InstructTable: Improving Table Structure Recognition Through Instructions

Boming Chen[1], Zining Wang[1], Zhentao Guo[2], Jianqiang Liu[1], Chen Duan[1], Yu Gu[1], Kai Zhou[1], Pengfei Yan[1]
[1] Meituan
[2] Beijing Institute of Technology
{chennboming,wangzining03}@meituan.com,

Abstract

Table structure recognition (TSR) holds widespread practical importance by parsing tabular images into structured representations, yet encounters significant challenges when processing complex layouts involving merged or empty cells. Traditional visual-centric models rely exclusively on visual information while lacking crucial semantic support, thereby impeding accurate structural recognition in complex scenarios. Vision-language models leverage contextual semantics to enhance comprehension; however, these approaches underemphasize the modeling of visual structural information. To address these limitations, this paper introduces InstructTable, an instruction-guided multi-stage training TSR framework. Meticulously designed table instruction pre-training directs attention toward fine-grained structural patterns, enhancing comprehension of complex tables. Complementary TSR fine-tuning preserves robust visual information modeling, maintaining high-precision table parsing across diverse scenarios. Furthermore, we introduce Table Mix Expand (TME), an innovative template-free method for synthesizing large-scale authentic tabular data. Leveraging TME, we construct the Balanced Complex Dense Synthetic Tables (BCDSTab) benchmark, comprising 900 complex table images synthesized through our method to serve as a rigorous benchmark. Extensive experiments on multiple public datasets (FinTabNet, PubTabNet, MUSTARD) and BCDSTab demonstrate that InstructTable achieves state-of-the-art performance in TSR tasks. Ablation studies further confirm the positive impact of the proposed tabular-data-specific instructions and synthetic data. Code and datasets are available at `https://github.com/Nevar07/InstructTable`

1. Introduction

Tables, as a fundamental medium for organizing and presenting structured information, are ubiquitous in academic publications, financial reports, official documents, and even natural scene images. Accurate **T**able **S**tructure **R**ecognition (TSR) which involves identifying cell boundaries, logical relationships (e.g., row/column merges), and spatial layouts, serves as a critical prerequisite to downstream tasks including data extraction, knowledge graph construction, and automated document analysis [10]. However, due to the high heterogeneity of table layouts, such as empty cells, merged cells, and visual ambiguity, TSR remains a challenging task demanding methodological breakthroughs, particularly for complex tabular structures.

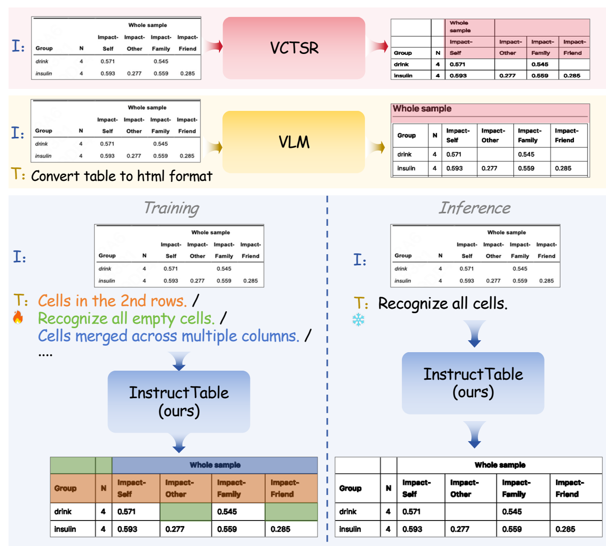

Figure 1. Visualized comparison among traditional **visual-centric TSR (VCTSR) models**, **vision-language models (VLM)**, and **InstructTable**. By leveraging instruction pre-training and TSR fine-tuning to jointly model visual information and instruction dependencies, InstructTable enhance fine-grained structural comprehension of tables.

As illustrated in Fig. 1, existing methods exhibit a semantic-visual imbalance: traditional visual-centric TSR models [19, 23, 43, 47, 53] solely rely on visual features while neglecting semantic associations, resulting in misjudgment of fine-grained table structure such merged cells or empty cells, whereas vision-language models [2, 3, 5, 34, 41] overemphasize semantic information and compromise critical visual details such as alignment. To address this, we propose an instruction-guided TSR framework **InstructTable**, which dynamically modulates attention through multi-stage training strategy to collaboratively model visual structures and semantic dependencies.

Beyond model limitations, acquiring high-quality tabular data remains a significant challenge due to the complex annotation process including object detection, text recognition, and structural labeling. While existing method [23] attempts to address this by synthesizing data through templates, the generated data exhibits limited diversity and authenticity. More critically, we identified systematic errors in existing template-based synthetic table datasets. Specifically, when cells with both *rowspan* and *colspan* attributes occupy full rows/columns, renderer generates implicit empty rows/columns during image generation. As illustrated in Fig. 2, this causes visual representations to contradict ground truth structural annotations, affecting about 2% of the dataset and introducing over **20,000** redundant rows, compromising the dataset's utility for training. To resolve these limitations, we propose **T**able **M**ix **E**xpand (TME), a synthesis method that parses authentic tables into atomic cell matrices and subsequently splices them for data generation with the help of large language models (LLMs). While maximally preserving data authenticity, TME enables arbitrary mixing and expansion of tabular data. This approach effectively circumvents implicit structural errors and facilitates low-cost, large-scale generation of arbitrarily sized table datasets.

The primary contributions are summarized as follows:

- We introduce **InstructTable**, an instruction-guided multi-stage training TSR framework that bridges semantic instructions and visual information via instruction pre-training and TSR fine-tuning, significantly improving recognition accuracy for complex table structures.
- We propose **TME**, parsing authentic tables into atomic cell matrix for arbitrary mixing and expansion. It enables large-scale, diverse synthetic tabular data generation from real samples, eliminating template-induced bias and alleviating high-quality data scarcity.
- We introduce **B**alanced **C**omplex **D**ense **S**ynthetic **Tab**les (BCDSTab), a TME-synthesized benchmark with 900 complex table images for evaluating models in complex table scenarios.

2. Related Work

Advances in deep learning have driven the proliferation of TSR methods, which can be categorized into two paradigms: visual-centric and vision-language model based approaches, differentiated by their use of textual semantics.

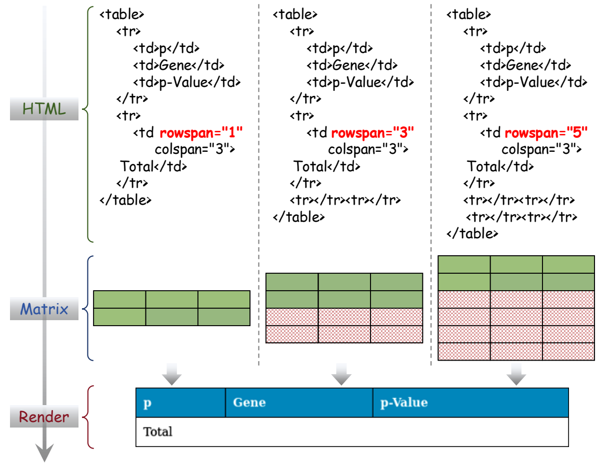

Figure 2. Implicit row problem visualization – identical images from divergent ground truths causing input-output misalignment, misleading model training. In the matrix, green regions denote normal cells while red dotted patterns indicate implicit rows.

2.1. Visual-Centric Methods

Visual-centric methods recognize table structures solely through visual information while disregarding semantic content, which typically results in their suboptimal performance in relatively complex scenarios. Based on the recognition pipeline, these methods can be broadly categorized into the following three classes.

Split-and-Merge Methods. These methods typically involve two phases: initial segmentation into row/column grids followed by cell merging. Early approaches like SPLERGE [38] and SEM [50] use semantic line segmentation with merge models. DeepDeSRT [36] employs Faster R-CNN [35] and FCN [16] for joint detection/segmentation on electronic documents. RobusTabNet [22] introduces spatial CNN [27] for robust splitting. SEMv3 [32] improves efficiency by predicting key point offsets on table lines. TABLET [12] formulates splitting as sequence labeling and merging as cell classification via transformer encoders. Due to their reliance on table lines, these methods demonstrate limited robustness in scenarios involving wireless tables, occlusions, creases, or similar challenging conditions.

Detect-and-Classify Methods. This group of methods operate by detecting individual cells and predicting their relationships to reconstruct table structures. Early approaches [4, 30, 45] treat OCR text boxes as graph nodes, classifying relationships via graph neural networks. Subsequent methods focus on cell detection: Cycle-CenterNet [17], CascadeTabNet [29], and LGPMA [31] employ diverse techniques to enhance detection precision. TGRNet [46] formulates relationship classification as logical link prediction using Graph Convolutional Networks [7]. LORE [44] and LORE++ [18] leverage cascaded

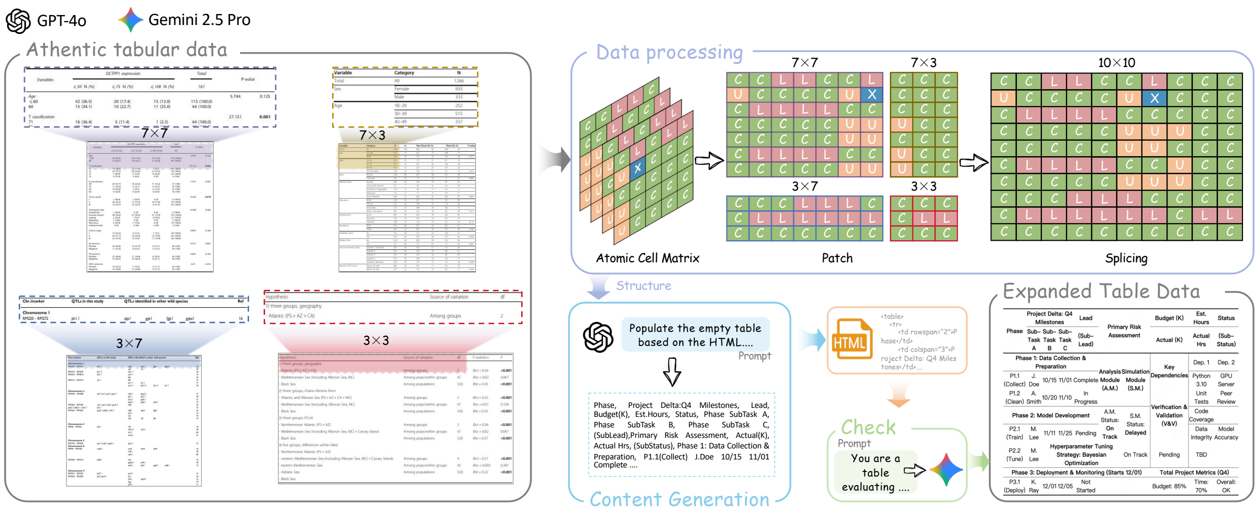

Figure 3. TME synthesis pipeline. Authentic table data undergoes matrix processing, partitioning, splicing, content generating and validity checking to generate novel synthetic tables for scenario-agnostic data expansion. In the atomic cell matrix, "C" denotes independent cells, "L" denotes left-merged cells, "U" denotes up-merged cells, and "X" denotes bidirectionally merged (left and up) cells.

transformers with multitask pretraining for physical/logical location prediction. These methods remain constrained by cell detection accuracy, often exhibiting degraded performance in wireless table and empty cell scenarios.

Image-to-Markup Methods. These methods directly generate structural sequences (e.g., HTML). EDD [53] pioneers the encoder-dual-decoder paradigm: CNN feature extraction followed by RNN-based sequential generation of structure tokens and cell text. TableMaster [47] and TableFormer [23] enhance decoding accuracy with Transformers, replacing text prediction with bounding box detection to reduce sequence length. VAST [13] introduces Visual-HTML Alignment to align image regions with decoder features. UniTable [28] pretrains its encoder via VQ-VAE [39] and generates sequences through three independent Transformers. SPRINT [14] deduces logical structures by predicting optimised table-structure language (OTSL) sequences [20] coupled with grid estimation [37]. Methods of this category are computationally intensive and data-hungry.

2.2. Vision-Language Model Based Methods

Vision-language models have demonstrated powerful capabilities across various image-text tasks. Beyond general-purpose large vision-language models (e.g., GPT-4o [25]), several research efforts have explored task-specific fine-tuning for table scenarios. The OmniParser series [40, 49] proposes a unified framework to simultaneously handle multiple text parsing tasks including TSR. TabPedia [51] integrates various visual table understanding tasks through a concept synergy mechanism. GOT [41] extends the OCR to recognize various optical signals, including tables, text, and other elements, under a unified character concept. MonkeyOCR [15] employs a Structure-Recognition-Relation (SRR) triplet paradigm, decomposing document parsing into layout analysis, content identification, and logical ordering to balance accuracy and efficiency. More recent work [6, 24, 34] employs a two-stage scheme of layout detection-content recognition for document parsing, while MinerU 2.5 and PaddleOCR-VL adopt OTSL for table representation to achieve higher efficiency and accuracy. Deepseek-OCR [42] achieves high-accuracy table parsing with a reduced number of visual tokens through optical 2D mapping compression and an end-to-end VLM architecture. Despite achieving superior performance in complex tabular scenarios through advanced semantic understanding, they may underperform visual-centric methods in scenarios requiring precise spacial information alignment.

3. Methodology

To address the dual challenges of scarce high-quality tabular data and models' imbalance between semantic and visual information, we propose: (1) TME, a template-free tabular data generation method based on real table structures and general LLM capabilities to generate tables at arbitrary scale; and (2) InstructTable, an end-to-end TSR approach leveraging instructions to enhance structural understanding.

3.1. Table Mix Expand

Inspired by OTSL, we employ an atomic cell matrix approach to generate template-free authentic tabular data at arbitrary scales. Compared to HTML, this representation reduces token counts by over 80% and generation lengths by approximately 50% while preserving layout flexibility. Crucially, we observe that atomic cell matrices exhibit crop-

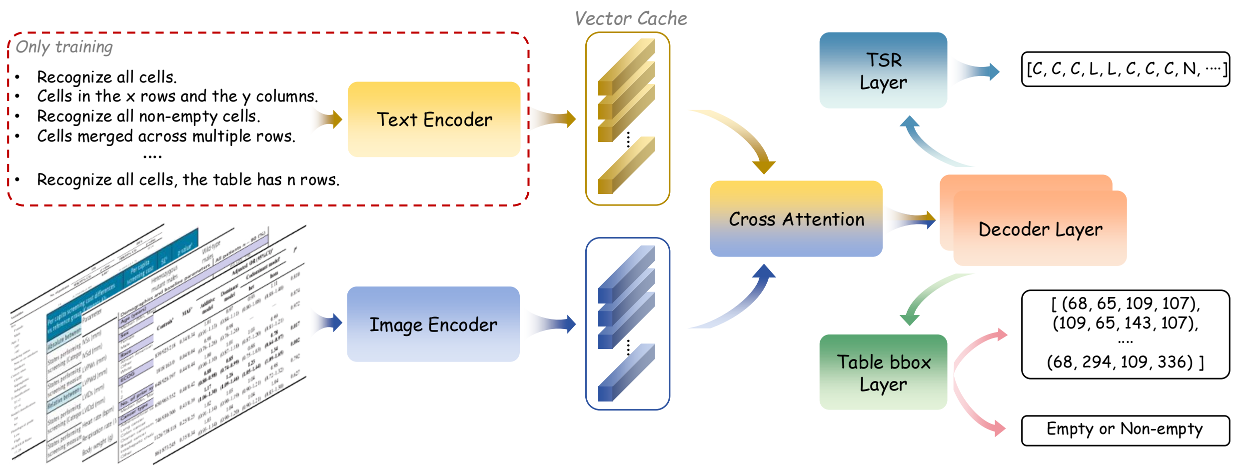

Figure 4. Main framework of InstructTable. The red dashed region indicates text encoding occurs only in training, where instruction embeddings are cached. During inference, cached vectors enable efficient processing without real-time text encoding.

ping invariance: any top-left submatrix remains valid, maintaining content topology and converting conflict-freely to HTML. Leveraging this property, our method: 1) initializes an empty matrix with randomized dimensions and partitions it into N blocks; 2) populates each block by selecting real data, cropping dimensionally compatible submatrices and introducing additional cell merges; 3) converts the matrix into a structural-token-only HTML table sequence, populates the table content using LLMs such as GPT-4o [25], and evaluates the validity and contextual semantic coherence of the generated table with Gemini 2.5 Pro [5], iterating this process until successful validation is achieved; 4) applies random CSS augmentation followed by image rendering. As illustrated in Fig. 3, this process generates tables that intrinsically preserve tabular structure authenticity while ensuring semantic coherence of table contexts, operating without template constraints.

3.2. Multi-Stage Training Strategy

A multi-stage training strategy is utilized to enhance joint modeling of visual information and instructional dependencies, thereby strengthening tabular parsing capabilities across diverse scenarios. The strategy primarily comprises initialization, instruction pre-training, and TSR fine-tuning. In initialization, the input instruction is fixed as “Recognize all cells” with several TSR training epochs, enabling the model to acquire fundamental TSR competence for improved subsequent training. Subsequently, we introduce human-designed fine-grained table understanding instructions to initiate a comprehensive instruction pre-training. During this stage, the model’s attention is directed by instructions toward diverse structural granularities, significantly enhancing perception capabilities for complex tabular scenarios including merged cells and empty cells.Finally, we execute domain-adaptive fine-tuning on downstream benchmarks to augment comprehensive visual representation learning, which is aligned with established VCTSR paradigms.

Table 1. Instruction templates for TSR in InstructTable. n and m denote table rows and columns respectively; R and C be subsets of $\{1, 2, \dots, n\}$ and $\{1, 2, \dots, m\}$, representing the randomly selected rows and columns; $x \in \{1, 2, \dots, n\}$ and $y \in \{1, 2, \dots, m\}$ indicate randomly chosen single row or column indices, ensuring high instruction diversity.

Group	Templates
(1)	*Recognize all cells.* *Recognize all cells, the table has n rows.* *Recognize all cells, the table has m columns.* *Recognize all cells, the table has n rows and m columns.*
(2)	*Cells in the R rows.* *Cells in the C columns.* *Cells in the x row and the y column.* *Cells around the cell in the x row and the y column.*
(3)	*Recognize all empty cells.* *Recognize all non-empty cells.*
(4)	*Cells merged across multiple rows.* *Cells merged across multiple columns.* *Cells merged across multiple rows and multiple columns.*

3.3. Instructions Design

Building on prior findings that text-image fusion modules align linguistic instructions with visual information to enhance OCR performance [9, 48], we design four groups of table-specific instructions: (1) recognition of table structure, (2) recognition of cells at specific locations, (3) recognition of empty cells, and (4) recognition of merged cells. The specific instructions for each group are comprehensively detailed in Tab. 1. These linguistic instructions enable context-sensitive structural parsing while addressing tabular complexities, including structural nuances like empty and merged cells.

During instruction pre-training, instructions are randomly selected. Each training sample x_i represented as a triplet $\{t_i, s_i, o_i\}$ where t_i denotes the instruction text, s_i represents the source image, and o_i is the target output. For every source image s_i, we apply the control instruction t_i to filter valid instances into the candidate set $O' = \{o'_0, o'_1, \ldots, o'_{n-1}\}$, then randomly sample a target instance $o_i \in O'$ as the generation objective conditioned on (t_i, s_i). During prediction, the instruction is unified as: "Recognize all cells".

3.4. Model Structure

We propose InstructTable, an end-to-end TSR framework that leverages instructions to modulate attention to enhance perception of fine-grained table structures. This approach improves structural understanding without introducing extraneous semantics. As illustrated in Fig. 4, the model jointly processes instructions and images to simultaneously generate three outputs: atomic cell matrices, cell bounding boxes, and empty cell classifications.

Image Encoder. The image encoder employs TableResNetExtra [47], a table-optimized CNN backbone with 26 convolutional layers and 3 GCA blocks to enhance tabular feature perception. This architecture processes 960×960 input images into 60×60 feature maps, augmented with 2D positional embeddings before flattening into 1D vectors for cross-attention modules.

Text Encoder. During training, the text encoder utilizes BERT [8] and a liner projector to encode human-designed instructions into 512-dimensional feature vectors. These text embeddings are fused with image features through cross-attention to produce instruction-conditioned representations. For inference efficiency, we precompute and cache text embeddings after training, since the input instructions remain fixed during prediction.

Decoder. The InstructTable decoder, built upon Transformer architecture, integrates three core components: instruction interaction via cross-attention, TSR layer generating atomic cell matrices, and bounding box prediction comprising coordinate regression and binary classification for empty cell detection. Offline OCR-derived text coordinates and content are fused with model output via matching rules to produce complete HTML representations, enabling efficient and accurate multilingual table parsing.

3.5. Loss Function

Multi-task learning is utilized on InstructTable to fit the model output to the corresponding data. For atomic cell matrix prediction and empty cell classification, a standard cross-entropy loss is used.

$$\mathcal{L}_{seq} = -\sum_{i=1}^{N} y_i \log(p_i) \tag{1}$$

$$\mathcal{L}_{box\_cls} = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(p_i) + (1-y_i)\log(1-p_i)] \tag{2}$$

where N denotes the number of samples in the batch, y_i represents the true labels of the samples, and p_i is the predicted probability distribution of the labels by the model. It is noteworthy that the empty cell classification is a binary classification task.

Moreover, our pipeline requires predicting the location of each cell to match with OCR results. Therefore, the L1 loss is used to constrain the position prediction, which is defined as follows:

$$\mathcal{L}_{box\_reg} = \sum_{i=1}^{N} \|x_i - \hat{x}_i\|_1 \tag{3}$$

where x_i denotes the ground truth position, $\hat{x}_i$ represents the predicted position.

Thus, the overall training loss is represented as follows:

$$\mathcal{L}_{total} = \alpha\mathcal{L}_{seq} + \beta\mathcal{L}_{box\_reg} + \gamma\mathcal{L}_{box\_cls} \tag{4}$$

where α, β, and γ are set to 1, 1, and 1, respectively.

4. Experiments

4.1. Datasets and Evaluation Metrics

We train and evaluate InstructTable on three public benchmarks: PubTabNet [53], FinTabNet [52], and MUSTARD [14]. These datasets collectively encompass diverse table layouts and multilingual content. PubTabNet provides 568K scientific publication tables with ground-truth HTML structural sequences, cell contents, and bounding box annotations. FinTabNet comprises 113K financial tables from S&P 500 annual reports in PDF format, featuring comparable structural and cell-level annotations. MUSTARD serves as an evaluation-only benchmark comprising 1,428 multilingual tables spanning 13 languages. Sourced from scanned/printed documents and scene images, all tables feature OTSL-formatted logical structures.

In addition, we report the results on our proposed benchmark, BCDSTab. BCDSTab comprises 900 table images synthesized via TME based on PubTabNet and FinTabNet. BCDSTab provides comprehensive annotations: HTML structures, atomic cell matrix representations, nested cell/content bounding boxes, and text content. Noting that existing public datasets predominantly feature small-scale tables with limited cell counts, they

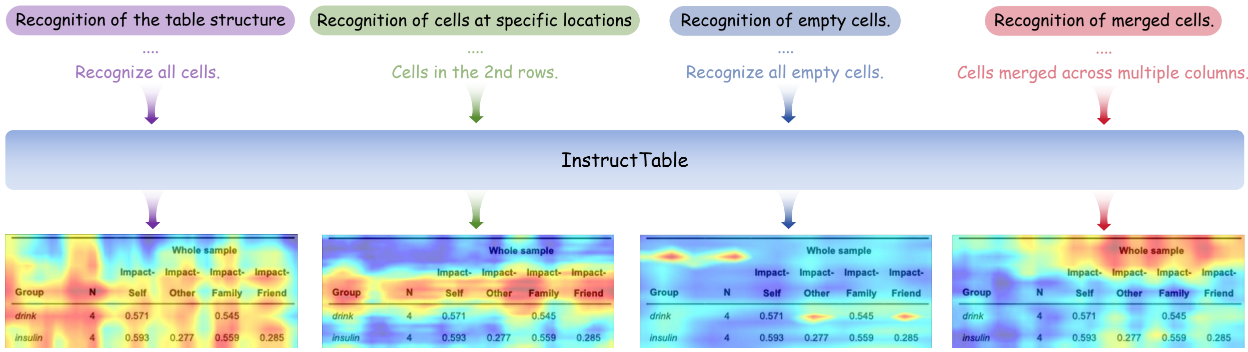

Figure 5. Attention heatmaps of the cross-attention layer under four instruction groups. Visualizations reveal how task-specific instructions dynamically modulate attention focus across table regions during parsing.

inadequately represent prevalent real-world long-form tables. This sparsity creates academia-industry misalignment. BCDSTab bridges this gap by balancing data distributions and accounting for real-world size variations, establishing a challenging yet practical benchmark for TSR. Benchmark details are provided in Section 1 of supplementary.

We employ Tree-Edit-Distance-based Similarity (TEDS) [53] and S-TEDS [13] to evaluate end-to-end and structure-only table recognition performance. These metrics represent table structures as HTML trees, computing similarity scores via normalized tree-edit-distance between prediction and ground truth. In structure-only evaluation scenarios (e.g., MUSTARD), we exclusively utilize S-TEDS, which excludes textual content matching.

4.2. Implement Details

For each training dataset, 50k synthetic data is generated by TME for training augmentation. Each generated table receives randomly sampled row and column dimensions between 4 and 20, partitioned into 4 patches, and populated by authentic data from the corresponding dataset only.

The decoder comprises 4 transformer decoder layers with 8 attention heads. Output sequences have the maximum length of 1024, representing tables up to approximately 30×30 cells. The input image size is 960×960, with model dimension fixed at 512.

InstructTable is trained distributively across 8 NVIDIA A100-80G GPUs using Adam optimizer. Both initialization and instruction pre-training stages utilize a combined dataset of PubTabNet, FinTabNet, and TME-generated data, with global batch size 160. The initialization stage employs a fixed learning rate of 0.008 for 10 epochs. During instruction pre-training, an initial learning rate of 0.01 is applied with step decay scheduling across 200 total epochs. Following sufficient pretraining, 50 epochs of TSR fine-tuning are conducted on each downstream dataset, maintaining identical configurations as the initialization stage.

4.3. Comparison with the State-of-the-Art Methods

Results of TSR. To ensure a fair comparison, we conduct two separate experiments on InstructTable: one using only public datasets during the entire training process and another combining public datasets with TME-generated data. All results evaluated by us are obtained from official checkpoints or online APIs. Comprehensive experimental details are provided in Section 2 of supplementary.

As demonstrated in Tab. 2, InstructTable achieves superior performance compared to all previous methods on FinTabNet, PubTabNet and MUSTARD. FinTabNet and PubTabNet containing predominantly English electronic documents, traditional visual-centric models outperform VLMs due to the relative simplicity of these scenarios and sufficient fine-tuning. When trained solely on public datasets, InstructTable shows comprehensive improvements over previous state-of-the-art models on both datasets. The incorporation of TME-generated data yields further performance gains, demonstrating that both proposed enhancements effectively strengthen the model's perception of table structures. On MUSTARD, characterized by its linguistic diversity and scenario richness, traditional visual-centric models struggle to achieve competitive performance. However, vision-language models, which are pre-trained on large-scale multimodal corpora, leverage their multilingual understanding capabilities to deliver superior results. Our evaluation shows that InstructTable achieves a TEDS score of 95.30% on MUSTARD, surpassing the previous state-of-the-art visual-centric method SPRINT by 10.11% and outperforming the strongest VLM Gemini 2.5 Pro by 1.65%. This robust performance across varied scenarios demonstrates the effectiveness of our approach.

On our BCDSTab benchmark for complex long tables, we evaluated several open-source and API-based solutions. When trained solely on public datasets, InstructTable outperformed competitors in S-TEDS but trailed MinerU 2.5 in TEDS by 1.19%. This gap stems from the domain discrep-

Table 2. TSR results on various datasets, respectively. InstructTable results are derived from pre-training on mixed data and fine-tuning on target datasets. Bolds denote the best and underlines denote the secondary best. * indicates results calculated by us from official checkpoints or online APIs. † denotes no incorporation of TME-generated data during the entire training process.

Methods	Param	FinTabNet		PubTabNet		MUSTARD	BCDSTab	
		S-TEDS	TEDS	S-TEDS	TEDS	S-TEDS	S-TEDS	TEDS
General Vision-Language Models								
GPT-4o* [25]	-	89.02	81.30	87.05	74.68	89.13	62.70	47.17
GPT-4.1* [26]	-	90.75	84.64	87.61	77.14	91.37	74.88	56.75
Qwen2.5-VL-72B* [2]	72B	87.75	79.86	85.95	76.36	89.00	78.88	68.36
Qwen3-VL-235B-A22B* [33]	235B	92.26	88.36	91.75	85.37	92.94	81.41	72.31
Seed1.5-VL* [11]	20B	84.46	68.14	84.85	68.55	90.69	70.07	62.03
Seed1.6* [3]	230B	85.31	71.74	86.14	70.13	93.35	78.97	62.55
InternVL3-78B* [54]	78B	90.07	76.44	88.17	75.90	88.31	69.68	46.51
Claude 3.7 Sonnet* [1]	-	92.93	81.98	87.93	72.68	91.28	70.27	62.05
Gemini 2.5 Pro* [5]	-	94.40	87.46	92.95	88.47	93.65	79.51	70.07
Table-Aware Vision-Language Models								
OmniParser [40]	-	90.45	88.83	91.55	89.75	-	-	-
TabPedia [51]	7B	95.41	-	95.11	-	-	-	-
OmniParser V2 [49]	-	93.20	90.50	90.50	88.90	-	-	-
MonkeyOCR* [15]	3B	89.56	81.40	94.61	90.08	69.86	76.16	67.90
dots.ocr [34]	1.7B	87.86	84.12	93.76	90.65	90.92*	78.43*	71.28*
MinerU2.5 [24]	1.2B	97.61	95.97	93.11	89.07	77.02*	82.72*	<u>74.07*</u>
PaddleOCR-VL* [6]	0.9B	95.03	93.12	89.82	85.09	67.65	78.08	72.71
DeepSeek-OCR* [42]	3B	96.48	95.43	90.69	86.44	80.93	62.98	48.72
Visual-Centric TSR Models								
EDD [53]	-	90.60	-	89.90	88.30	-	-	-
TableMaster [47]	68M	98.32	97.19	96.04	96.16	73.21*	59.17*	37.63*
TableFormer [23]	-	96.80	-	96.75	93.60	-	-	-
TableFormer+OTSL [20]	-	-	95.90	-	95.50	-	-	-
LORE [44]	23M	-	-	98.10	-	-	-	-
VAST [13]	-	98.63	98.21	97.23	96.31	-	-	-
MTL-TabNet [19]	76M	98.79	-	97.88	96.67	74.07	-	-
GridFormer [21]	-	98.63	-	97.00	95.84	-	-	-
UniTable [28]	382M	98.89	-	97.89	96.50	84.14*	76.22*	60.94*
SPRINT [14]	80M	98.03	-	95.71	-	85.19	-	-
TABLET [12]	-	98.71	98.54	97.67	96.79	-	-	-
InstructTable†	69M	<u>99.03</u>	<u>98.57</u>	<u>98.11</u>	<u>97.26</u>	<u>94.21</u>	<u>83.39</u>	72.88
InstructTable	69M	**99.20**	**98.68**	**98.22**	**97.46**	**95.30**	**84.75**	**75.45**

ancy between simple public data and the complex scenarios in BCDSTab, which means that InstructTable never encountered lengthy table images during training. In contrast, MinerU2.5 undergoes extensive training on over 1M large-scal, high-quality table data. Coupled with its billion-scale parameters that confer powerful generalization capabilities, it achieves superior performance on the complex scenarios of BCDSTab. After supplementing training with TME-generated data, InstructTable achieves a 2.57% TEDS improvement on BCDSTab, surpassing all existing solutions to set a new state-of-the-art.

Detailed visual analyses of different methods on various benchmarks are presented in Section 3 of supplementary.

TME Effectiveness Analysis. To verify the impact of TME-generated data on structural recognition, we conduct comparative experiments using both UniTable and InstructTable on FinTabNet and PubTabNet. For experimental consistency, all models were trained from scratch exclusively on their corresponding datasets, following the specifications of the original studies. Identical training procedures and data configurations are maintained across the four experimental groups. As detailed in Tab. 3, both models exhibit performance improvements when trained with TME-generated data. Across both datasets, the TEDS metric improves by an average of 1.2%, while S-TEDS exhibits a mean increase of 0.7%.

Table 3. Comparative results of UniTable and InstructTable. "TME" refers to the incorporation of TME-generated data.

Method	TME	FinTabNet		PubTabNet	
		S-TEDS	TEDS	S-TEDS	TEDS
Unitable	✘	96.91	95.10	95.00	93.23
Unitable	✓	97.75	95.85	96.09	94.83
InstructTable	✘	98.66	96.82	96.84	94.97
InstructTable	✓	**98.78**	**97.65**	**97.51**	**96.63**

4.4. Ablation Studies

Impact of Framework Components. In this section, we conduct a comparative analysis to evaluate the impact of the two core framework components: instruction pre-training and TME-generated data utilization. As evidenced in Tab. 4, both components positively impact model performance on both datasets. Instruction pre-training significantly improves structural parsing accuracy, while TME data enhances cell localization precision, yielding greater TEDS improvements. When combined, these components demonstrate synergistic effects, achieving average gains of +1.6% TEDS and +1.8% S-TEDS across the two datasets.

Table 4. Ablation study of our proposed components on FinTabNet and PubTabNet, "INS" and "TME" refer to utilization of instruction pre-training and the incorporation of TME-generated data.

INS	TME	FinTabNet		PubTabNet	
		S-TEDS	TEDS	S-TEDS	TEDS
✘	✘	97.11	96.57	95.91	94.03
✓	✘	98.66	96.82	96.84	94.97
✘	✓	98.37	97.07	96.22	95.01
✓	✓	**98.78**	**97.65**	**97.51**	**96.63**
		(+1.67)	**(+1.08)**	**(+1.60)**	**(+2.60)**

Impact of Distinct Instruction Groups. A set of experiments are conducted to investigate the impact of distinct instruction groups on InstructTable. Each group is individually removed from the instruction library for experimentation. The final results are presented in Tab. 5. The removal of any single group leads to a slight performance decline, with the first group, which guides the model to recognize the overall table structure, exhibiting a more significant impact (2.41% S-TEDS decrease). The results demonstrate that incorporating instructions for recognizing overall table structures is essential as they define the model's core output objective. Moreover, a sufficiently large and diverse instruction library facilitates balanced attention distribution and strengthen its understanding of fine-grained table structures.

Table 5. Ablation study of instruction groups on PubTbaNet without further fine-tuning. "G_1","G_2","G_3" and "G_4" refer to the corresponding instruction groups.

Instruction Group				PubTabNet
G_1	G_2	G_3	G_4	S-TEDS
✘	✓	✓	✓	93.93
✓	✘	✓	✓	96.21
✓	✓	✘	✓	96.04
✓	✓	✓	✘	96.05
✓	✓	✓	✓	**96.34**

Furthermore, we present quantitative results demonstrating the impact of varying instructions. As visualized in Fig. 5, which depicts cross-attention heatmaps for the same image under different instructions, InstructTable dynamically allocates attention to instruction-relevant regions. Unlike text-spotting models that exhibit localized attention patterns, our approach requires global image context to comprehend table structures, even when locating partial cell subsets. Consequently, while concentrating on instruction-specified cells, the model maintains holistic image attention rather than restricting focus exclusively to target cells.

5. Conclusion

In this paper, we propose InstructTable, a novel instruction-guided multi-stage training TSR framework that leverages human-designed instructions to enhance the understanding of fine-grained table structures. Through three training stages, the model integrates our meticulously crafted tabular-data-specific instruction sets with visual information to mitigate the visual-semantic imbalance observed in previous TSR methods. In addition, we introduce TME, a template-free tabular data synthesis method. By parsing tables into atomic cell matrices followed by mixed expansion, then employing LLMs for content population and validation, it enables large-scale, automatic construction of high-quality tabular data while preserving authenticity. Furthermore, leveraging this approach, we propose the BCDSTab benchmark, comprising 900 balanced complex dense tables for TSR capability evaluation. Extensive experiments demonstrate state-of-the-art performance on multiple benchmarks, with particularly gains in complex scenarios.

References

[1] Anthropic. Claude 3.7 sonnet and claude code. https://www.anthropic.com/news/claude-3-7-sonnet, 2025. 7

[2] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 2, 7

[3] ByteDance. Seed1.6 tech introduction. https://seed.bytedance.com/en/seed1_6, 2025. 2, 7

[4] Stéphane Clinchant, Hervé Déjean, Jean-Luc Meunier, Eva Maria Lang, and Florian Kleber. Comparing machine learning approaches for table recognition in historical register books. In *2018 13th IAPR International Workshop on Document Analysis Systems (DAS)*, pages 133–138. IEEE, 2018. 2

[5] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025. 2, 4, 7

[6] Cheng Cui, Ting Sun, Suyin Liang, Tingquan Gao, Zelun Zhang, Jiaxuan Liu, Xueqing Wang, Changda Zhou, Hongen Liu, Manhui Lin, et al. Paddleocr-vl: Boosting multilingual document parsing via a 0.9 b ultra-compact vision-language model. *arXiv preprint arXiv:2510.14528*, 2025. 3, 7

[7] Michaël Defferrard, Xavier Bresson, et al. Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in neural information processing systems*, 29, 2016. 2

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186, 2019. 5

[9] Chen Duan, Qianyi Jiang, Pei Fu, Jiamin Chen, Shengxi Li, Zining Wang, Shan Guo, and Junfeng Luo. Instructocr: Instruction boosting scene text spotting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2807–2815, 2025. 4

[10] Max Gobel, Tamir Hassan, Ermelinda Oro, and Giorgio Orsi. ICDAR 2013 Table Competition . In *2013 12th International Conference on Document Analysis and Recognition (ICDAR)*, pages 1449–1453, Los Alamitos, CA, USA, 2013. IEEE Computer Society. 1

[11] Dong Guo, Faming Wu, Feida Zhu, Fuxing Leng, Guang Shi, Haobin Chen, Haoqi Fan, Jian Wang, Jianyu Jiang, Jiawei Wang, et al. Seed1. 5-vl technical report. *arXiv preprint arXiv:2505.07062*, 2025. 7

[12] Qiyu Hou and Jun Wang. Tablet: Table structure recognition using encoder-only transformers. *arXiv preprint arXiv:2506.07015*, 2025. 2, 7

[13] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11134–11143, 2023. 3, 6, 7

[14] Dhruv Kudale, Badri Vishal Kasuba, Venkatapathy Subramanian, Parag Chaudhuri, and Ganesh Ramakrishnan. sprint: Script-agnostic structure recognition in tables. In *International Conference on Document Analysis and Recognition*, pages 350–367. Springer, 2024. 3, 5, 7

[15] Zhang Li, Yuliang Liu, Qiang Liu, Zhiyin Ma, Ziyang Zhang, Shuo Zhang, Zidun Guo, Jiarui Zhang, Xinyu Wang, and Xiang Bai. Monkeyocr: Document parsing with a structure-recognition-relation triplet paradigm. *arXiv preprint arXiv:2506.05218*, 2025. 3, 7

[16] Jonathan Long, Evan Shelhamer, et al. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 2

[17] Rujiao Long, Wen Wang, Nan Xue, Feiyu Gao, Zhibo Yang, Yongpan Wang, and Gui-Song Xia. Parsing table structures in the wild. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 944–952, 2021. 2

[18] Rujiao Long, Hangdi Xing, Zhibo Yang, Qi Zheng, Zhi Yu, Fei Huang, and Cong Yao. Lore++: Logical location regression network for table structure recognition with pre-training. *Pattern Recognition*, 157:110816, 2025. 2

[19] Nam Tuan Ly and Atsuhiro Takasu. An end-to-end multi-task learning model for image-based table recognition. *arXiv preprint arXiv:2303.08648*, 2023. 1, 7

[20] Maksym Lysak, Ahmed Nassar, Nikolaos Livathinos, Christoph Auer, and Peter Staar. Optimized table tokenization for table structure recognition. In *International Conference on Document Analysis and Recognition*, pages 37–50. Springer, 2023. 3, 7

[21] 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 *Proceedings of the 31st ACM International Conference on Multimedia*, pages 7747–7757, 2023. 7

[22] Chixiang Ma, Weihong Lin, Lei Sun, and Qiang Huo. Robust table detection and structure recognition from heterogeneous document images. *Pattern Recognition*, 133:109006, 2023. 2

[23] Ahmed Nassar, Nikolaos Livathinos, Maksym Lysak, and Peter Staar. Tableformer: Table structure understanding with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4614–4623, 2022. 1, 2, 3, 7

[24] Junbo Niu, Zheng Liu, Zhuangcheng Gu, Bin Wang, Linke Ouyang, Zhiyuan Zhao, Tao Chu, Tianyao He, Fan Wu, Qintong Zhang, et al. Mineru2. 5: A decoupled vision-language model for efficient high-resolution document parsing. *arXiv preprint arXiv:2509.22186*, 2025. 3, 7

[25] OpenAI. Hello gpt-4o. https://openai.com/index/hello-gpt-4o, 2024. 3, 4, 7

[26] OpenAI. Introducing gpt-4.1 in the api. https://openai.com/index/gpt-4-1/, 2025. 7

[27] Xingang Pan, Jianping Shi, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Spatial as deep: Spatial cnn for traffic scene understanding. In *Proceedings of the AAAI conference on artificial intelligence*, 2018. 2

[28] ShengYun Peng, Aishwarya Chakravarthy, Seongmin Lee, Xiaojing Wang, Rajarajeswari Balasubramaniyan, and Duen Horng Chau. Unitable: Towards a unified framework for table recognition via self-supervised pretraining. *arXiv preprint arXiv:2403.04822*, 2024. 3, 7

[29] Devashish Prasad, Ayan Gadpal, Kshitij Kapadni, Manish Visave, and Kavita Sultanpure. Cascadetabnet: An approach for end to end table detection and structure recognition from image-based documents. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 572–573, 2020. 2

[30] Shah Rukh Qasim, Hassan Mahmood, et al. Rethinking table recognition using graph neural networks. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 142–147. IEEE, 2019. 2

[31] Liang Qiao, Zaisheng Li, Zhanzhan Cheng, Peng Zhang, Shiliang Pu, Yi Niu, Wenqi Ren, Wenming Tan, and Fei Wu. Lgpma: Complicated table structure recognition with local and global pyramid mask alignment. In *International conference on document analysis and recognition*, pages 99–114. Springer, 2021. 2

[32] Chunxia Qin, Zhenrong Zhang, Pengfei Hu, Chenyu Liu, Jiefeng Ma, and Jun Du. Semv3: A fast and robust approach to table separation line detection. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, pages 1191–1199. International Joint Conferences on Artificial Intelligence Organization, 2024. Main Track. 2

[33] Qwen. Qwen3-vl: Sharper vision, deeper thought, broader action. https://qwen.ai/blog?id=99f0335c4ad9ff6153e517418d48535ab6d8afef&from=research.latest-advancements-list, 2025. 7

[34] rednote. dots.ocr: Multilingual document layout parsing in a single vision-language model. https://github.com/rednote-hilab/dots.ocr, 2025. 2, 3, 7

[35] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015. 2

[36] Sebastian Schreiber, Stefan Agne, Ivo Wolf, Andreas Dengel, and Sheraz Ahmed. Deepdesrt: Deep learning for detection and structure recognition of tables in document images. In *2017 14th IAPR international conference on document analysis and recognition (ICDAR)*, pages 1162–1167. IEEE, 2017. 2

[37] Brandon Smock, Rohith Pesala, et al. Pubtables-1m: Towards comprehensive table extraction from unstructured documents. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4634–4642, 2022. 3

[38] Chris Tensmeyer, Vlad I Morariu, Brian Price, Scott Cohen, and Tony Martinez. Deep splitting and merging for table structure decomposition. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 114–121. IEEE, 2019. 2

[39] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017. 3

[40] Jianqiang Wan, Sibo Song, Wenwen Yu, Yuliang Liu, Wenqing Cheng, Fei Huang, Xiang Bai, Cong Yao, and Zhibo Yang. Omniparser: A unified framework for text spotting key information extraction and table recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15641–15653, 2024. 3, 7

[41] Haoran Wei, Chenglong Liu, Jinyue Chen, Jia Wang, Lingyu Kong, Yanming Xu, Zheng Ge, Liang Zhao, Jianjian Sun, Yuang Peng, et al. General ocr theory: Towards ocr-2.0 via a unified end-to-end model. *arXiv preprint arXiv:2409.01704*, 2024. 2, 3

[42] Haoran Wei, Yaofeng Sun, and Yukun Li. Deepseek-ocr: Contexts optical compression. *arXiv preprint arXiv:2510.18234*, 2025. 3, 7

[43] Anyi Xiao and Cihui Yang. Tablecenternet: A one-stage network for table structure recognition. *arXiv preprint arXiv:2504.17522*, 2025. 1

[44] Hangdi Xing, Feiyu Gao, Rujiao Long, Jiajun Bu, Qi Zheng, Liangcheng Li, Cong Yao, and Zhi Yu. Lore: logical location regression network for table structure recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2992–3000, 2023. 2, 7

[45] Wenyuan Xue, Qingyong Li, et al. Res2tim: Reconstruct syntactic structures from table images. In *2019 international conference on document analysis and recognition (ICDAR)*, pages 749–755. IEEE, 2019. 2

[46] Wenyuan Xue, Baosheng Yu, Wen Wang, Dacheng Tao, and Qingyong Li. Tgrnet: A table graph reconstruction network for table structure recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1295–1304, 2021. 2

[47] 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. 1, 3, 5, 7

[48] Jiabo Ye, Anwen Hu, Haiyang Xu, Qinghao Ye, Ming Yan, Yuhao Dan, Chenlin Zhao, Guohai Xu, Chenliang Li, Junfeng Tian, et al. mplug-docowl: Modularized multimodal large language model for document understanding. *arXiv preprint arXiv:2307.02499*, 2023. 4

[49] Wenwen Yu, Zhibo Yang, Jianqiang Wan, Sibo Song, Jun Tang, Wenqing Cheng, Yuliang Liu, and Xiang Bai. Omniparser v2: Structured-points-of-thought for unified visual text parsing and its generality to multimodal large language models. *arXiv preprint arXiv:2502.16161*, 2025. 3, 7

[50] Zhenrong Zhang, Jianshu Zhang, Jun Du, and Fengren Wang. Split, embed and merge: An accurate table structure recognizer. *Pattern Recognition*, 126:108565, 2022. 2

[51] Weichao Zhao, Hao Feng, Qi Liu, Jingqun Tang, Binghong Wu, Lei Liao, Shu Wei, Yongjie Ye, Hao Liu, Wengang

Zhou, et al. Tabpedia: Towards comprehensive visual table understanding with concept synergy. *Advances in Neural Information Processing Systems*, 37:7185–7212, 2024. 3, 7

[52] 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 *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 697–706, 2021. 5

[53] Xu Zhong et al. Image-based table recognition: data, model, and evaluation. In *European conference on computer vision*, pages 564–580. Springer, 2020. 1, 3, 5, 6, 7

[54] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025. 7