Visual-Semantic Transformer for Scene Text Recognition

Liang Diao[1,†], Xin Tang[1,†], Jun Wang[2,†], 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: Semantic information is as important as visual information in scene text recognition (STR). However, many state-of-the-art models have processed these two kinds of information independently or need extra external language models to refine semantic features through context information, which reduces the efficiency and the performance of the models.

Motivation: Motivated by the fact that our cognitive system detects and interprets information from the optical spectrum of an image to build a representation, the proposed model employs two core components, visual-semantic alignment modules and visual-semantic interaction modules to learn and fuse visual features with semantic features. The visual-semantic modules are able to learn the semantic features from visual features without explicitly using linguistic features or external language models. The visual-semantic interaction modules make semantic and visual features extracted at different stages interact with each other. Fig. 1 demonstrates the architecture of the proposed model.

Methodology

Visual-Semantic Transformer (VST) consists of several key modules, namely ConvNet ($\mathcal{C}$), Visual module ($\mathcal{V}$), Interaction module ($\mathcal{I}$), Semantic module ($\mathcal{S}$) and two Visual-Semantic Alignment ($\mathcal{A}_1$ and $\mathcal{A}_2$) modules.

The ConvNet ($\mathcal{C}$) Module: can be any kind of convolutional networks. In this work, we use a resnet-like architecture.

The Visual ($\mathcal{V}$) Module: a transformer-based visual module that is added after Module $\mathcal{C}$ to learn the global correlation and enhance the feature maps. The output is called *primary visual features*.

The Visual-Semantic Alignment ($\mathcal{A}$) Module: adopts a simple parallel attention to learn semantic features from visual features. The visual features are first projected using a linear matrix Q and normalized using softmax, obtaining t attention maps, each of which has the same spatial dimension as the origin feature maps. The visual features are weightily summed by each heat map to obtain t *semantic features*

$$S = \text{softmax}(QV^T)V. \tag{1}$$

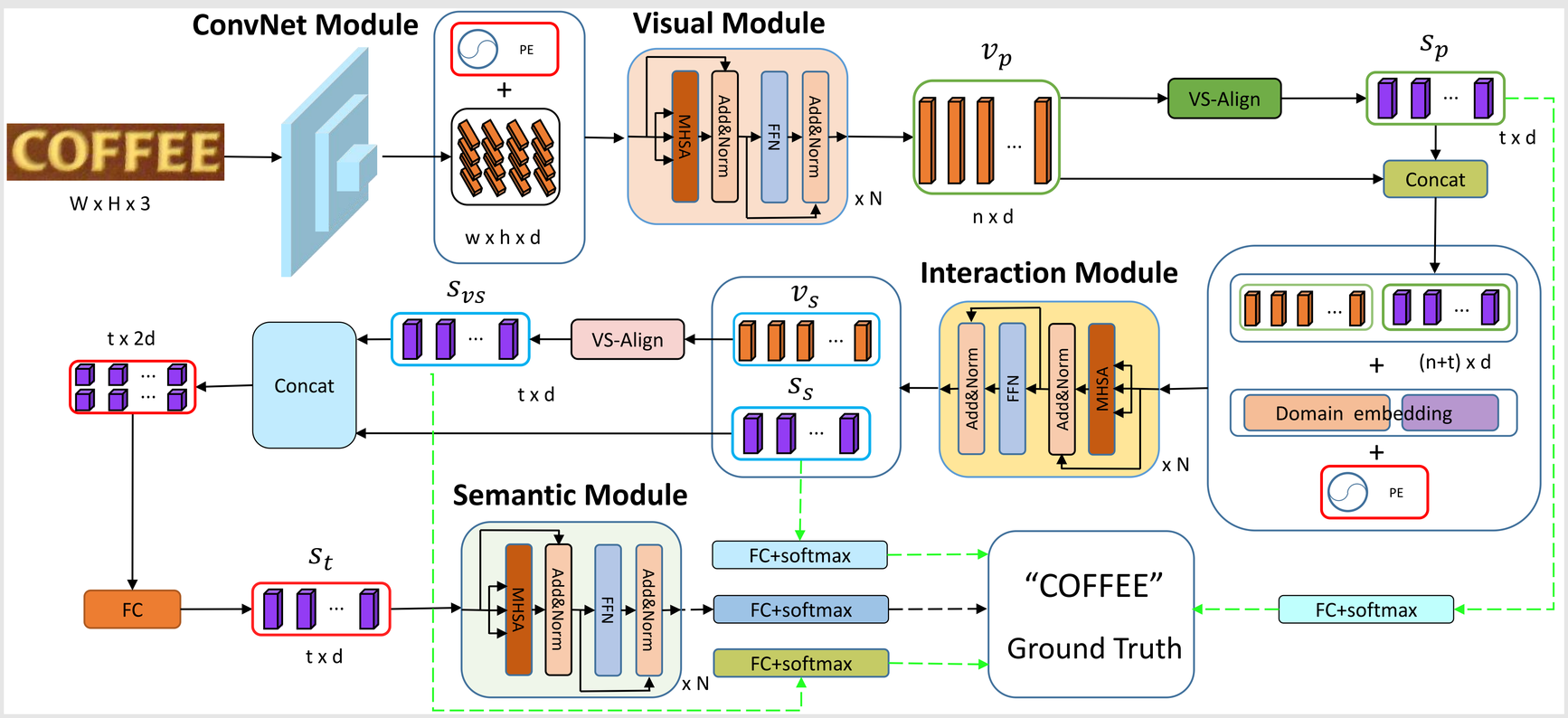

Figure 1: The architecture of Visual-Semantic Transformer (VST). The VST consists of several key modules, namely ConvNet ($\mathcal{C}$), Visual module ($\mathcal{V}$), Interaction module ($\mathcal{I}$), Semantic module ($\mathcal{S}$) and two Visual-Semantic Alignment ($\mathcal{A}_1$ in green, $\mathcal{A}_2$ in pink) modules. v_p, s_p, v_s, s_s, s_{vs}, s_t represents primary visual, primary semantic, secondary visual, secondary semantic, visually semantic and tertiary semantic features respectively. Best view in color.

Methodology (cont.)

The Interaction ($\mathcal{I}$) Module: takes *the primary semantic features* and *the primary visual features* as input, and produce *the secondary visual features* and *the secondary semantic features*. Module $\mathcal{I}$ is a transformer-based model, like in [36], and can be formulated as,

$$S = \text{softmax}(\frac{[Q_s; Q_v][K_s; K_v]^T}{\sqrt{d_k}})[V_s; V_v], \tag{2}$$

where s and v denote the semantic and the visual features.

The Semantic ($\mathcal{S}$) Module: is used to further fuse the secondary semantic features and the visually semantic features, with the same architecture as Module $\mathcal{V}$, and output the *tertiary semantic features*. Module $\mathcal{S}$ is optional. When it is inserted, the model is named VST-F (full); otherwise, VST-B (basic).

Training Objective: The loss of the proposed model is defined by the cross-entropy loss calculated between the ground truth and the predictions given by intermediate semantic features. Please see Sec. 2.7 for the details.

Experiments

Ablation study

Module	Regular test datasets				Irregular test datasets		
	IIIT	SVT	IC03	IC13	IC15	SVTP	CUTE
$\mathcal{CV}$	95.10	91.94	95.72	95.52	81.83	86.23	91.33
$\mathcal{CVA}_1$	95.63	91.94	96.41	95.55	82.31	86.95	91.76
$\mathcal{CVA}_1\mathcal{IA}_2/S_s$	96.57	93.82	97.35	96.55	85.20	88.37	94.79
$\mathcal{CVA}_1\mathcal{IA}_2/S_{vs}$	96.67	93.82	**97.46**	96.55	85.15	88.37	95.49
$\mathcal{CVA}_1\mathcal{IA}_2\mathcal{S}$	**96.73**	**94.00**	97.34	**96.65**	**85.42**	**89.00**	**95.49**

Table 1: Recognition accuracy increases when extra module is added. $/S_s$ and $/S_{vs}$ denote decoding from secondary semantics and visually semantics respectively, $\mathcal{CVA}_1\mathcal{IA}_2\mathcal{S}$ means modules $\mathcal{C}$+$\mathcal{V}$+$\mathcal{A}_1$+I+$\mathcal{A}_2$+$\mathcal{S}$.

Experiments (cont.)

Comparison with state-of-the-art models

Method	Regular test datasets				Irregular test datasets		
	IIIT	SVT	IC03	IC13	IC15	SVTP	CUTE
AON [6]	87.0	82.8	91.5	_	68.2	73.0	76.8
ASTER [35]	93.4	89.5	94.5	91.8	76.1	78.5	79.5
NRTR [33]	86.5	88.3	95.4	94.7	_	_	_
SAR [17]	91.5	84.5		91.0	69.2	76.4	83.3
DAN [41]	94.3	89.2	95.0	93.9	74.5	80.0	84.4
HRGAT [46]	94.7	88.9	_	93.2	79.5	80.9	85.4
SRN [47]	94.8	91.5	_	95.5	82.7	85.1	87.8
SCATTER [19]	93.7	92.7	96.3	93.9	82.2	86.9	87.5
GTC [10]	95.5	92.9	95.2	94.3	82.5	86.2	92.3
RobustScanner [49]	95.3	88.1	_	94.8	77.1	79.5	90.3
JVSR [4]	95.2	92.2	_	95.5	84.0	85.7	89.7
PREN [45]	95.6	94.0	95.8	96.4	83.0	87.6	91.7
ABINet-SV [8]	95.4	93.2	_	96.8	84.0	87.0	88.9
ABINet-LV [8]	96.2	93.5	_	**97.4**	**86.0**	**89.3**	89.2
VST-B (OURS)	**96.7**	93.8	**97.5**	96.6	85.2	88.4	**95.5**
VST-F (OURS)	**96.7**	**94.0**	97.3	96.7	85.4	89.0	**95.5**

Table 2: Comparison results with State-of-the-art models. '-' denotes data not available or config not the same.

Visualization of attention maps

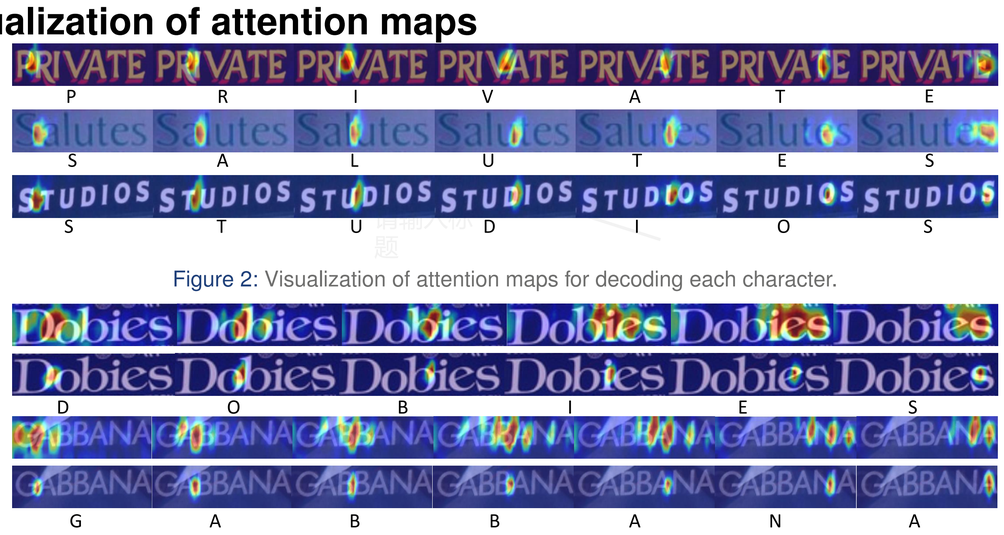

Figure 2: Visualization of attention maps for decoding each character.

Figure 3: Visualization of attention heatmaps of the primary and secondary VS-Align modules. For each of the two examples, the top (bottom) row shows the heatmaps of the primary (secondary) VS-Align module.