arXiv:2403.19128v1 [cs.CV] 28 Mar 2024

OmniParser: A Unified Framework for Text Spotting, Key Information Extraction and Table Recognition

Jianqiang Wan[1*] Sibo Song[1*] Wenwen Yu[2*] Yuliang Liu[2✉] Wenqing Cheng[2]
Fei Huang[1] Xiang Bai[2] Cong Yao[1] Zhibo Yang[1✉]
[1]Alibaba Group [2]Huazhong University of Science and Technology
{hustwjq,sibosongzju,yangzhibo450,yaocong2010}@gmail.com
f.huang@alibaba-inc.com {wenwenyu,ylliu,xbai,chengwq}@hust.edu.cn

Abstract

*Recently, visually-situated text parsing (VsTP) has experienced notable advancements, driven by the increasing demand for automated document understanding and the emergence of Generative Large Language Models (LLMs) capable of processing document-based questions. Various methods have been proposed to address the challenging problem of VsTP. However, due to the diversified targets and heterogeneous schemas, previous works usually design task-specific architectures and objectives for individual tasks, which inadvertently leads to modal isolation and complex workflow. In this paper, we propose a unified paradigm for parsing visually-situated text across diverse scenarios. Specifically, we devise a universal model, called OmniParser, which can simultaneously handle three typical visually-situated text parsing tasks: text spotting, key information extraction, and table recognition. In OmniParser, all tasks share the unified encoder-decoder architecture, the unified objective: **point-conditioned text generation**, and the unified input&output representation: **prompt & structured sequences**. Extensive experiments demonstrate that the proposed OmniParser achieves state-of-the-art (SOTA) or highly competitive performances on 7 datasets for the three visually-situated text parsing tasks, despite its unified, concise design. The code is available at AdvancedLiterateMachinery.*

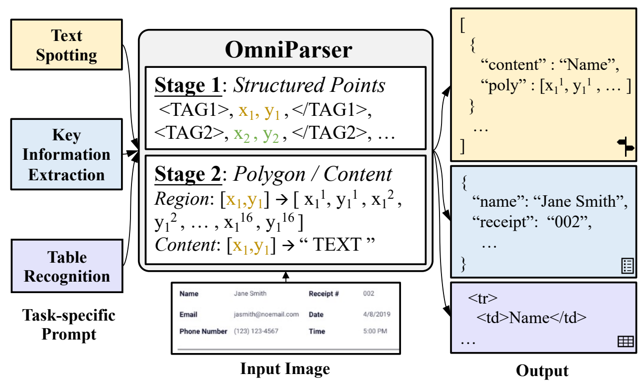

Figure 1. **A task-agnostic architecture for visually-situated text parsing.** The proposed OmniParser takes an image and a task-specific indicator as input and generates structured text sequences tailored to the specified task, including text spotting, key information extraction, and table recognition.

1. Introduction

Visually-situated text parsing (VsTP) is designed to extract structured information from document images. It involves the spotting and parsing of textual and visual elements within the text-rich image, such as text, tables, graphics, and other visual entities, partly shown in Fig. 1. With the rapid growth in the volume of text-related data and the enormous advance in Large Language Models [58, 59] and Multi-modal Large Language Models [60], there has been recently a surge of research on the topic of VsTP [7, 37, 39, 89]. These methods can be further categorized into generalist models [7, 37] and specialist models [49, 76, 89].

Both generalist models and specialist models have limitations in handling multiple multimodal tasks that are closely interconnected in the domain of VsTP. Generalist models excel in their versatility and universality across domains, but fall short in achieving high precision and interpretability. The performances will be restricted if an external OCR engine is not available [7]. Moreover, the prediction processes of such models are usually non-transparent, due to their black-box nature. Regarding specialist models, they frequently achieve higher performance in their respective sub-tasks [49, 89]. However, when confronted with the requirement of multi-tasking, the pipeline will be usually more complex. Furthermore, discrete specialist models inadvertently lead to modal isolation and limit in-depth understanding.

In recent years, there has been a trend towards unified models capable of performing multiple visually-situated text

*Equal contribution. ✉Corresponding authors.

parsing tasks, as illustrated in Tab. 1. While these models have shown effectiveness, handling diverse text structures and various relations in VsTP remains challenging. Accordingly, tasks in visual document parsing can be categorized into: 1) Sequential text detection and recognition, 2) Table structure and content recognition, and 3) Visual entity extraction and localization. Addressing these diversities while maintaining superior performance in a unified framework poses several challenges. First, incorporating task-specific heads [89], adapters [40, 46], and formulations [30, 49] can hinder achieving generality. Second, handling cross-dependencies between tasks is crucial, for instance, table recognition encompasses text spotting. Third, the unified representation of tasks should consider both primary elements (*words, points, lines, cells*) and various types of relations (*the adjacency between characters, the linking between keys and values, and the alignment of table cells.*).

Along with this line of works, we propose a unified paradigm for visually-situated text parsing in this paper (named ***OmniParser***). By adopting a single architecture, standardizing modeling objective as well as output representation, OMNIPARSER seamlessly handles text spotting, key information extraction (KIE), and table recognition (TR) in a unified framework, as shown in Fig. 1. To boost performance and increase transparency, we adopt a two-stage generation strategy. In the first stage, a structured sequence consisting of center points of text segments and task-related structural tokens is generated, given the embeddings of the input image and task prompt. In the second stage, polygonal contour and recognition results are predicted for each center point.

The philosophy behind the two-stage design is straightforward. The first stage produces center point sequences which can represent word-level/line-level text instances with complex structures encoded in various markup languages, e.g., JSON or HTML. The second stage can uniformly generate polygonal contours and recognition results across different tasks. An obvious advantage of our two-stage strategy is that the explicit decoupling could greatly reduce the difficulty of learning structured sequences, since the sequence lengths are significantly reduced. As such, higher performance and better generalization ability could be achieved.

To summarize, our major contributions are as follows:

- We propose OMNIPARSER, a unified framework for visually-situated text parsing. To the best of our knowledge, this is the first work that can simultaneously handle text spotting, key information extraction, and table recognition with a single, unified model.
- We introduce a two-stage decoder that leverages structured points sequences as an adapter, which not only enhances the parsing capability for structural information, but also provides better interpretability.
- We devise two pre-training strategies, namely spatial-aware prompting and content-aware prompting, which enable a powerful Structured Points Decoder for learning complex structures and relations in VsTP.
- Experiments on standard benchmarks demonstrate that the proposed OMNIPARSER outperforms the existing unified models on the three tasks. Meanwhile, it compares favorably with models with task-specific customization.

Methods	Visually-situated Text Parsing		
	Text Spotting	KIE	Table Recognition
Donut [30]	×	E2E, w/o Loc.	×
BROS [21]	×	OCR-dependent	TSR
DocReL [39]	×	OCR-dependent	TSR
UniDoc [14]	✓	E2E, w/o Loc.	×
SeRum [4]	✓	E2E, w/o Loc.	×
OMNIPARSER	✓	E2E	E2E (TSR + TCR)

Table 1. **Comparing the parsing capabilities achieved by different unified paradigms.** 'TSR' and 'TCR' denote Table Structure Recognition and Table Content Recognition respectively. To the best of our knowledge, OMNIPARSER is the first paradigm that accomplishes end-to-end visually-situated text parsing for text spotting, key information extraction, and table recognition.

2. Related Work

Scene Text Spotting. Text spotting aims to simultaneously detect and recognize all the texts in an image. Early end-to-end spotting methods [15, 20, 36, 44, 72], connected detection and recognition through customized ROI operations, which were not well-suited for curved text. Some segmentation-based methods [40, 54, 65, 66] can handle arbitrary-shaped text, but the post-processing and smoothing operations of the segmentation map are not trivial. Recently, transformer-based methods have achieved greater progress with their simple and efficient structures. TESTR [94] utilizes two similar decoders to obtain detection and recognition results separately, while DeepSolo [89] models text semantics and positions explicitly through learnable point queries. However, query-based spotting methods are often limited by the maximum number of detectable texts. Some autoregressive spotting methods can better deal with a large number of texts, such as UNITS [29], which outputs text sequences using start point prompts until the end. The SPTS series [47, 62] represent texts with corresponding center points but lack the ability to localize text precisely.

Key Information Extraction. Existing KIE approaches can be roughly separated into two categories: OCR-dependent models and OCR-free models. Early research efforts focus on building layout-aware or graph-based representation for KIE via sequence labeling with OCR inputs [1, 9, 17, 18, 23, 34, 35, 38, 41, 52, 63, 68, 84–86, 90, 98]. However, most of these methods rely on text with proper reading order or extra modules [80, 92] for OCR serialization, which is not practical in real-world scenarios. To address the serialization issue, other methods [21, 26, 52, 81, 85, 87, 91, 92] leverage extra detection modules or linking modules for modeling complex relations of text blocks or tokens. Although these methods employ extra links or modules to solve the reading order issue, the complicated decoding or post-processing strategy limits their generalization ability. Beyond that, generation-based methods [3, 5, 74] are proposed to alleviate the burden of post-processing and task-specific link designs. Another category of OCR-free methods employ OCR-aware pre-training or extends with OCR modules in an end-to-end fashion. Donut and other Seq2Seq-like methods [4, 10, 12, 30] adopt a text reading pre-training objective and generate structured outputs consisting of text and entity tokens. By explicitly equipping text reading modules, previous work [33, 73, 76, 91, 93] can achieve end-to-end key information extraction with task-specific design.

Table Recognition. Recent advances in vision-based approaches have improved table extraction from documents, traditionally divided into table detection, table structure recognition (TSR), and table content recognition (TCR). While table detection [71, 96] is beyond our scope, TSR, recently adopting an encoder-decoder fashion [56, 88], focuses on identifying table structures. TCR involves recognizing text within table cells using established OCR models. Our paper focuses on table recognition (TR), integrating TSR and TCR. TR methods fall into non-end-to-end [19, 24, 43, 55] and end-to-end [53, 97] categories. Non-end-to-end methods recover table structure with a specific model and employ offline OCR models for complete HTML sequences. Note that end-to-end table recognition tasks remain less explored due to their complexity and challenging nature.

Unified Frameworks. We are witnessing a clear trend in building unified frameworks for text-rich image parsing tasks. Prior arts such as DocReL [39] and BROS [21] model relations between table cells or entities through binary classification or a relational matrix, which also requires an off-the-shelf OCR engine. StrucTexTv2 [91] proposes a multi-modal learning framework aiming at document image understanding tasks by constructing self-supervised tasks. However, it relies on several task-specific lightweight designs for downstream tasks, such as Cascade R-CNN for table cell detection. Another example, HierText [50] pursues unifying scene text detection and layout analysis through an affinity matrix for modeling grouping relations. Additionally, SeRum [4] converts the end-to-end KIE task into a local decoding process and then shows its effectiveness on text spotting task.

In this work, we propose OmniParser that is capable of executing a variety of visually-situated parsing tasks in an end-to-end manner. These tasks encompass text spotting, key information extraction, and table recognition, all of which are consolidated within a unified framework. OmniParser is able to represent the heterogeneous structures of text in natural scenes or document images by decoupling structured points with text regions and contents. This bifurcated approach caters to the intrinsic characteristics of text-rich images where the text instances can be parsed concurrently, thereby facilitating an enhancement in universality.

3. Methodology

3.1. Task Unification

As shown in Fig. 2, we propose a new unified interface that represents structured sequences with three sub-sequences across diverse tasks. Points are employed as bridges to effectively link structural tags with region and content sequences.

Structured Points Sequence Construction comprises center points tokens as well as a variety of structural tokens designed for different tasks. The x and y coordinates of each point are first normalized to the width and height of the image, respectively. Subsequently, they are quantized into discrete tokens within the range of $[0, n_{bins} - 1]$. Moreover, structural tokens are introduced to represent the entire sequence, such as <address> in KIE task and <tr> in table recognition task. Note that text spotting can be seen as a special case that no structural token is incorporated.

Polygon & Content Sequence Construction is consistent across all tasks. We adopt 16-point polygonal formats to represent the polygonal contour for each text instance. Each point in the polygon sequence is tokenized following the same procedure as the center point tokenization. Besides, the transcription of text instances is converted into discrete tokens through char-level tokenization.

3.2. Unified Architecture

In light of our overarching goal to enhance the general-purpose paradigm for parsing text-rich images, we utilize a straightforward framework to assess the effectiveness of our proposed representation. To this end, we propose an encoder-decoder architecture that effectively addresses a wide range of visual text parsing tasks, as depicted in Fig. 2.

Image Encoder. We adopt the Swin-B [48] pre-trained on ImageNet 22k dataset as the fundamental visual feature extractor. Specifically, given an image $\mathbf{I} \in \mathbb{R}^{H \times W \times 3}$, we first use the image encoder to extract block-wise visual features which have strides of 4, 8, 16, 32 with respect to the input image. Afterward, we employ FPN [42] for feature fusion in order to better capture text features at various scales, following [70]. Formally, a set of visual embeddings $\{\mathbf{v}_i \mid \mathbf{v}_i \in \mathbb{R}^d, 1 \leq i \leq n\}$ is generated, where n is feature map size after FPN and d is the dimension of the latent embeddings of the decoders.

Decoders. Structured Points Decoder, Region Decoder, and Content Decoder are used for structure points sequence generation, detection, and recognition, respectively. These three decoders share identical network architectures but have independent parameters. Each decoder includes four transformer decoder layers with eight heads and pre-attention layer nor-

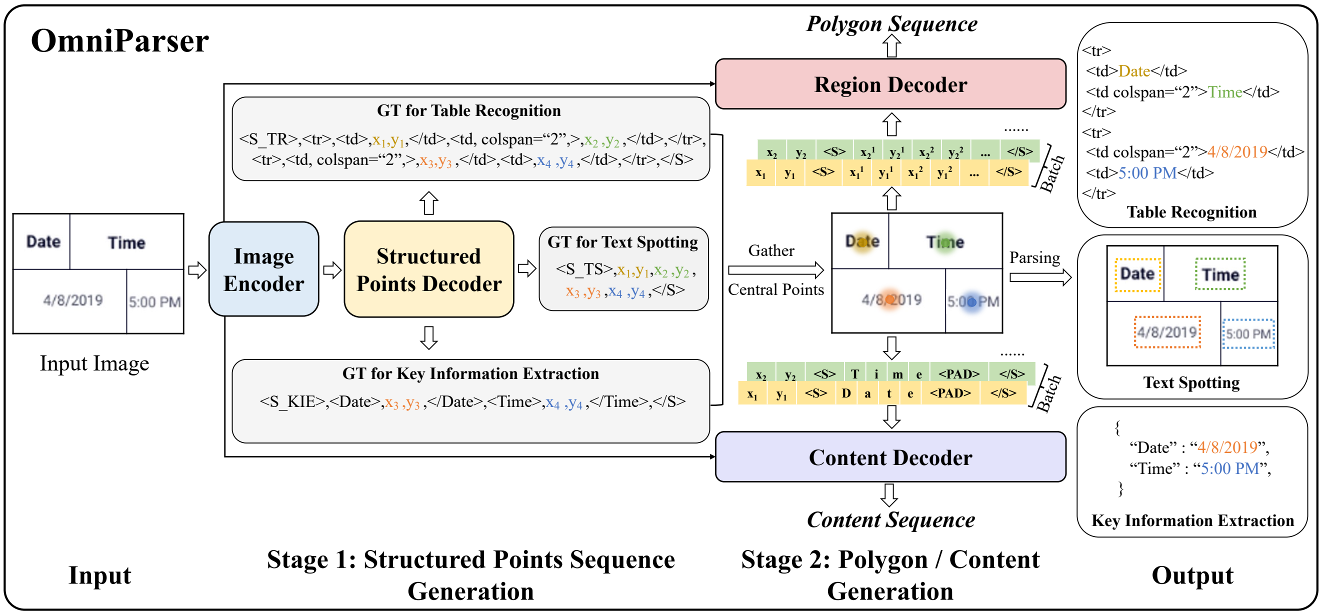

Figure 2. **Schematic illustration of the proposed OmniParser framework.** Structured Points Decoder homogenizes three tasks through a unified structural points representation without designing task-specific branches. Furthermore, benefiting from decoupling points with content recognition and region prediction, the Region Decoder and Content Decoder can generate polygonal contour and text content in parallel given the text points.

malization [83]. The hidden dimension of each decoder layer and amplification factor for the MLP layer are set to 512 and 4 respectively. Due to varying maximum decoding lengths for the three decoders, we assign uniquely randomly initialized positional encodings to each decoder, aiming to better model the dependencies within the sequences.

Objective. During pre-training and fine-tuning, the model is trained by minimizing negative log-likelihood given the input sequence $\mathbf{s}$ and visual embeddings $\mathbf{v}$ at j^{th} time step,

$$L = -\sum_{j=k}^{N} w_j \log P\left(\tilde{\mathbf{s}}_j \mid \mathbf{v}, \mathbf{s}_{k:j-1}\right), \tag{1}$$

where $\tilde{\mathbf{s}}$ denote the target sequence and N is the length of the sequence. Additionally, w_j is the weight value for the j^{th} token. We empirically set w to 4.0 for structural or entity tags and 1.0 for other tokens. First k prompt tokens are excluded from the loss calculation.

3.3. Pre-training Methods

In our framework, generating structural points sequence is more challenging as it requires Structured Points Decoder to understand the text structure and reason entity semantics with image-based input only. Therefore, we adopt spatial-aware and content-aware pre-training strategies: spatial-window prompting and prefix-window prompting, to enhance richer spatial and semantic representation learning.

Spatial-Window Prompting guides the Structured Points Decoder to read text inside a specified window. As shown

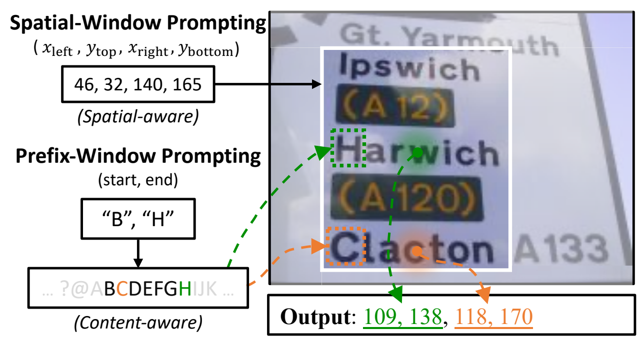

Figure 3. **Spatial-Window Prompting** utilizes a 2-point prompt denoted as $(x_{\texttt{left}}, y_{\texttt{top}}, x_{\texttt{right}}, y_{\texttt{bottom}})$, which specifies the location of the prompting spatial window. **Prefix-Window Prompting** employs a 2-character prompt which indicates the starting and ending characters of the prefix-window with the entire dictionary. The selected prefix range is highlighted in **black**, while others are shaded in gray. The outputs comprise the center points of two words: "Harwich" and "Clacton", as the prefixes 'H' and 'C' fall within the predefined prefix range.

in Fig. 3, only the text center point located in the specified window is considered during training. The spatial-window prompting mechanism consists of two patterns: fixed pattern and random pattern. In the fixed pattern, the window is uniformly sampled from a list of pre-defined layouts, such as 3×3 or 2×2 grids. In the random pattern, the window is randomly sampled from an image, ensuring it covers at least $1/9$ of the image. More details are provided in the supplementary material. Similar to Starting-Point Prompting [29], this spatial-aware prompting strategy allows detecting numerous text from images, even with a limited decoder length.

Prefix-Window Prompting guides the Structured Points Decoder to output center points of text with a specified single char prefix. This strategy aims to instruct the model in locating text instances whose single-character prefix falls within the designated prefix-window charset, while disregarding instances with prefixes outside this charset. The prefix-window charset is sampled from an ordered list of character dictionaries, including 26 uppercase letters, 26 non-capital lowercase, 10 digits, and 34 ASCII punctuation marks, defined by the starting and ending characters. With the aid of prefix-window prompting, the Structured Points Decoder can encode character-level semantics and thus achieve better performance for predicting complex text structures from various tasks such as KIE.

4. Experiments

In this section, we conduct both qualitative and quantitative experiments on standard benchmarks, to verify the effectiveness and advantages of the proposed OMNIPARSER.

4.1. Implementation Details

Pre-training. OMNIPARSER is first trained on a hybrid dataset containing Curved SynthText [46], ICDAR 2013 [27], ICDAR 2015 [28], MLT 2017 [57], Total-Text [8], TextOCR [69], HierText [50], COCO Text [16], and Open Image V5 [32]. To accelerate convergence, we adopt a two-stage pre-training strategy following Pix2seq [6]. In the first stage, the model is trained with a batch size of 128 and image resolution of 768×768 for 500k steps. Subsequently, we continue training for an additional 200k steps with a batch size of 16 and image resolution of 1920×1920. Both stages utilize the AdamW [51] optimizer, with initial learning rates of 5×10^{-4} and 2.5×10^{-4}, respectively. Warm-up schedule is used for the first 5k steps, after which the learning rate is linearly decayed to 0. For data augmentation, we employ instance-aware random cropping, random rotation between $-90°$ and $90°$, random resizing, and color jittering. During pre-training, the center points of text instances are arranged in a raster scan order.

Fine-Tuning. For text spotting and KIE tasks, the model is fine-tuned on the corresponding dataset for 20k and 200k steps respectively, with a learning rate set to 1×10^{-4}. For table recognition, the default maximum sequence lengths for Structured Points Decoder and Content Decoder are set to 1,500 and 200, respectively. The Structured Points Decoder is trained for 400k steps and the Content Decoder is trained for 200k steps with the learning rate set to 1×10^{-4}. For all tasks, the cosine learning rate scheduler is utilized. Besides, the spatial-window prompting and prefix-window prompting are modified as $[0, 0, n_{bins} - 1, n_{bins} - 1]$ and [$\texttt{char}_{\texttt{first}}$, $\texttt{char}_{\texttt{last}}$] ('!' and '~' in the dictionary) respectively, to cover full spatial and prefix range.

4.1.1 Text Spotting

Datasets. We conduct experiments on three popular scene text datasets, Total-Text, ICDAR 2015, and CTW1500 [45]. Total-Text is mainly for arbitrary-shaped text detection and spotting evaluation, consisting of 1255 training images and 300 testing images with word-level polygon annotations. The ICDAR 2015 dataset contains 1000 training images and 500 testing images, annotated with quadrilateral bounding boxes. CTW1500 is another benchmark for curved text detection and recognition, which is annotated at text-line level, including 1000 training images and 500 testing images.

Evaluation Metrics. For Total-Text and CTW1500, we report the end-to-end recognition results over two lexicons: "None" and "Full". "None" means that no lexicons are provided, and "Full" lexicon provides all words in the test set. For ICDAR 2015, we report results over three lexicons: "Strong", "Weak" and "Generic". Strong lexicon provides 100 words that may appear in each image. Weak lexicon provides words in the whole test set, and generic lexicon provides a 90k vocabulary.

4.1.2 Key Information Extraction

Datasets. We evaluate our model's performance on two commonly used benchmark datasets for KIE task: CORD [61] and SROIE [25]. CORD [61] consists of 30 labels across 4 categories. It has 1,000 receipt samples. The train, validation, and test splits contain 800, 100, and 100 samples respectively. The SROIE dataset [25] comprises a training set with 626 receipts and a test set with 347 receipts. Each receipt in the dataset contains four predefined entities, namely: "company", "date", "address", and "total". Annotations in the dataset provide segment-level bounding boxes for the text regions and their corresponding transcriptions.

Evaluation Metrics. Following [30], two evaluation metrics are used to evaluate the performance: field-level F1 measure and tree-edit-distance-based accuracy. The field-level F1 score checks whether each extracted field corresponds exactly to its value in the ground truth.

4.1.3 Table Recognition

Datasets. Given our model's dual prediction of table logical structures (with cell bounding box central points) and cell content, datasets lacking annotations for both cell content and corresponding bounding boxes, as well as those using metrics incompatible with our approach, are excluded from evaluation. For model assessment, PubTabNet (PTN) [97] and FinTabNet (FTN) [95] are selected. **PubTabNet** has 500,777 training images and 9,115 validation images, featuring diverse structures from scientific documents. Our model is evaluated on the validation set due to the lack of

Methods	Total-Text					CTW1500					ICDAR 2015					
	Detection			E2E		Detection			E2E		Detection			E2E		
	P	R	F	None	Full	P	R	F	None	Full	P	R	F	S	W	G
TextDragon [15]	85.6	75.7	80.3	48.8	74.8	82.8	84.5	83.6	39.7	72.4	92.5	83.8	87.9	82.5	78.3	65.2
CharNet [82]	88.6	81.0	84.6	63.6	-	-	-	-	-	-	91.2	88.3	89.7	80.1	74.5	62.2
TextPerceptron [64]	88.8	81.8	85.2	69.7	78.3	-	-	-	57.0	-	92.3	82.5	87.1	80.5	76.6	65.1
CRAFTS [2]	89.5	85.4	87.4	78.7	-	-	-	-	-	-	89.0	85.3	87.1	83.1	82.1	74.9
Boundary [75]	88.9	85.0	87.0	65.0	76.1	-	-	-	-	-	89.8	87.5	88.6	79.7	75.2	64.1
Mask TextSpotter v3 [40]	-	-	-	71.2	78.4	-	-	-	-	-	-	-	-	83.3	78.1	74.2
PGNet [77]	85.5	86.8	86.1	63.1	-	-	-	-	-	-	91.8	84.8	88.2	83.3	78.3	63.5
MANGO [65]	-	-	-	72.9	83.6	-	-	-	58.9	78.7	-	-	-	85.4	80.1	73.9
PAN++ [78]	-	-	-	68.6	78.6	87.1	81.0	84.0	-	-	-	-	-	82.7	78.2	69.2
ABCNet v2 [46]	90.2	84.1	87.0	70.4	78.1	83.8	85.6	84.7	57.5	77.2	90.4	86.0	88.1	82.7	78.5	73.0
TPSNet [79]	90.2	86.8	88.5	76.1	82.3	-	-	-	59.7	79.2	-	-	-	-	-	-
ABINet++ [13]	-	-	-	77.6	84.5	-	-	-	60.2	80.3	-	-	-	84.1	80.4	75.4
GLASS [67]	90.8	85.5	88.1	79.9	86.2	-	-	-	-	-	86.9	84.5	85.7	84.7	80.1	76.3
TESTR [94]	93.4	81.4	86.9	73.3	83.9	92.0	82.6	87.1	56.0	81.5	90.3	89.7	90.0	85.2	79.4	73.6
SwinTextSpotter [22]	-	-	88.0	74.3	84.1	-	-	88.0	51.8	77.0	-	-	-	83.9	77.3	70.5
SPTS [62]	-	-	-	74.2	82.4	-	-	-	63.6	83.8	-	-	-	77.5	70.2	65.8
TTS [31]	-	-	-	78.2	86.3	-	-	-	-	-	-	-	-	85.2	81.7	77.4
UNITS [29]	-	-	89.8	82.2	88.0	-	-	88.6	66.4	82.3	91.0	94.0	92.5	89.0	84.1	**80.3**
DeepSolo [89]	93.2	84.6	88.7	82.5	88.7	-	-	-	56.7	-	92.5	87.2	89.8	88.0	83.5	79.1
DeepSolo* [89]	92.8	82.4	87.4	81.2	87.8	91.5	84.8	88.0	64.9	81.2	92.4	88.8	90.6	88.9	84.4	79.5
OmniParser (ours)	88.4	88.6	88.5	**84.0**	**88.9**	87.9	87.6	87.8	**66.8**	**85.1**	90.3	91.0	90.7	**89.6**	**84.5**	79.9

Table 2. **Comparisons on text spotting task.** 'S', 'W', and 'G' refer to the spotting performance obtained by utilizing strong, weak, and generic lexicons, respectively. The end-to-end metrics are highlighted as they are the primary metrics for text spotting. Bold and underline denote the first and second performances, respectively. * indicates the use of open-source code on our dataset configuration.

public annotations for the test set. **FinTabNet** comprises 112k single-page PDFs with 92,000 cropped training images and 10,656 testing images.

Evaluation Metrics. For evaluation, we utilized Tree-Edit-Distance-based Similarity (TEDS) [97]. TEDS comprehensively evaluates table similarity, considering both structural and cell content aspects in HTML format. The metric represents the HTML table as a tree, and the TEDS score is computed through the tree-edit distance between the ground truth and predicted trees. In addition to overall results, we also provide S-TEDS results, focusing exclusively on the structural aspects and ignoring cell content.

4.2. Comparisons with State-of-The-Art

Text Spotting. In Tab. 2, we compare OmniParser with previous text spotting approaches. On arbitrarily shaped text datasets, Total-Text [8] and CTW1500 [45], our method establishes new state-of-the-art under two end-to-end metrics. In particular, our method surpasses previous SOTA by +1.5% and +3.2% on Total-Text and CTW1500 respectively without lexicon, outperforming all the other competitors. It should be noted that our approach achieves comparable detection results, meanwhile outperforming previous work by a significant margin under the end-to-end metrics. We attribute this superior performance to the decoupling of the detection and recognition processes. On ICDAR 2015 dataset, our method surpasses other approaches, with the exception of the UNITS on generic setting. We presume that joint learning heterogeneous region representations such as bounding boxes, quadrilaterals, and polygons can boost detection performance for tiny and distorted text on the ICDAR 2015, therefore facilitating end-to-end spotting. However, to ensure a more cohesive and standardized region representation, we adopt a 16-point polygonal representation across various visually-situated text parsing tasks.

Methods	Localization Ability	CORD		SROIE	
		F1	Acc	F1	Acc
TRIE [93]	Yes	-	-	82.1	-
Donut [30]	No	84.1	**90.9**	83.2	92.8
Dessurt [10]	No	82.5	-	84.9	-
DocParser [12]	No	84.5	-	**87.3**	-
SeRum [4]	No	80.5	85.8	85.6	92.8
OmniParser (ours)	Yes	**84.8**	88.0	85.6†	**93.6†**

Table 3. **Comparisons of end-to-end methods on key information extraction.** 'F1' denotes the field-level F1 score and 'Acc' denotes the tree-edit-distance-based accuracy. † Since the SROIE dataset does not provide the necessary point location for each entity word, we generate these locations for evaluation purposes.

Key Information Extraction. Tab. 3 reports the performance of KIE task compared to state-of-the-art end-to-end methods on CORD and SROIE datasets. We have exclusively reported $\text{SeRum}_{\text{total}}$ [4] since all generation-based methods utilize a schema that encompasses the entire token sequence of all key information, making it directly comparable. Our

PubTabNet (PTN)				
Methods	Input Size	Decoder Len.	S-TEDS	TEDS
WYGIWYS [11]	512	-	-	78.6
Donut* [30]	1,280	4,000	25.28	22.7
EDD [97]	512	1,800	89.9	88.3
OMNIPARSER (ours)	1,024	1,500	**90.45**	**88.83**
FinTabNet (FTN)				
Methods	Input Size	Decoder Len.	S-TEDS	TEDS
Donut* [30]	1,280	4,000	30.66	29.1
EDD [97]	512	1,800	90.6	-
OMNIPARSER (ours)	1,024	1,500	**91.55**	**89.75**

Table 4. **Comparisons of end-to-end table recognition methods on PubTabNet and FinTabNet datasets.** * represents our reproduced results, where the model was finetuned on PubTabNet and FinTabNet, respectively.

model achieves an 84.8% field-level F1 score on CORD, outperforming previous generation-based approaches. In addition, our method achieves the best TED-based accuracy on SROIE, indicating its superior character-level prediction performance. Notably, the proposed paradigm ensures accurate localization, which is essential for detailed document analysis and correction, a deficiency of other generation-based approaches. Moreover, in contrast to prior studies that utilized a massive corpus of document data for pre-training, our model is pre-trained on scene text data only. This highlights the exceptional generalizability of our unified model.

Table Recognition. In Tab. 4, we compare OMNIPARSER's performance with end-to-end table recognition models. Specifically, we fine-tuned the OCR-free model Donut [30] for table recognition with the official default training configuration. Experimental results show that OMNIPARSER consistently outperforms previous end-to-end methods in TEDS and S-TEDS on various datasets. It's noteworthy that non-end-to-end table structure recognition models [19, 24, 43, 55, 56, 88] use bounding boxes of cell contents for model training and employ offline OCR models for constructing final complete HTML sequences. In contrast, OMNIPARSER utilizes points, achieving comparable results in an end-to-end manner, simplifying post-processing and requiring fewer annotations compared to box-based methods.

5. Analysis

In this section, we begin by conducting ablation experiments on crucial designs in OMNIPARSER. We evaluate these ablations using the Total-Text and ICDAR 2015 text spotting tasks. Furthermore, we provide visualizations on downstream tasks to illustrate the effectiveness of OMNIPARSER.

Ablating Pre-training Strategies. To investigate the effects of spatial-window prompting and prefix-window prompting techniques, we conduct ablative experiments and present the findings in Tab. 5. The inclusion of spatial-window prompting yields a significant enhancement in the performance of our model. This improvement can be attributed to the heightened perception of spatial coordinate positions, thereby enabling more accurate predictions of structured point sequences. Similarly, the incorporation of prefix-window prompting also results in a noticeable improvement in performance, as it enhances the model's ability to perceive diverse textual content within images. The spatial-window prompting and prefix-window prompting enhance the model's perception ability in coordinate space and semantic space respectively. Notably, when both prompting techniques are employed simultaneously, the model achieved state-of-the-art performance on both datasets.

Window-Prompting		Total-Text		ICDAR 2015		
Spatial-	Prefix-	None	Full	S	W	G
		82.4	87.6	88.1	83.0	78.3
	✓	82.9	88.1	88.4	83.2	78.5
✓		83.5	88.5	89.2	84.2	79.4
✓	✓	84.0	88.9	89.6	84.5	79.9

Table 5. **Ablation of pre-training strategies** on text spotting.

Visual Backbone	Decoder	Total-Text		ICDAR 2015		
		None	Full	S	W	G
ResNet50	Not Shared	82.1	87.1	88.2	83.0	78.4
Swin-B	Shared	82.5	87.3	88.5	83.2	78.7
Swin-B	Not Shared	84.0	88.9	89.6	84.5	79.9

Table 6. **Ablation of encoder and decoder designs** on the text spotting task.

Ablating Architectural Designs. We conduct a comparative analysis of various architectural designs for both the visual encoder and decoders, as presented in Tab. 6. As our model comprises three decoders that share the same architecture, we aim to investigate whether weight sharing among these decoders can enhance the overall performance. However, our observations reveal that when employing a shared decoder, the performance on text spotting tasks diminishes, suggesting a potential discrepancy among the subtasks of decoding center points, polygons, and content. Additionally, we compare the backbones of ResNet50 and Swin-B. Remarkably, Swin-B outperforms ResNet50, demonstrating its superiority in visually-situated text parsing tasks.

PubTabNet (PTN)				
Methods	S-Decoder Len.	C-Decoder Len.	S-TEDS	TEDS
OMNIPARSER	1,124	200	89.94	88.21
	1,500	200	**90.45**	88.83
	2,000	300	**90.45**	**88.96**

Table 7. **Ablation of decoder length for the table recognition task on PubTabNet datasets.** S-Decoder Len. and C-Decoder Len.: short for the length of Structured Points Decoder and Content Decoder, respectively.

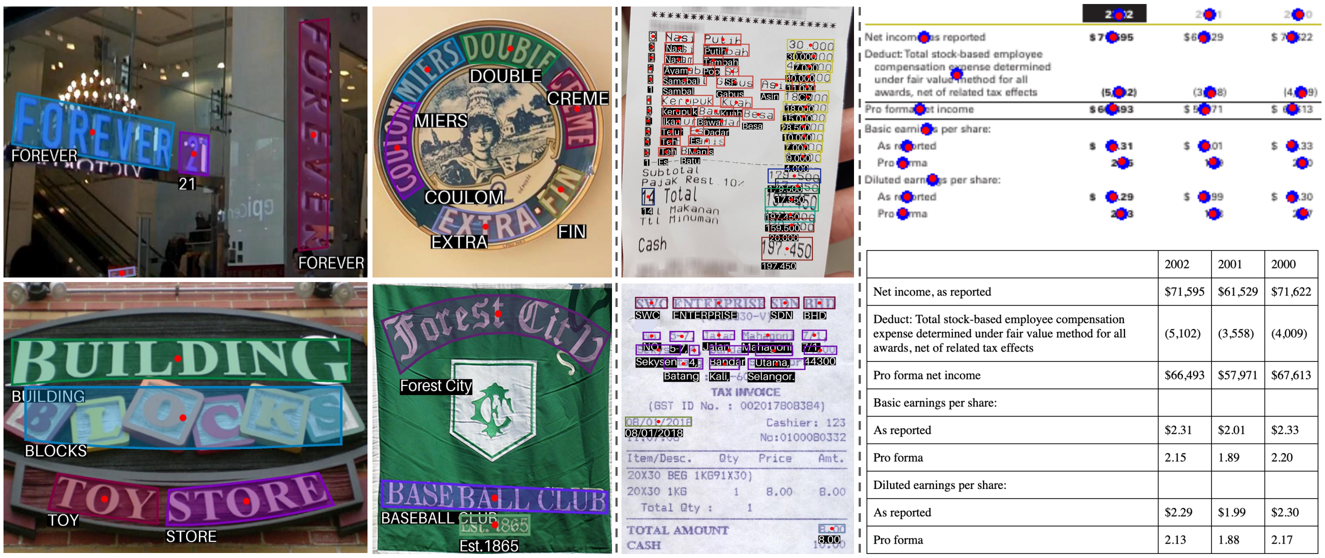

Figure 4. **Qualitative results** of text spotting (column 1-2), KIE (column 3), and table recognition (column 4). For KIE, points, polygons, and recognition are visualized. The color assigned to polygons indicates the entity type. For table recognition, we present point locations and a rendered table based on the prediction sequence, with an additional border for readability. Blue points and red points denote the GT and predicted points respectively. More details can be found in the supplementary material. (The figure is best viewed in color.)

Ablating Decoder Length. In Tab. 7, we perform an ablation study on decoder lengths for end-to-end table recognition. Due to GPU constraints, Donut's max length is set to 4,000 (shown in Tab. 4), while our model at 1,500 achieves better results. Note that the average inference speed of our method and Donut are 1.3 and 0.8 FPS, respectively. Training end-to-end models like Donut with complete HTML sequences poses challenges for lengthy sequences, such as those encountered in table recognition, where there is a high probability of error accumulation and attention drift. Our modularized architecture separates pure table HTML tags and cell text sequences, enabling end-to-end recognition without length restrictions. Besides, increasing the length of Structured Points Decoder from 1,500 to 2,000 shows no improvement in S-TEDS, with slight TEDS enhancement when the text length increases from 200 to 300. In practice, decoder length choice requires a trade-off between performance and efficiency.

Qualitative Results. We show qualitative results for three tasks in Fig. 4: 1) For text spotting, our model can accurately detect and recognize curve texts, vertical texts, and artistic texts under challenging scenarios. Despite some imprecise detections, the recognition results are entirely accurate. 2) In table recognition results, hard cases of spanning cells, borderless tables, and cells with multi-line content are presented. These examples show that our method can correctly localize cell centers through the structured points sequence. 3) KIE results demonstrate the efficacy of our approach in effectively localizing, recognizing texts and, more importantly, extracting entity information.

Limitations. Despite achieving promising results on visually-situated text tasks, the proposed OMNIPARSER has a few limitations. Firstly, it relies on having precise word point locations during training, which may not be always available in certain real-world scenarios. Secondly, it does not account for parsing non-text elements such as figures or charts, limiting its potential in solving complex document parsing tasks. Addressing such limitations and improving the robustness as well as the applicability of our model in real-world settings will be the focus of our future research.

6. Conclusions and Future Works

In this paper, we have proposed a general-purpose parsing framework OMNIPARSER, which brings together the tasks of text spotting, key information extraction, and table recognition in a visually-situated text parsing context. This is realized through a two-stage decoding procedure, leveraging structured points as an adapter. To enhance the effectiveness of pre-training across all tasks, we also introduce two pre-training strategies to enable the Structured Points Decoder to learn complex structures and relations among visually-situated texts, further improving the overall performance.

The proposed OMNIPARSER achieves state-of-the-art or highly competitive performance on standard benchmarks, even compared with specialist models that rely on task-specific designs. As a general-purpose parser, OMNIPARSER has been proven quite effective on various visually-situated text tasks, so we will extend it to more tasks and scenarios, e.g., layout analysis and chart parsing.

Acknowledgements. This work was supported by the National Natural Science Foundation of China (No.62225603), and Alibaba Innovative Research (AIR) program.

References

[1] Srikar Appalaraju, Bhavan Jasani, Bhargava Urala Kota, Yusheng Xie, and R Manmatha. Docformer: End-to-end transformer for document understanding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 993–1003, 2021. 2

[2] Youngmin Baek, Seung Shin, Jeonghun Baek, Sungrae Park, Junyeop Lee, Daehyun Nam, and Hwalsuk Lee. Character region attention for text spotting. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIX 16*, pages 504–521. Springer, 2020. 6

[3] Haoyu Cao, Xin Li, Jiefeng Ma, Deqiang Jiang, Antai Guo, Yiqing Hu, Hao Liu, Yinsong Liu, and Bo Ren. Query-driven generative network for document information extraction in the wild. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 4261–4271, 2022. 3

[4] Haoyu Cao, Changcun Bao, Chaohu Liu, Huang Chen, Kun Yin, Hao Liu, Yinsong Liu, Deqiang Jiang, and Xing Sun. Attention where it matters: Rethinking visual document understanding with selective region concentration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19517–19527, 2023. 2, 3, 6

[5] Panfeng Cao, Ye Wang, Qiang Zhang, and Zaiqiao Meng. Genkie: Robust generative multimodal document key information extraction. *arXiv preprint arXiv:2310.16131*, 2023. 3

[6] Ting Chen, Saurabh Saxena, Lala Li, David J Fleet, and Geoffrey Hinton. Pix2seq: A language modeling framework for object detection. In *International Conference on Learning Representations*, 2021. 5

[7] Xi Chen, Josip Djolonga, Piotr Padlewski, Basil Mustafa, Soravit Changpinyo, Jialin Wu, Carlos Riquelme Ruiz, Sebastian Goodman, Xiao Wang, Yi Tay, Siamak Shakeri, Mostafa Dehghani, Daniel M. Salz, Mario Lucic, Michael Tschannen, Arsha Nagrani, Hexiang Hu, Mandar Joshi, Bo Pang, Ceslee Montgomery, Paulina Pietrzyk, Marvin Ritter, A. J. Piergiovanni, Matthias Minderer, Filip Pavetic, Austin Waters, Gang Li, Ibrahim M. Alabdulmohsin, Lucas Beyer, Julien Amelot, Kenton Lee, Andreas Steiner, Yang Li, Daniel Keysers, Anurag Arnab, Yuanzhong Xu, Keran Rong, Alexander Kolesnikov, Mojtaba Seyedhosseini, Anelia Angelova, Xiaohua Zhai, Neil Houlsby, and Radu Soricut. Pali-x: On scaling up a multilingual vision and language model. *ArXiv*, abs/2305.18565, 2023. 1

[8] Chee-Kheng Ch'ng, Chee Seng Chan, and Cheng-Lin Liu. Total-text: toward orientation robustness in scene text detection. *International Journal on Document Analysis and Recognition (IJDAR)*, 23(1):31–52, 2020. 5, 6

[9] Cheng Da, Peng Wang, and Cong Yao. Multi-granularity prediction with learnable fusion for scene text recognition. *arXiv preprint arXiv:2307.13244*, 2023. 2

[10] Brian Davis, Bryan Morse, Brian Price, Chris Tensmeyer, Curtis Wigington, and Vlad Morariu. End-to-end document recognition and understanding with dessurt. In *European Conference on Computer Vision*, pages 280–296. Springer, 2022. 3, 6

[11] Yuntian Deng, Anssi Kanervisto, Jeffrey Ling, and Alexander M Rush. Image-to-markup generation with coarse-to-fine attention. In *International Conference on Machine Learning*, pages 980–989. PMLR, 2017. 7

[12] Mohamed Dhouib, Ghassen Bettaieb, and Aymen Shabou. Docparser: End-to-end ocr-free information extraction from visually rich documents. *arXiv preprint arXiv:2304.12484*, 2023. 3, 6

[13] Shancheng Fang, Zhendong Mao, Hongtao Xie, Yuxin Wang, Chenggang Yan, and Yongdong Zhang. Abinet++: Autonomous, bidirectional and iterative language modeling for scene text spotting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 6

[14] Hao Feng, Zijian Wang, Jingqun Tang, Jinghui Lu, Wengang Zhou, Houqiang Li, and Can Huang. Unidoc: A universal large multimodal model for simultaneous text detection, recognition, spotting and understanding. *arXiv preprint arXiv:2308.11592*, 2023. 2

[15] Wei Feng, Wenhao He, Fei Yin, Xu-Yao Zhang, and Cheng-Lin Liu. Textdragon: An end-to-end framework for arbitrary shaped text spotting. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9076–9085, 2019. 2, 6

[16] Raul Gomez, Baoguang Shi, Lluis Gomez, Lukas Numann, Andreas Veit, Jiri Matas, Serge Belongie, and Dimosthenis Karatzas. Icdar2017 robust reading challenge on coco-text. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, pages 1435–1443. IEEE, 2017. 5

[17] Jiuxiang Gu, Jason Kuen, Vlad I Morariu, Handong Zhao, Rajiv Jain, Nikolaos Barmpalios, Ani Nenkova, and Tong Sun. Unidoc: Unified pretraining framework for document understanding. *Advances in Neural Information Processing Systems*, 34:39–50, 2021. 2

[18] Zhangxuan Gu, Changhua Meng, Ke Wang, Jun Lan, Weiqiang Wang, Ming Gu, and Liqing Zhang. Xylayoutlm: Towards layout-aware multimodal networks for visually-rich document understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4583–4592, 2022. 2

[19] Zengyuan Guo, Yuechen Yu, Pengyuan Lv, Chengquan Zhang, Haojie Li, Zhihui Wang, Kun Yao, Jingtuo Liu, and Jingdong Wang. Trust: An accurate and end-to-end table structure recognizer using splitting-based transformers. *arXiv preprint arXiv:2208.14687*, 2022. 3, 7

[20] Tong He, Zhi Tian, Weilin Huang, Chunhua Shen, Yu Qiao, and Changming Sun. An end-to-end textspotter with explicit alignment and attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5020–5029, 2018. 2

[21] Teakgyu Hong, Donghyun Kim, Mingi Ji, Wonseok Hwang, Daehyun Nam, and Sungrae Park. Bros: A pre-trained language model focusing on text and layout for better key information extraction from documents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 10767–10775, 2022. 2, 3

[22] Mingxin Huang, Yuliang Liu, Zhenghao Peng, Chongyu Liu, Dahua Lin, Shenggao Zhu, Nicholas Yuan, Kai Ding, and

Lianwen Jin. Swintextspotter: Scene text spotting via better synergy between text detection and text recognition. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4593–4603, 2022. 6

[23] Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. Layoutlmv3: Pre-training for document ai with unified text and image masking. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 4083–4091, 2022. 2

[24] Yongshuai Huang, Ning Lu, Dapeng Chen, Yibo Li, Zecheng Xie, Shenggao Zhu, Liangcai Gao, and Wei Peng. Improving table structure recognition with visual-alignment sequential coordinate modeling. In *CVPR*, pages 11134–11143, 2023. 3, 7

[25] Zheng Huang, Kai Chen, Jianhua He, Xiang Bai, Dimosthenis Karatzas, Shijian Lu, and CV Jawahar. Icdar2019 competition on scanned receipt ocr and information extraction. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1516–1520. IEEE, 2019. 5

[26] Wonseok Hwang, Jinyeong Yim, Seunghyun Park, Sohee Yang, and Minjoon Seo. Spatial dependency parsing for semi-structured document information extraction. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 330–343, 2021. 2

[27] 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. 5

[28] 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. 5

[29] Taeho Kil, Seonghyeon Kim, Sukmin Seo, Yoonsik Kim, and Daehee Kim. Towards unified scene text spotting based on sequence generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15223–15232, 2023. 2, 4, 6

[30] Geewook Kim, Teakgyu Hong, Moonbin Yim, JeongYeon Nam, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. Ocr-free document understanding transformer. In *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part XXVIII*, pages 498–517. Springer, 2022. 2, 3, 5, 6, 7

[31] Yair Kittenplon, Inbal Lavi, Sharon Fogel, Yarin Bar, R Manmatha, and Pietro Perona. Towards weakly-supervised text spotting using a multi-task transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4604–4613, 2022. 6

[32] Ilya Krylov, Sergei Nosov, and Vladislav Sovrasov. Open images v5 text annotation and yet another mask text spotter. In *Asian Conference on Machine Learning*, pages 379–389. PMLR, 2021. 5

[33] Jianfeng Kuang, Wei Hua, Dingkang Liang, Mingkun Yang, Deqiang Jiang, Bo Ren, and Xiang Bai. Visual information extraction in the wild: practical dataset and end-to-end solution. In *International Conference on Document Analysis and Recognition*, pages 36–53. Springer, 2023. 3

[34] Chen-Yu Lee, Chun-Liang Li, Timothy Dozat, Vincent Perot, Guolong Su, Nan Hua, Joshua Ainslie, Renshen Wang, Yasuhisa Fujii, and Tomas Pfister. Formnet: Structural encoding beyond sequential modeling in form document information extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3735–3754, 2022. 2

[35] Chenliang Li, Bin Bi, Ming Yan, Wei Wang, Songfang Huang, Fei Huang, and Luo Si. Structurallm: Structural pre-training for form understanding. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6309–6318, 2021. 2

[36] Hui Li, Peng Wang, and Chunhua Shen. Towards end-to-end text spotting with convolutional recurrent neural networks. In *Proceedings of the IEEE international conference on computer vision*, pages 5238–5246, 2017. 2

[37] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, 2023. 1

[38] Peizhao Li, Jiuxiang Gu, Jason Kuen, Vlad I Morariu, Handong Zhao, Rajiv Jain, Varun Manjunatha, and Hongfu Liu. Selfdoc: Self-supervised document representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5652–5660, 2021. 2

[39] Xin Li, Yan Zheng, Yiqing Hu, Haoyu Cao, Yunfei Wu, Deqiang Jiang, Yinsong Liu, and Bo Ren. Relational representation learning in visually-rich documents. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 4614–4624, 2022. 1, 2, 3

[40] Minghui Liao, Guan Pang, Jing Huang, Tal Hassner, and Xiang Bai. Mask textspotter v3: Segmentation proposal network for robust scene text spotting. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, pages 706–722. Springer, 2020. 2, 6

[41] 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

[42] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 3

[43] Weihong Lin, Zheng Sun, Chixiang Ma, Mingze Li, Jiawei Wang, Lei Sun, and Qiang Huo. Tsrformer: Table structure

recognition with transformers. In *ACM MM*, pages 6473–6482, 2022. 3, 7

[44] Xuebo Liu, Ding Liang, Shi Yan, Dagui Chen, Yu Qiao, and Junjie Yan. Fots: Fast oriented text spotting with a unified network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5676–5685, 2018. 2

[45] Yuliang Liu, Lianwen Jin, Shuaitao Zhang, Canjie Luo, and Sheng Zhang. Curved scene text detection via transverse and longitudinal sequence connection. *Pattern Recognition*, 90: 337–345, 2019. 5, 6

[46] Yuliang Liu, Chunhua Shen, Lianwen Jin, Tong He, Peng Chen, Chongyu Liu, and Hao Chen. Abcnet v2: Adaptive bezier-curve network for real-time end-to-end text spotting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):8048–8064, 2021. 2, 5, 6

[47] Yuliang Liu, Jiaxin Zhang, Dezhi Peng, Mingxin Huang, Xinyu Wang, Jingqun Tang, Can Huang, Dahua Lin, Chunhua Shen, Xiang Bai, et al. Spts v2: single-point scene text spotting. *arXiv preprint arXiv:2301.01635*, 2023. 2

[48] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 3

[49] Rujiao Long, Wen Wang, Nan Xue, Feiyu Gao, Zhibo Yang, Yongpan Wang, and Gui-Song Xia. Parsing table structures in the wild. In *ICCV*, pages 944–952, 2021. 1, 2

[50] Shangbang Long, Siyang Qin, Dmitry Panteleev, Alessandro Bissacco, Yasuhisa Fujii, and Michalis Raptis. Towards end-to-end unified scene text detection and layout analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1049–1059, 2022. 3, 5

[51] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2018. 5

[52] Chuwei Luo, Changxu Cheng, Qi Zheng, and Cong Yao. Geolayoutlm: Geometric pre-training for visual information extraction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7092–7101, 2023. 2

[53] Nam Tuan Ly and Atsuhiro Takasu. An end-to-end local attention based model for table recognition. In *ICDAR*, pages 20–36. Springer, 2023. 3

[54] Pengyuan Lyu, Minghui Liao, Cong Yao, Wenhao Wu, and Xiang Bai. Mask textspotter: An end-to-end trainable neural network for spotting text with arbitrary shapes. In *Proceedings of the European conference on computer vision (ECCV)*, pages 67–83, 2018. 2

[55] Pengyuan Lyu, Weihong Ma, Hongyi Wang, Yuechen Yu, Chengquan Zhang, Kun Yao, Yang Xue, and Jingdong Wang. Gridformer: Towards accurate table structure recognition via grid prediction. In *ACM MM*, pages 7747–7757, 2023. 3, 7

[56] Ahmed Nassar, Nikolaos Livathinos, Maksym Lysak, and Peter Staar. Tableformer: Table structure understanding with transformers. In *CVPR*, pages 4614–4623, 2022. 3, 7

[57] Nibal Nayef, Fei Yin, Imen Bizid, Hyunsoo Choi, Yuan Feng, Dimosthenis Karatzas, Zhenbo Luo, Umapada Pal, Christophe Rigaud, Joseph Chazalon, et al. Icdar2017 robust reading challenge on multi-lingual scene text detection and script identification-rrc-mlt. In *2017 14th IAPR international conference on document analysis and recognition (ICDAR)*, pages 1454–1459. IEEE, 2017. 5

[58] OpenAI. ChatGPT. `https://openai.com/chatgpt`, 2023. Accessed: 2023-09-27. 1

[59] OpenAI. GPT-4. `https://openai.com/gpt-4`, 2023. Accessed: 2023-09-27. 1

[60] OpenAI. GPT-4V(ision) System Card. `https://cdn.openai.com/papers/GPTV_System_Card.pdf`, 2023. Accessed: 2023-10-09. 1

[61] Seunghyun Park, Seung Shin, Bado Lee, Junyeop Lee, Jaeheung Surh, Minjoon Seo, and Hwalsuk Lee. Cord: A consolidated receipt dataset for post-ocr parsing. In *Document Intelligence Workshop at Neural Information Processing Systems*, 2019. 5

[62] Dezhi Peng, Xinyu Wang, Yuliang Liu, Jiaxin Zhang, Mingxin Huang, Songxuan Lai, Jing Li, Shenggao Zhu, Dahua Lin, Chunhua Shen, et al. Spts: single-point text spotting. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 4272–4281, 2022. 2, 6

[63] Qiming Peng, Yinxu Pan, Wenjin Wang, Bin Luo, Zhenyu Zhang, Zhengjie Huang, Yuhui Cao, Weichong Yin, Yongfeng Chen, Yin Zhang, et al. Ernie-layout: Layout knowledge enhanced pre-training for visually-rich document understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3744–3756, 2022. 2

[64] Liang Qiao, Sanli Tang, Zhanzhan Cheng, Yunlu Xu, Yi Niu, Shiliang Pu, and Fei Wu. Text perceptron: Towards end-to-end arbitrary-shaped text spotting. In *Proceedings of the AAAI conference on artificial intelligence*, pages 11899–11907, 2020. 6

[65] Liang Qiao, Ying Chen, Zhanzhan Cheng, Yunlu Xu, Yi Niu, Shiliang Pu, and Fei Wu. Mango: A mask attention guided one-stage scene text spotter. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2467–2476, 2021. 2, 6

[66] Siyang Qin, Alessandro Bissacco, Michalis Raptis, Yasuhisa Fujii, and Ying Xiao. Towards unconstrained end-to-end text spotting. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4704–4714, 2019. 2

[67] Roi Ronen, Shahar Tsiper, Oron Anschel, Inbal Lavi, Amir Markovitz, and R Manmatha. Glass: Global to local attention for scene-text spotting. In *European Conference on Computer Vision*, pages 249–266. Springer, 2022. 6

[68] 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. 2

[69] Amanpreet Singh, Guan Pang, Mandy Toh, Jing Huang, Wojciech Galuba, and Tal Hassner. Textocr: Towards large-scale end-to-end reasoning for arbitrary-shaped scene text. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8802–8812, 2021. 5

[70] Sibo Song, Jianqiang Wan, Zhibo Yang, Jun Tang, Wenqing Cheng, Xiang Bai, and Cong Yao. Vision-language pre-training for boosting scene text detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15681–15691, 2022. 3

[71] Peter WJ Staar, Michele Dolfi, Christoph Auer, and Costas Bekas. Corpus conversion service: A machine learning platform to ingest documents at scale. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 774–782, 2018. 3

[72] Yipeng Sun, Chengquan Zhang, Zuming Huang, Jiaming Liu, Junyu Han, and Errui Ding. Textnet: Irregular text reading from images with an end-to-end trainable network. In *Computer Vision–ACCV 2018: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers, Part III 14*, pages 83–99. Springer, 2019. 2

[73] Guozhi Tang, Lele Xie, Lianwen Jin, Jiapeng Wang, Jingdong Chen, Zhen Xu, Qianying Wang, Yaqiang Wu, and Hui Li. Matchvie: Exploiting match relevancy between entities for visual information extraction. *arXiv preprint arXiv:2106.12940*, 2021. 3

[74] Zineng Tang, Ziyi Yang, Guoxin Wang, Yuwei Fang, Yang Liu, Chenguang Zhu, Michael Zeng, Cha Zhang, and Mohit Bansal. Unifying vision, text, and layout for universal document processing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19254–19264, 2023. 3

[75] Hao Wang, Pu Lu, Hui Zhang, Mingkun Yang, Xiang Bai, Yongchao Xu, Mengchao He, Yongpan Wang, and Wenyu Liu. All you need is boundary: Toward arbitrary-shaped text spotting. In *Proceedings of the AAAI conference on artificial intelligence*, pages 12160–12167, 2020. 6

[76] Jiapeng Wang, Chongyu Liu, Lianwen Jin, Guozhi Tang, Jiaxin Zhang, Shuaitao Zhang, Qianying Wang, Yaqiang Wu, and Mingxiang Cai. Towards robust visual information extraction in real world: New dataset and novel solution. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2738–2745, 2021. 1, 3

[77] Pengfei Wang, Chengquan Zhang, Fei Qi, Shanshan Liu, Xiaoqiang Zhang, Pengyuan Lyu, Junyu Han, Jingtuo Liu, Errui Ding, and Guangming Shi. Pgnet: Real-time arbitrarily-shaped text spotting with point gathering network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2782–2790, 2021. 6

[78] Wenhai Wang, Enze Xie, Xiang Li, Xuebo Liu, Ding Liang, Zhibo Yang, Tong Lu, and Chunhua Shen. Pan++: Towards efficient and accurate end-to-end spotting of arbitrarily-shaped text. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(9):5349–5367, 2021. 6

[79] Wei Wang, Yu Zhou, Jiahao Lv, Dayan Wu, Guoqing Zhao, Ning Jiang, and Weipinng Wang. Tpsnet: Reverse thinking of thin plate splines for arbitrary shape scene text representation. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 5014–5025, 2022. 6

[80] Zilong Wang, Yiheng Xu, Lei Cui, Jingbo Shang, and Furu Wei. Layoutreader: Pre-training of text and layout for reading order detection. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4735–4744, 2021. 2

[81] Kaiwen Wei, Jie Yao, Jingyuan Zhang, Yangyang Kang, Fubang Zhao, Yating Zhang, Changlong Sun, Xin Jin, and Xin Zhang. Ppn: Parallel pointer-based network for key information extraction with complex layouts. *arXiv preprint arXiv:2307.10551*, 2023. 2

[82] Linjie Xing, Zhi Tian, Weilin Huang, and Matthew R Scott. Convolutional character networks. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9126–9136, 2019. 6

[83] Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tieyan Liu. On layer normalization in the transformer architecture. In *International Conference on Machine Learning*, pages 10524–10533. PMLR, 2020. 4

[84] Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. Layoutlm: Pre-training of text and layout for document image understanding. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1192–1200, 2020. 2

[85] Yiheng Xu, Tengchao Lv, Lei Cui, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, and Furu Wei. Layoutxlm: Multimodal pre-training for multilingual visually-rich document understanding. *arXiv preprint arXiv:2104.08836*, 2021. 2

[86] Yang Xu, Yiheng Xu, Tengchao Lv, Lei Cui, Furu Wei, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, Wanxiang Che, et al. Layoutlmv2: Multi-modal pre-training for visually-rich document understanding. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2579–2591, 2021. 2

[87] Zhibo Yang, Rujiao Long, Pengfei Wang, Sibo Song, Humen Zhong, Wenqing Cheng, Xiang Bai, and Cong Yao. Modeling entities as semantic points for visual information extraction in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15358–15367, 2023. 2

[88] Jiaquan Ye, Xianbiao Qi, Yelin He, Yihao Chen, Dengyi Gu, Peng Gao, and Rong Xiao. Pingan-vcgroup's solution for icdar 2021 competition on scientific literature parsing task b: table recognition to html. *arXiv preprint arXiv:2105.01848*, 2021. 3, 7

[89] Maoyuan Ye, Jing Zhang, Shanshan Zhao, Juhua Liu, Tongliang Liu, Bo Du, and Dacheng Tao. Deepsolo: Let transformer decoder with explicit points solo for text spotting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19348–19357, 2023. 1, 2, 6

[90] Wenwen Yu, Ning Lu, Xianbiao Qi, Ping Gong, and Rong Xiao. Pick: processing key information extraction from documents using improved graph learning-convolutional networks. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 4363–4370. IEEE, 2021. 2

[91] Yuechen Yu, Yulin Li, Chengquan Zhang, Xiaoqiang Zhang, Zengyuan Guo, Xiameng Qin, Kun Yao, Junyu Han, Errui

Ding, and Jingdong Wang. Structextv2: Masked visual-textual prediction for document image pre-training. In *The Eleventh International Conference on Learning Representations*, 2022. 2, 3

[92] Chong Zhang, Ya Guo, Yi Tu, Huan Chen, Jinyang Tang, Huijia Zhu, Qi Zhang, and Tao Gui. Reading order matters: Information extraction from visually-rich documents by token path prediction. *arXiv preprint arXiv:2310.11016*, 2023. 2

[93] Peng Zhang, Yunlu Xu, Zhanzhan Cheng, Shiliang Pu, Jing Lu, Liang Qiao, Yi Niu, and Fei Wu. Trie: end-to-end text reading and information extraction for document understanding. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1413–1422, 2020. 3, 6

[94] Xiang Zhang, Yongwen Su, Subarna Tripathi, and Zhuowen Tu. Text spotting transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9519–9528, 2022. 2, 6

[95] Xinyi Zheng, Douglas Burdick, Lucian Popa, Xu Zhong, and Nancy Xin Ru Wang. Global table extractor (gte): A framework for joint table identification and cell structure recognition using visual context. In *WACV*, pages 697–706, 2021. 5

[96] Xu Zhong, Jianbin Tang, and Antonio Jimeno Yepes. Publaynet: largest dataset ever for document layout analysis. In *ICDAR*, pages 1015–1022. IEEE, 2019. 3

[97] Xu Zhong, Elaheh ShafieiBavani, and Antonio Jimeno Yepes. Image-based table recognition: data, model, and evaluation. In *ECCV*, pages 564–580. Springer, 2020. 3, 5, 6, 7

[98] Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, and Jiajun Liang. East: an efficient and accurate scene text detector. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 5551–5560, 2017. 2