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,

1. BCDSTab Benchmark Detailed Specifications

The Balanced Complex Dense Synthetic Tables (BCDSTab) benchmark comprises 900 dense table images synthesized through our TableMixExpand (TME) framework, with source data derived from FinTabNet [8] and PubTabNet [9]. During synthesis, we first sample the total cell count C from a normal distribution $\mathcal{N}$ bounded within $[4, 1000]$. Next, we uniformly select the row count R from discrete integers in $[2, 100]$. The column count K is then computed via integer division: $K = \lfloor C/R \rfloor$. If K falls outside the valid range $[2, 15]$, we iteratively resample C and R until satisfying $2 \leq K \leq 15$. This rejection sampling procedure initializes an empty atomic cell matrix ready for content generation.

Subsequently, the initialized matrix is partitioned into four patches according to its row-column dimensions. We randomly sample four authentic tables from the hybrid corpus combining FinTabNet and PubTabNet. For each source table, a top-left submatrix matching the corresponding patch dimensions is extracted. When source dimensions are insufficient, tables are resampled until meeting size requirements. These submatrices are then populated into their respective patches within the target matrix, resulting in a populated atomic cell matrix with complete structural, while without any text content.

Upon obtaining the complete atomic cell matrix, it is converted into a minimal HTML representation containing only table structure elements. This processed HTML is then combined with the prompt "Populate the empty table based on the HTML provided. Return a complete table! Ensure the table structure exactly match the empty table provided!" and fed into GPT-4o [6], leveraging the extensive knowledge of this general-purpose large language model to generate contextually appropriate textual content based on the table structure.Then we employ the powerful reasoning model Gemini-2.5 Pro [1] to verify the synthetic table' validity and contextual coherence, using the prompt: "You are a table evaluating expert, you will receive an HTML-formatted table to verify both its structural compliance and the contextual coherence of its content."

We then apply randomized CSS augmentations to the HTML, incorporating stochastic variations in:

- Text alignment direction (left/center/right/justify)
- Font families (12 serif/sans-serif options)
- Font sizes (range: 10-25pt)
- Vertical padding (2-12px)
- Border styles (solid/dashed/dotted/double/none)
- Table line types (single/double/hidden)
- Text colors
- Cell background colors

The augmented HTML is rendered on a Linux server using headless Chrome (v91.0.4472.77), constrained by maximum dimensions of 5000 pixels in height and 3000 pixels in width, with a minimum font height requirement of 12 pixels. Samples violating these constraints are discarded. Chrome's rendering engine enables precise element localization via XPath queries, yielding accurate pixel coordinates for all cell boundaries.

For non-empty cells, we perform text bounding box extraction through computer vision processing:

1. Convert cell sub-image to binary format
2. Detect text contours using OpenCV's `boundingRect`
3. Map local coordinates to global image
4. Output cell content bounding boxes in $[x_{min}, y_{min}, x_{max}, y_{max}]$ format matching FinTabNet/PubTabNet specifications

The controlled color differentiation ensures reliable binarization. This process yields the complete BCDSTab benchmark, comprising 1,000 dense table images, HTML ground-truth representations, atomic cell matrix structures, cell-level bounding boxes, and content-level bounding boxes. Sample visualizations of the benchmark are presented in Figure Fig. 1.

Furthermore, we conduct quantitative comparisons across multiple dimensions between BCDSTab and leading public table datasets. Specifically, FinTabNet, PubTab-

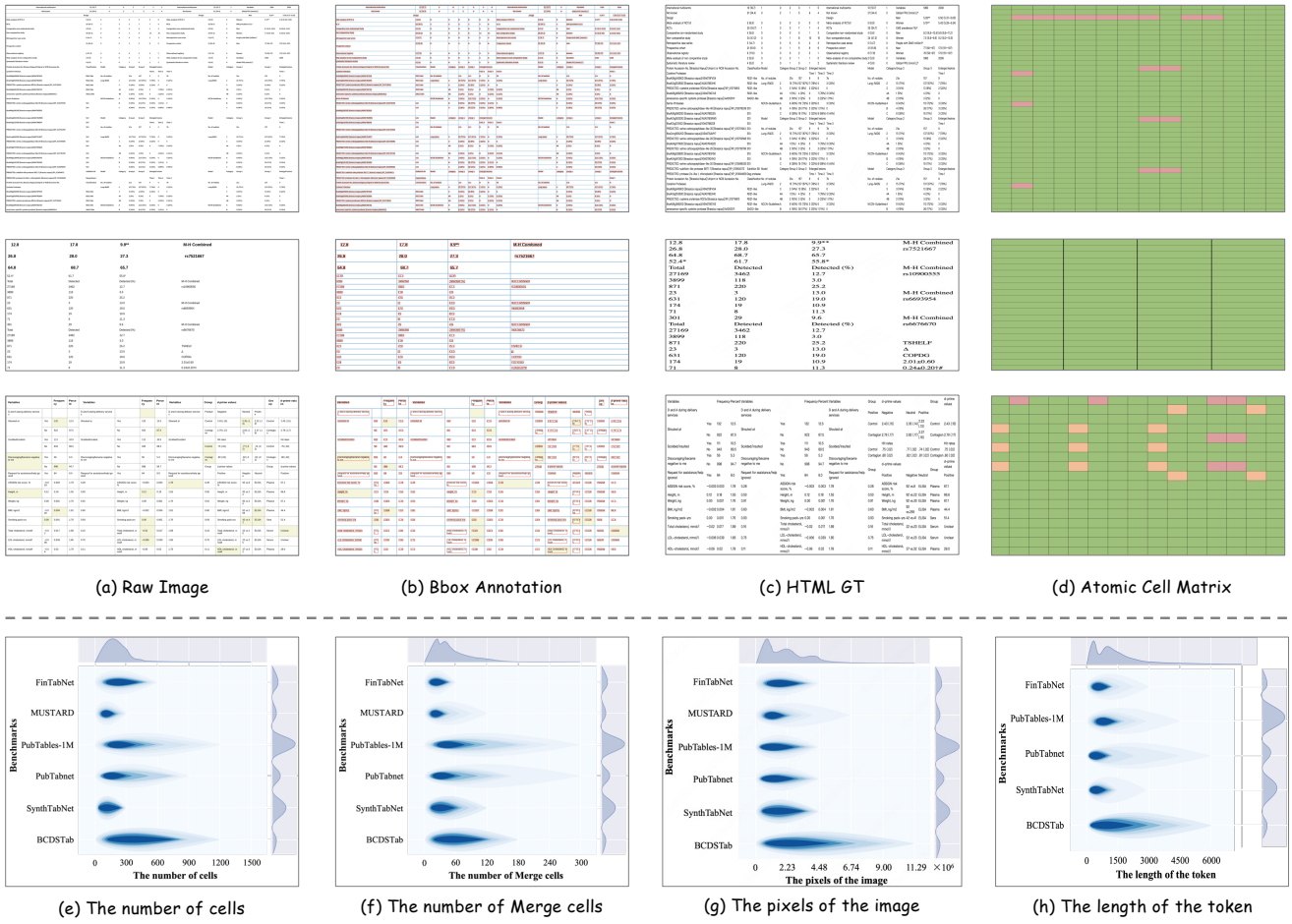


Figure 1. An overview of BCDSTab benchmark. (a) Representative image sample; (b) Dual-level bounding box visualization with blue indicating cell-level boundaries and red showing content-level text regions; (c) HTML ground truth representation for metric evaluation; (d) Atomic cell matrix color-coded by merge patterns: green for normal cells, red for left-merged cells, yellow for up-merged cells; (e)-(h) Multidimensional data statistics.

Net, PubTables-1M [7], SynThTabNet [4], and MUSTARD [3] with their test sets. As quantified in Figure Fig. 1, BCDSTab demonstrates superior distribution balance and broader value ranges in both cell counts and merged cell quantities, fulfilling our objective to evaluate TSR models under dense, long-table scenarios. BCDSTab provides images with significantly higher pixel counts than existing datasets, delivering enhanced visual detail to support flexible downstream processing by users.

To prevent truncation of vision-language model outputs, we analyze token counts of full HTML representations across datasets using the BERT tokenizer [2]. The maximum observed token length was 5867, prompting our configuration of the vision-language model's output limit at 8192 tokens, ensuring complete table generation capacity. Notably, MUSTARD is excluded from this analysis as it provides only structural annotations without text content.

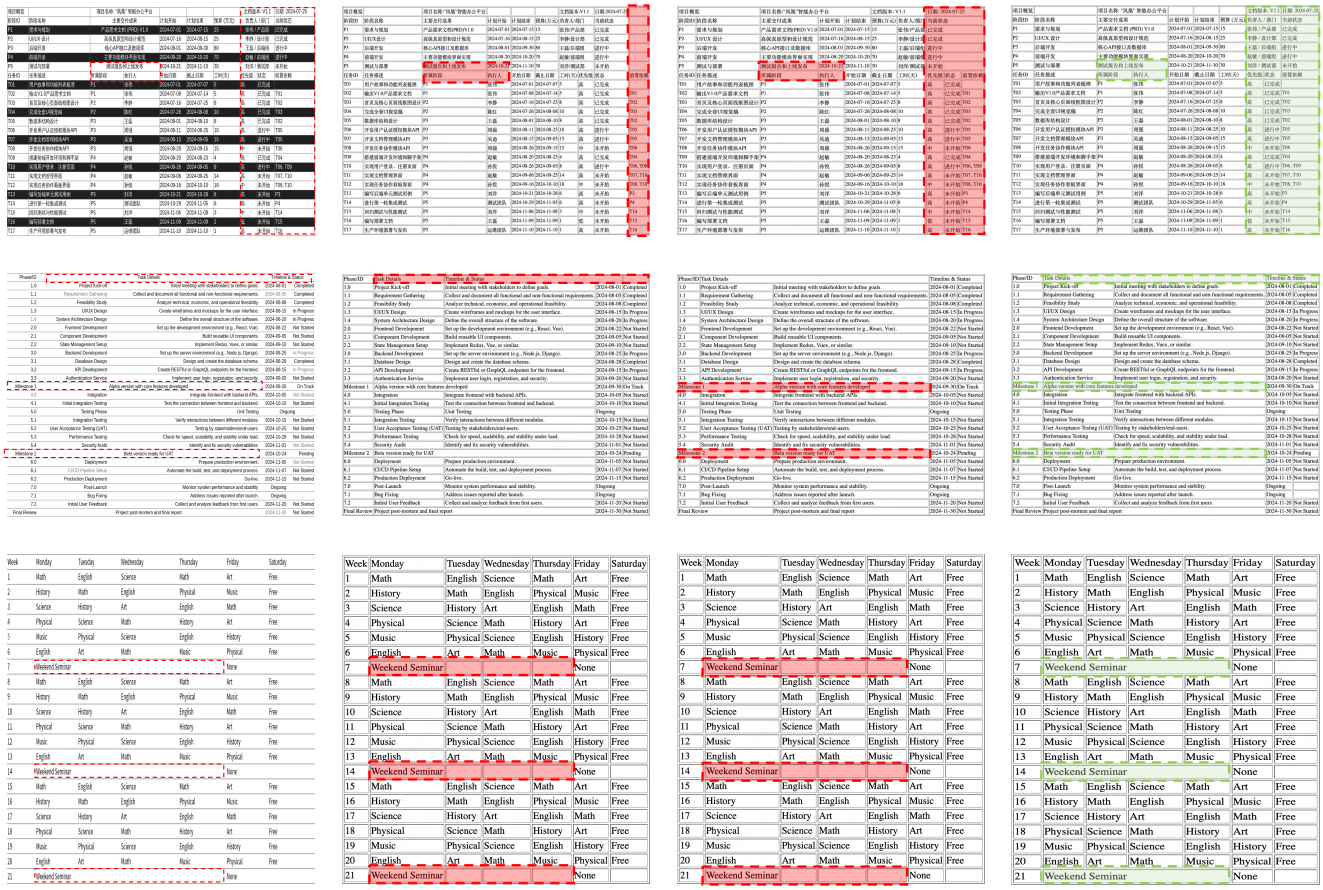
2. Vision-Language Model Experimental Setup

In our experimental evaluation, we conduct extensive testing across multiple vision-language models while maintaining fixed configurations for each model to ensure fairness, with detailed settings specified as follows:

1. We fix the prompt to "This is an image containing only one table, please convert the table in the image to HTML (begin with ⟨table⟩ and end with ⟨/table⟩) format. Only the content in the image needs to be output without expanding other content." uniformly across all general VLMs.
2. We set the maximum output length to 8192 tokens, which is sufficient for generating complete HTML table outputs across all datasets.

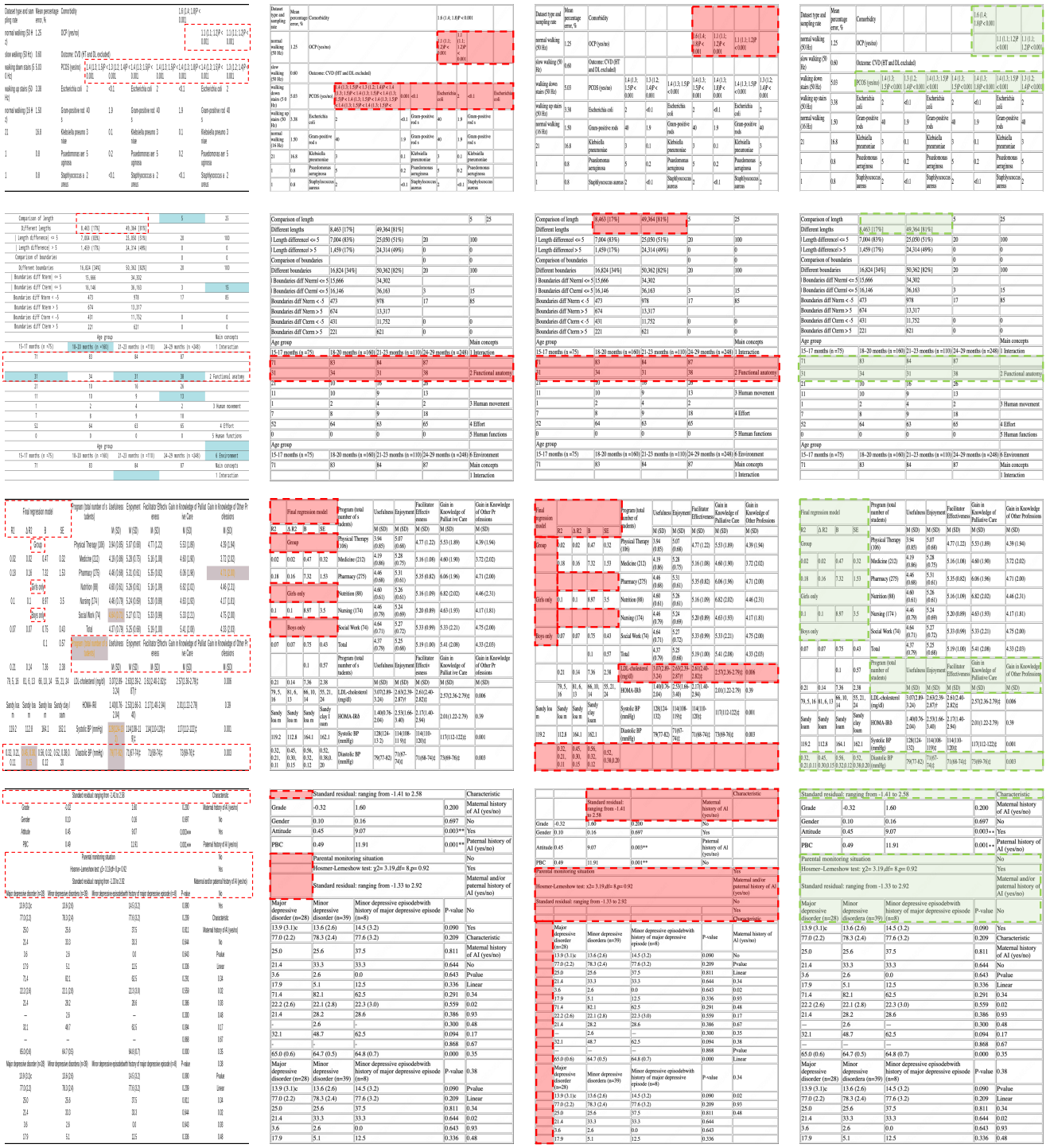
3. We systematically employ regular expressions to extract clean table content enclosed within "$\langle\mathrm{table}\rangle$" and "$\langle/\mathrm{table}\rangle$" tags from model outputs, ensuring effective isolation from extraneous textual elements.
4. For reasoning-capable models (e.g., Gemini 2.5 Pro), we consistently enable their think mode during inference.
5. For the multi-stage Table-aware VLMs (e.g., MinerU2.5 [5]), we isolate their table structure recognition module separately and employ official parameters and prompts to perform table structure recognition.

3. Case Study



(a) Input image (b) MinerU2.5 (c) Gemini 2.5 Pro (d) InstructTable

Figure 2. Visual comparison of different methods on BCDSTab, including: (a) input image, (b) MinerU2.5 results, (c) Gemini 2.5 Pro results, and (d) our InstructTable results. Red regions highlight erroneous positions in the predictions, with corresponding areas marked by green zones in our predictions and delineated by red dashed boxes in the input image for intuitive comparison.



(a) Input image (b) MinerU2.5 (c) Gemini 2.5 Pro (d) InstructTable

Figure 3. Visual comparison of different methods on BCDSTab, including: (a) input image, (b) MinerU2.5 results, (c) Gemini 2.5 Pro results, and (d) our InstructTable results. Red regions highlight erroneous positions in the predictions, with corresponding areas marked by green zones in our predictions and delineated by red dashed boxes in the input image for intuitive comparison.

References

[1] 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. 1

[2] 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. 2

[3] 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. 2

[4] 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. 2

[5] 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

[6] OpenAI. Hello gpt-4o. https://openai.com/index/hello-gpt-4o, 2024. 1

[7] 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. 2

[8] 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. 1

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