|\mathbf{I}) = \prod_{t=1}^{T} p(y_t|\mathbf{X})$ that is visually based and the visual features are obtained by position transformer

$$\mathbf{V} = \text{softmax}(\frac{\mathbf{Q}\mathcal{U}(\mathbf{X})^T}{\sqrt{D}})\mathbf{X}, \tag{3}$$

Then the output probability over characters at time step t is computed by $\mathbf{Pr}_t^{(0)} = \text{Softmax}(\mathbf{W}^{(0)}\mathbf{v}_t + \mathbf{b}^{(0)})$, $\mathbf{v}_t$ is computed by Eq. (3).

Robust Visual-Semantic Rectifier (RVSR)

We use RVSR to rectify the initial decoder $\mathbf{Pr}_t^{(0)}$. At Stage s, the refined prediction $Y^{(s)}$ is modeled as

$$\begin{aligned} p(Y^{(s)}|Y^{(s-1)}, \mathbf{I}) &= \prod_{t=1}^{T} p(y_t^{(s)}|y_1^{(s-1)}, y_2^{(s-1)}, \cdots, y_T^{(s-1)}, \mathbf{I}) \\ &\approx \prod_{t=1}^{T} p(y_t^{(s)}|\mathcal{F}(\mathbf{Pr}_1^{(s-1)}), \cdots, \mathcal{F}(\mathbf{Pr}_T^{(s-1)}), \mathbf{X}), \end{aligned} \tag{4}$$

$\mathcal{F}$ is the fully connected layer. RVSR employs a dual-path semantic alignment (DPSA) module and a visual-lingual alignment (VLA) module to refine the prediction with linguistic knowledge $\mathbf{e}_t^{(s-1)} = \mathcal{F}(\mathbf{Pr}_1^{(s-1)})$

Methods (cont.)

Dual-Path Semantic Alignment

On one path, applies 1D convolution and multi-head transformer to rectify $\{\mathbf{e}_t^{(s-1)}\}$

$$\mathbf{e}_t^{(s)} = \sum_{i=1}^{k} \mathbf{w}_i \cdot \mathbf{e}_{t-\lfloor \frac{k}{2} \rfloor + i}^{(s-1)}. \tag{5}$$

$$\hat{\mathbf{e}}_t^{(s)} = \alpha_t v_t + \sum_{j \neq t} \alpha_j v_j, \tag{6}$$

where $\alpha_j = \exp(q_t^T k_j) / \sum_{i=1}^{T} \exp(q_t^T k_i)$ and $q_t = \mathbf{W}_q \mathbf{e}_t^{(s)}$, $k_t = \mathbf{W}_k \mathbf{e}_t^{(s)}, v_t = \mathbf{W}_v \mathbf{e}_t^{(s)}$.

On another path, semantic aligning is imposed on $\{\mathbf{p}_t\}$ and $\{\mathbf{e}_t^{(s)}\}$ by using multi-head transformer

$$\widetilde{\mathbf{E}}_t^{(s)} = \text{FFN}(\text{MultiHead}(\mathbf{P}, \mathbf{E}^{(s)}, \mathbf{E}^{(s)})). \tag{7}$$

Combining the outputs of the two paths

$$\begin{aligned} \mathbf{e}_\mathbf{t}^{(\mathbf{s})} &= \mathbf{G} \odot \hat{\mathbf{e}}_t^{(s)} + (1 - \mathbf{G}) \odot \widetilde{\mathbf{e}}_t^{(s)} \\ \mathbf{G} &= \sigma(\mathbf{W}_g \, \text{Concat}(\hat{\mathbf{e}}_t^{(s)}, \widetilde{\mathbf{e}}_t^{(s)})) \end{aligned}, \tag{8}$$

Visual-Linguistic Alignment

Apply an attention model on visual features **X** and linguistic features $\mathbf{e}_\mathbf{t}^{(\mathbf{s})}$ to rectify fusion misalignment error

$$\mathbf{e}_t^{(s)} = \text{FFN}(\text{MultiHead}(\mathbf{e}_t^{(s)}, \mathbf{X}, \mathbf{X})), \tag{9}$$

Other training techniques such as Online Probability Augmentation (OPA) have been adopted in the training scheme.

Training Objective

PRTR is trained in an end-to-end manner by minimizing the objective function

$$L = -\lambda_0 \sum_{t=1}^{T} y_t^* \log p(\mathbf{Pr}_t^{(0)}) - \sum_{s=1}^{S} \lambda_s \sum_{t=1}^{T} y_t^* \log p(\mathbf{Pr}_t^{(s)}|\mathbf{Pr}_t^{(s-1)}) \tag{10}$$

where $\{y_1^*, y_2^*, \cdots, y_T^*\}$ is the ground-truth label, λ_0, λ_s and S are hyper-parameters.

Experiments

PRTR was implemented using Pytorch on 4 Tesla V100 GPUs with 16G memory. The hyper-parameters S, λ_0, λ_1, λ_2, λ_3 were set to 3, 0.2, 0.2, 0.2, 0.8.

Ablation study

DPSA	VLA	OPA	IIIT	SVT	IC03	IC13	IC15	SVTP	CUTE	Avg
			96.3	92.3	95.2	94.8	84.7	87.0	93.8	92.3
✓			95.6	94.0	96.8	96.2	85.6	90.4	94.1	93.0
✓		✓	96.2	94.6	97.0	95.9	84.9	90.2	94.8	93.1
✓	✓		96.8	93.5	96.0	96.0	85.0	89.5	95.5	93.1
✓	✓	✓	97.0	94.4	96.4	95.8	86.1	89.8	96.5	93.6

Table 1: Ablation study of RVSR. We design a serial of experiments to validate the impact of each component.

Comparison with State-of-the-Art Models

Method	Regular test dataset				Irregular test dataset		
	IIIT	SVT	IC03	IC13	IC15	SVTP	CUTE
ASTER [22]	93.4	89.5	94.5	91.8	76.1	78.5	79.5
SAR [11]	91.5	84.5	_	91.0	69.2	76.4	83.3
DAN [25]	94.3	89.2	95.0	93.9	74.5	80.0	84.4
SRN [27]	94.8	91.5	_	95.5	82.7	85.1	87.8
SCATTER [12]	93.7	92.7	96.3	93.9	82.2	86.9	87.5
RobustScanner [28]	95.3	88.1	_	94.8	77.1	79.5	90.3
GTC [5]	95.5	92.9	95.2	94.3	82.5	86.2	92.3
JVSR [1]	95.2	92.2	_	95.5	84.0	85.7	89.7
PREN [26]	95.6	94.0	95.8	96.4	83.0	87.6	91.7
ABINet-SV [2]	95.4	93.2	_	96.8	84.0	87.0	88.9
ABINet-LV [2]	96.2	93.5	_	**97.4**	86.0	89.3	89.2
PRTR (OURS)	**97.0**	94.4	**96.4**	95.8	**86.1**	**89.8**	**96.5**
PRTR WO\SA (OURS)	96.9	**94.6**	96.1	95.5	85.7	88.8	96.2

Table 2: Performance Comparison on seven benchmarks. **Bold** and underline represents the best and the second best performance. The **PRTR** row shows the results of PRTR trained only MJ and ST. The **PRTR** WO\SA row shows the results removing the usage of SynthAdd.