This ICCV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Parsing Table Structures in the Wild

Rujiao Long*[3], Wen Wang*[1,2], Nan Xue[1] Feiyu Gao[3],
Zhibo Yang[3], Yongpan Wang[3], Gui-Song Xia†[1,2]
[1] *School of Computer Science, Wuhan University, Wuhan, China*
[2] *LIESMARS, Wuhan University, Wuhan, China*
[3] *Alibaba-Group, Hangzhou, China*
https://github.com/wangwen-whu/WTW-Dataset

Abstract

This paper tackles the problem of table structure parsing (TSP) from images in the wild. In contrast to existing studies that mainly focus on parsing well-aligned tabular images with simple layouts from scanned PDF documents, we aim to establish a practical table structure parsing system for real-world scenarios where tabular input images are taken or scanned with severe deformation, bending or occlusions. For designing such a system, we propose an approach named Cycle-CenterNet on the top of CenterNet with a novel cycle-pairing module to simultaneously detect and group tabular cells into structured tables. In the cycle-pairing module, a new pairing loss function is proposed for the network training. Alongside with our Cycle-CenterNet, we also present a large-scale dataset, named Wired Table in the Wild (WTW), which includes well-annotated structure parsing of multiple style tables in several scenes like photo, scanning files, web pages, etc.*. In experiments, we demonstrate that our Cycle-CenterNet consistently achieves the best accuracy of table structure parsing on the new WTW dataset by 24.6% absolute improvement evaluated by the TEDS metric. A more comprehensive experimental analysis also validates the advantages of our proposed methods for the TSP task.*

1. Introduction

Tables are commonly used in our daily life to record and summarize important data for quick and better visualization of information. With the increasing popularity of smartphones and portable cameras, it is very common to share information with photo of tables. Accordingly, it is highly demanded to automatically extract and parse table structures from photos or images in the wild.

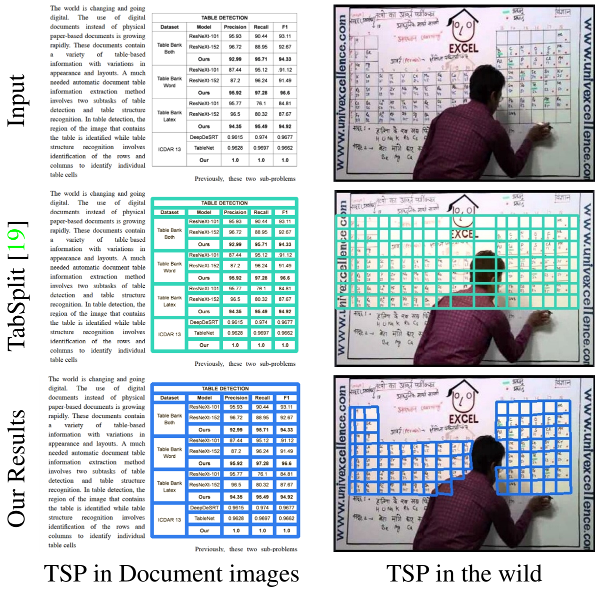


Figure 1. Visual comparison for the difference between the problem of table structure parsing (TSP) in document images and the images taken in the wild. We leverage the state-of-the-art approach for document images proposed in [19] and our proposed Cycle-CenterNet for both input images to obtain the parsing results.

Given an image, *Table Structure Parsing* (TSP) aims at extracting all the tables, locating their cells, and obtaining the row-column information in the image. Previously, this problem is studied as table structure recognition focusing on document images. In such scenario, the tabular images are taken with well imaging conditions and are often horizontally (or vertically) aligned with clean background and clear table structures. Early pioneering works, *e.g*. [7, 8, 20, 6], tackle the TSP problem in a bottom-up manner by heuristically grouping detected cells based on low-level cues (*e.g*., lines, boundaries and word regions). Recently, deep learning-based approaches are presented to avoid the heuristic grouping scheme design and resort to de-

*Equal Contribution.
†Correspondence Author.

veloping end-to-end models. However, limited by the training datasets [9, 17, 2, 24, 5] used for table structure parsing, they still addressed this problem under the well-aligned assumption of tabular images.

For a more practical requirement of parsing table structures from images taken by hand-held cameras in the wild, the existing state-of-the-art approaches [13, 14, 11, 16, 9, 23] are prone to fail as the commonly-used assumption of tabular images no longer holds. Specifically, the tabular images in the widely-used datasets (*e.g.*, ICDAR-2013 [5], Tablebank [9]) are usually with clean background and clear table structures. Limited by this, existing TSP approaches can only handle table structure parsing in a relative simple scenario by grouping detected cells into tables [11, 16, 9, 23]. Moreover, few research pays attention to the precision of cell boundary, which is important in text recognition.

To tackle the TSP problem in the wild, we present a large-scale dataset in this paper to address the data lacking issue. When we collecting the real-scene tabular images, we found that the wired tables and wireless tables have a very large difference. The wireless tables in natural images are very challenging to be recognized as the lacking of reference for perceptual grouping by human annotators. Therefore, we mainly focus on the challenging wired tables for annotation. Our proposed dataset, the *Wired Tables in the Wild* (WTW), contains 14,581 images with the annotated information of *table id*, *tabular cells* and *corresponding row/column information*. Following the data splitting strategy used in ICDAR 2019 [3], we split our WTW into *training*/*testing* subsets with 10,970 and 3611 data samples respectively.

As shown in Fig. 1, the images in the WTW dataset are very different from the document images, which thus poses a new problem to the table structures parsing task. For instance, the non-rigid image deformation and complicated image background presented in natural images will challenge existing approaches [14] for document images on detecting and grouping the tabular cells.

With our proposed WTW dataset available, we address the problem of table structure parsing in the wild by proposing a simple yet effective approach Cycle-CenterNet. It simultaneously detects the vertices and center points of tabular cells, and groups the cells into tables by learning the common vertices. Specifically, we found that the center point and vertices of a cell have a mutual-directed relationship that can be used to group the cells into tables by using the common vertex that is located in the intersect of the adjacent cells. Based on this, we propose a loss function named *pairing loss* to end-to-end group the cells in training phase. Once the structures of tables are obtained, we use a simple post-processing algorithm to retrieve the row and column information for the parsed tables. In experiments, we evaluate the proposed Cycle-CenterNet on WTW dataset. Compared with the strong baseline of vanilla CenterNet, our approach largely improves the F1-score of physical coordinate accuracy from 73.1% to 78.3%, while improves the F1-score of adjacent relationship estimation from 84.8% to 92.4%. In the metric of TEDS [24], the proposed Cycle-CenterNet also obtains an absolute improvement by 24.6 points.

Our contributions are summarized as follows:

- We build a large-scale dataset in wild complex scenes, which provides a variety of new challenges for table structure parsing with several real image distortions.

- We present an approach Cycle-CenterNet by exploiting the cycle-pairing module optimization with a novel pairing loss proposed, which enables us to precisely group the discrete cells into the structured tables.

- In the experiments, our method improves the performance of table structure parsing on the WTW dataset by large margins. It also outperforms the state-of-the-art methods on the ICDAR2019 dataset, and achieves competitive results on the ICDAR2013 dataset.

2. Related Work

2.1. Existing Datasets

There are a number of datasets including UNLV [17], ICDAR-2013 [5], SciTSR [2], PubTabNet [24] and Table-Bank [9] etc. that are available in table structure parsing. Prior to the deep table structure parsing approaches emerged, the dataset UNLV [17] and ICDAR-2013 [5] were designed for benchmarking the table structure recognition systems with a limited number (less than 1,000) of tabular images and annotations. To meet the requirement of designing data-driven learning approaches for table structure parsing, large-scale datasets of PubTabNet [24] and Table-Bank [9] are proposed, but the incomplete annotations still hinder their development. Recently, FinTabNet [23] and SciTSR [2] datasets add the cell coordinates and row-column information to become the most complete and large-scale dataset for table structure parsing task.

Although the scale of table image datasets has been dramatically improved, these datasets are with a specific focus on the document images that are obtained from the digital documents (*e.g.*, PDF documents). For our purpose of parsing table structures in the wild, those datasets cannot be used for training a learning-based approach with an expected generalization ability. Recently, a new dataset ICDAR2019 [3] introduce a more challenging task of parsing table structures from the scanned archival documents instead of the digital documents. However, it only contains 750 images for table structure parsing, which will induce the same problem for training a data-driven table structure parsing model. Besides, the ICDAR2019 [3] dataset is still focused on the document images. In contrast to those existing datasets, we contribute a new large-scale table dataset WTW that contains 14,581 complex wired tables in multiple real scenes including photoing, scanning, and web pages. Different from the existing datasets, the images in our proposed dataset usually contain severe practical image distortions including bending, tilting, and occlusion, *etc*.

2.2. Table Structure Recognition and Parsing

The problem of table structure parsing was previously studied as two sub-problems of table detection and table structure recognition. Kieninger *et al*. [8] presented the first table structure recognition system that estimates the table structures by clustering the detecting the text blocks from tabular images in a heuristic way. Following this work, the rule-based heuristic approaches were proposed [21, 18] to recognize or parse table structures from the hand-crafted visual cues. Recently, the deep learning-based approaches were proposed to automatically learn informative visual features [5, 2, 24, 12]. However, these methods mainly focuses on the well-conditioned document images, where the tables [16, 23, 12, 9, 23, 14] are well-aligned to the image axes. As this assumption does not hold on to more challenging tables, some researches tried to get rid of the well-aligned assumption and modeled the table structure parsing problem with graph convolution networks [2, 13, 24]. However, they are implicitly using the adjacent relationship between the detected cells to construct an informative initial graph and then prune the unexpected edges during training. Different from those approaches, our proposed Cycle-CenterNet gets rid of using the assumptions mentioned above to meet more practical requirements for table structure parsing in the wild.

3. The WTW Dataset

This section presents the detail of our proposed dataset, *Wired Tables in the Wild* (WTW), which has a total of 14581 images in a wide range of real business scenarios and the corresponding full annotation (including cell coordinates and row/column information) of tables.

3.1. Image Collection and Annotation

The images in the WTW dataset are mainly collected from the natural images that contain at least one table. As our purpose is to parsing table structures without considering the image source, we additionally add the archival document images and the printed document images. Statically, the portion of images from natural scenes, archival, and printed document images are 50%, 30%, and 20%. After obtaining all the images, we statically found 7 challenging cases. As summarized in Tab. 1, our proposed WTW dataset covers all challenging cases with a reasonable proportion of each case.

Table 1. A statistical summary and comparison between our WTW dataset and the existing datasets for table structure parsing. Our proposed dataset covers all the 7 challenging cases of (1) Inclined tables, (2) Curved tables, (3) Occluded tables or blurred tables (short in *Occ. or Blur*, (4) Extreme aspect ratio tables (short in *Ex. AR*), (5) Overlaid tables, (6) Multi-color tables and (7) Irregular tables in table structure recognition. In the last row, we report the total number of samples for all those datasets.

Challenging Cases	Tablebank [9]	UNLV [17]	Marmot [11]	SciTSR [2]	ICDAR-13 [5]	ICDAR-19 [3]	WTW (Ours)
Inclined	-	✓	-	-	-	✓	✓
Curved	-	-	-	-	-	-	✓
Occ. or Blur	-	-	-	-	-	✓	✓
EX. AR	-	-	-	-	-	-	✓
Overlaid	-	-	-	-	-	-	✓
Multi Color	✓	-	-	-	✓	-	✓
Irregular	✓	-	✓	-	-	✓	✓
# Samples	145,000	423	1,000	15,000	156	750	14,581

In our dataset, we annotate all the tables presented in each image for their cell coordinates and the row/column information. For the images that have more than one table, their instance information is also annotated. When annotating the cell coordinate, we follow the benchmark of ICDAR2019 [3] to use the inner table lines for localization. To ensure that there is no leakage of sensitive information (names, telephone numbers, etc.), we erased them out.

Data splitting. In order to ensure that the training data and test data distributions approximately match, we randomly select approximately 75% of the original images as the training set, and the rest data samples are used for testing and evaluation. Finally, our WTW dataset has 10970 training samples and 3611 testing ones.

3.2. Baselines and Benchmark Evaluation

As our dataset contains a large number of tables that have deformations (*e.g*., the inclined, curved, and irregular tables), the commonly-used rectangular representation of tables cannot be generalized well to those challenging cases in our dataset. As a result, it would be problematic to directly leverage the state-of-the-art data-driven approaches designed for documents in our dataset. Accordingly, we present a more appropriate way to set up the baseline approaches and provide a comprehensive evaluation protocol to benchmark the new approaches on the WTW dataset.

Baseline configuration. Instead of modeling the table structures in natural images as large rectangles, we first represent the tabular cells as small objects since the small objects are more robust to the severe image deformation. Based on this, we formulate the problem of table structure parsing in two steps: (1) using a state-of-the-art object detector for cell detection, and then (2) grouping the detected cells into tables by heuristically calculate the spatial proximity between cells. After obtaining the table structures, a post-processing step is applied to the row/column information[*]. To make the baseline configuration more convincing, we use four widely-used object detectors of Faster-RCNN [15], TridenNet [10], Cascade-RCNN [1] and the anchor-free detector CenterNet [25] as the cell detector in our baseline.

Evaluation protocol for table structure parsing. A reasonable evaluation protocol is important for quantitatively compare different approaches. We evaluate a given table structure parser in two aspects of (1) *the correctness of physical structure* and (2) *the correctness of logical structure*, described as follow:

- *Precision, Recall and F-score for physical structure estimation.*

  We evaluate the accuracy of cell detection by calculating the precision, recall, and F1-score for the parsing results with regard to the ground truth in the testing split of the WTW dataset. Different from the general object detection, the table structure parsing requires more accuracy of tabular cells with low tolerance. Therefore, the detected cells whose IOU is below 0.9 are regarded as false positive detections.

- *Precision, Recall, F-score and TEDS [24] for adjacent relationship estimation.*

  For logical structure correctness, we follow the evaluation protocol used in document images by calculating the precision, recall, and F-score for the cell adjacency [4] and the tree-edit-distance similarity (TEDS) [24].

Results of baseline models. We train the cell detectors[†] in the baseline and then parse the table structures with the above-mentioned post-processing schemes. Tab. 2 shows the evaluation results of all baseline models on the testing split of our WTW dataset. Compared with the anchor-based approaches Faster-RCNN, TridenNet, and Cascade-RCNN, the anchor-free CenterNet obtains the best F-score in the aspect of physical structure accuracy as it does not require fine adjustment of parameters. With more accurate cell detection results, it also achieves the best performance for logical structure correctness.

To further analyze the challenges of table structure parsing in the wild, we visualize the parsing results of two example images taken from different scenes for the baseline models with different object detectors in Fig. 2. As shown in this figure, the cells can be detected well for both the anchor-based and the anchor-free approaches when the table is approximately aligned with the image domain. By contrast, when leveraging those approaches in the image that have non-rigid deformation, the anchor-based approaches will yield incorrect results. The anchor-free detector, CenterNet, performs better than others while still remaining room for better table parsing accuracy. For more challenging images, it would be of great interest in developing a robust table structure parsing approach.

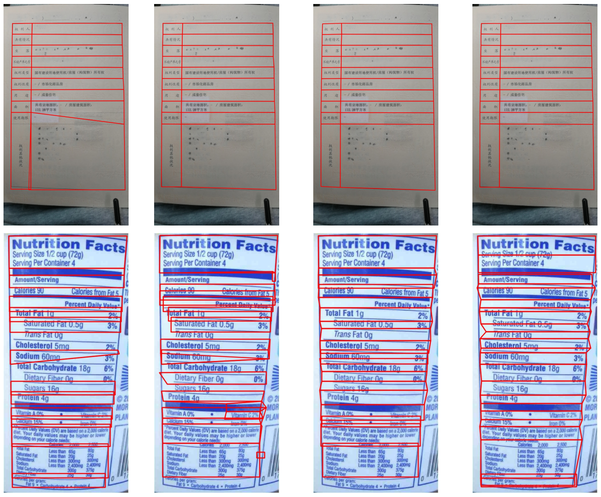

Faster-RCNN TridenNet Cascade-RCNN CenterNet

Figure 2. Visualization of the table structure parsing results of the baseline models on the WTW dataset.

Model	Physical Structure			Adjacency Relation			TEDS
	Prec.	Rec.	F1	Prec.	Rec.	F1	
Faster-RCNN	72.1	61.5	66.4	87.1	61.3	71.9	49.5
TridenNet	64.5	65.5	65.0	85.4	71.5	77.8	47.8
Cascade-RCNN	77.4	65.3	70.9	89.1	64.5	74.9	53.2
CenterNet	74.2	72.1	73.1	90.8	79.7	84.8	58.7

Table 2. Baseline models on WTW dataset. The physical structure is to measure the accuracy of cell coordinate when IOU=0.9, Adjacency Relation, and TEDS measure the row/col structure information, where the Adjacency Relation is based on the IOU=0.6

4. Cycle-CenterNet

Building on the top of CenterNet, our proposed network adds a Cycle-Pairing module and Pairing loss to learn the common vertex between neighbor cells on the basis of CenterNet [25]. Through the common vertex, we can splice all the cells together and get a complete table structure. Finally, using the same Parsing-Processing to get row/col information. An illustrative demonstration for our Cycle-CenterNet is shown in Fig. 3.

4.1. Cycle-Pairing Module

To recognize table structure, we proposed a Cycle-Pairing module to locate the cells and learn the splic-

[*] More detail and pseudo-codes for the heuristic grouping scheme and the post-processing module are described in the supplementary materials.

[†] The training detail is described in the supplemental material.

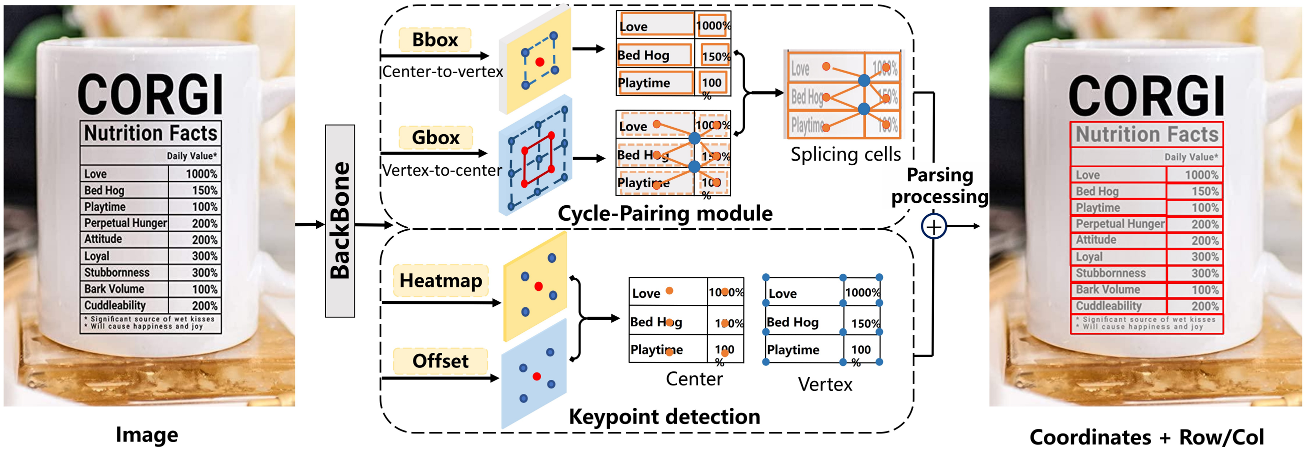


Figure 3. The pipeline of Cycle-CenterNet. Taking an image as input, our model produces one 2-channel keypoint heatmap and one 2-channel offset map. The Cycle-Pairing module outputs two 8-channel heatmaps, which learn the mutual-directed relationship between center point and vertices. According to the relationship, cells are grouped, and finally, the number of row and column information can be recovered by parsing processing.

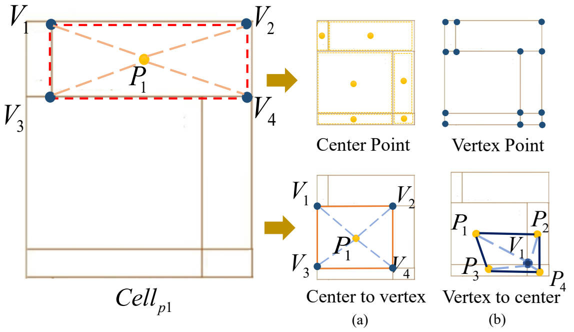


Figure 4. Illustration the process of grouping cells by mutual-directed relationship learned by cycle-pairing module.

ing information between cells, which consist of two branches, including *center-to-vertex branch* and *vertex-to-center branch*. As shown in Fig. 3, in the center-to-vertex branch, we regress the offset from a center of a table cell to its vertices, and following the post process of Centernet [25], the polygonal representation of table cells could be obtained; In the vertex-to-center branch, the offsets between common vertex and the centers of its surrounding cells centers are learned. Finally, the splicing information of tables could be deduced in the Parsing-Processing.

Center-to-Vertex branch for cells localization. Taking the feature map F from the DLA-34 [22] backbone as input, the center-to-vertex branch predict a $CV_{map} \in R^{\frac{h}{4}\times\frac{w}{4}\times 8}$. As shown in Fig. 4 (a), the CV_{map} indicates the coordinate offset $\{\Delta x, \Delta y\}$ between the center point $P = \{x_C, y_C\}$ and its four vertices $V = \{x_V, y_V\}$, denoted by

$$\begin{cases}\Delta x_{C_{ik}} = x_{C_i} - x_{V_{ik}} \\ \Delta y_{C_{ik}} = y_{C_i} - y_{V_{ik}}\end{cases} , i = 1 : N^C, k = 1 : 4 \, , \qquad (1)$$

where N^C is the number of all the center points of table cells.

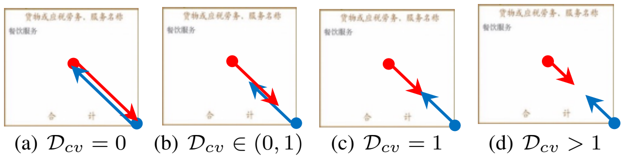
Vertex-to-Center branch for cells grouping. Taking the feature map F from backbone DLA-34 as input, the Vertex-to-Center branch predicts a $VC_{map} \in R^{\frac{h}{4}\times\frac{w}{4}\times 8}$. As shown in Fig. 4 (b), the VC_{map} encodes the coordinate offset $\{\Delta x, \Delta y\}$ between the common vertex $V = \{x_V, y_V\}$ and four center points $P = \{x_C, y_C\}$ of the surrounding table cells, denoted by

$$\begin{cases}\Delta x_{V_{ik}} = x_{V_i} - x_{C_{ik}} \\ \Delta y_{V_{ik}} = y_{V_i} - y_{C_{ik}}\end{cases} |i = 1 : N^V, k = 1 : 4 \, , \qquad (2)$$

where N^V denotes the number of all the common vertexes. If the number of cells sharing this vertex K is smaller than 4, the regression value of the remaining positions are set to 0.

4.2. Pairing Loss for Cycle-Pairing Module

Instead of directly applying loss functions on the output maps of Cycle-Pairing Module, we design a pairing loss to supervise the network learning better offsets for both center-to-vertex and vertex-to-center branches by pair-wise compute the loss function for each pair of the center and vertex that belongs to the same cell in the expected table. Denoted



(a) $\mathcal{D}_{cv} = 0$ (b) $\mathcal{D}_{cv} \in (0, 1)$ (c) $\mathcal{D}_{cv} = 1$ (d) $\mathcal{D}_{cv} > 1$

Figure 5. Illustration of the traditional cases for center-vertex pairs during training.

by $\mathcal{P}_{cv} = (\Delta x_{cv}, \Delta y_{cv}, \Delta x_{vc}, \Delta y_{vc})$ for the predicted offset of a pair of center c and v, we compute the loss function L_p by

$$L_p = \sum_{c,v} \omega(\mathcal{P}_{cv}) \left(\lambda_{cv} L_{cv} + \lambda_{vc} L_{vc}\right), \tag{3}$$

where L_{cv} and L_{vc} are the ℓ_1 loss between the predicted offsets and the corresponding groundtruth, $\lambda_{cv} = 1.0$ and $\lambda_{vc} = 0.5$ are the hyperparameters to tune the importance between those loss items, $\omega(\mathcal{P}_{cv})$ dynamically weighing the overall loss according to the regression quality.

Dynamic weighing function $\omega(\mathcal{P}_{cv})$. The cycle-pairing module represents the pairwise pointing relationship between vertex and center point. In fact, it is not necessary to regress cell bounding box and common vertex group box such accurately, as long as the prediction of center and vertex from one center-vertex pair are intersected. So we use $\omega(\mathcal{P}_{cv})$ to weight losses l_{cv} and l_{vc} for center-vertex pairs:

$$\omega(\mathcal{P}_{cv}) = 1 - \exp\left(-\pi \mathcal{D}_{cv}\right), \tag{4}$$

where $\mathcal{D}_{cv}$ is the pair distance defined as

$$\mathcal{D}_{cv} = \min\left(\frac{|x_{cv_i} - x_{cv_i}{}^*| + |x_{vc_i} - x^*_{vc_i}|}{|x_{cv_i}{}^*|}, 1\right) \tag{5}$$

where x_{cv} is a regression value from center to vertex, while x_{vc} is vertex to the center. Therefore, $\mathcal{D}_{cv}$ defines the regression error score for each center-vertex pair. As shown in Fig. 5, if $\mathcal{D}_{cv} = 0$, means that the vertex and the center point to each other strictly without any errors. If $0 < \mathcal{D}_{cv} < 1$, means that although the vertex and center couldn't strictly point to each other, but pointing into each other's bounding box. If $\mathcal{D}_{cv} \geq 1$, means that there is no intersection between the center and vertex, which is the main sample that needs to focus on learning.

The loss L_k of the keypoint branch and the loss L_{off} of the offset branch are consistent with CenterNet [25]. The overall training loss is:

$$L_{det} = L_k + \lambda_{off} L_{off} + L_p \tag{6}$$

4.3. Parsing-Processing Module

As the last step, we propose a Parsing-Processing module to recover the complete table structure information, including table id, start row/column, and end row/column. First, split every cell into 4 bounding edges, then merge the up edges and down edges to horizontal lines and merge left edges and right edges to vertical lines according to cell connectivity. Next, sort the horizontal lines, vertical lines and index them from 0. Finally, rank cells by line index and outputs row/column information. The pseudo code is given in the supplementary materials.

4.4. Training Detail

In the training process of the Cycle-Centernet, we use the pre-trained weight on COCO, and resize the max side of the training image to 1024 with scaling the short side equally. The initial learning rate is set to 1.25×10^{-3}, and decayed to 1.25×10^{-4} and 1.25×10^{-5} in the 90th and 120th epoch respectively. The model is trained with a total of 150 epochs. All the experiments are performed on a workstation with 8 NVIDIA GTX 1080Ti GPUs. During the training, we set the batch size to 32 per GPU in parallel.

5. Experiments

We perform extensive experiments on the proposed WTW dataset to verify the effectiveness of Cycle-CenterNet. Although we mainly focus on wired tables in wild scenes, we additionally do experiments on the widely used benchmarks of ICDAR2013 and ICDAR2019 to demonstrate that: 1) the WTW dataset covers a wide range of tabular images for practical applications and 2) our proposed Cycle-CenterNet is able to recognize wireless tables.

5.1. Evaluation on WTW

To evaluate the performance of Cycle-Centernet on the WTW dataset, we compare it with some latest table recognition methods including Split+Heuristic [19] and CascadeTabNet [12]. For a fair comparison, those methods are retrained on the WTW dataset with the author-provided hyperparameter settings. As discussed in Sec. 3.2, we evaluate the correctness of the parsed tables in both aspects of physical structures and logical structures. For the physical structure (cell coordinates), the precision, recall and, F1 score are used under a strict IoU threshold of 0.9. For the logical structure (row/column information), TEDS [24] and cell adjacent relationship (IOU=0.6) [4] are used as evaluation metrics.

Quantitative comparison. Tab. 3 shows the performance of all the models on the challenging WTW dataset. It is shown that the state-of-the-art table structure parsing approaches CascadeTabNet [12] and Split+Heuristic [19] for well-conditioned tabular images obtain poor performance on the challenging WTW dataset that has many wild images. By contrast, benefiting from the flexible design of our

Category	Model	Backbone	Physical coordinates			Adjacency Relation			TEDS
			Prec.	Rec.	F1	Prec.	Rec.	F1	
Base Model	CenterNet	DLA-34	74.2	72.1	73.1	90.8	79.7	84.8	58.7
Table structure models	Split+Heuristic	-	3.2	3.6	3.4	25.7	29.9	27.6	26.0
	CascadeTabNet	CascadeNet	-	-	-	16.4	3.6	5.9	11.4
Ours	CenterNet(Polygon box)	DLA-34	75.1	75.7	75.4	93.0	89.2	91.1	70.1
	Cycle-CenterNet	DLA-34	**78.2**	78.2	78.2	93.2	91.4	92.2	74.3
	Cycle-Centernet+PairLoss	DLA-34	78.0	**78.5**	**78.3**	**93.3**	**91.5**	**92.4**	**83.3**

Table 3. Results on WTW dataset. The evaluation metrics are same with Tab. 2

Method	Simple		Inclined		Curved		Occluded and blurred		Extreme aspect ratio		Overlaid		Muti color and grid	
	F1	TEDS	F1	TEDS	F1	TEDS	F1	TEDS	F1	TEDS	F1	TEDS	F1	TEDS
Ours	99.3	94.2	97.7	90.6	76.1	70	77.4	53.3	91.9	77.4	84.1	51.2	93.7	66.7

Table 4. Results on different categories.

Model	Training Datasets	IOU=0.6		
		Prec.	Rec.	F1
DeepDeSRT [16]	SciTSR	63.1	61.9	62.5
Split+Heuristic [19]	Private	93.8	92.2	93.0
TableNet [11]	Marmot Extended	92.2	89.9	91.0
Tabstruct-Net [14]	SciTSR	91.5	89.7	90.6
Ours	WTW+ICDAR19	95.5	88.3	91.7
Ours*	WTW	97.5	98.4	98.0

Table 5. Comparison for cell adjacency relation on ICDAR-2013 dataset. Here ”*” denotes for the result on wired tables only.

Model	Training Datasets	IOU				WAvg.
		0.6	0.7	0.8	0.9	
NLPR-PAL	-	36.5	30.5	19.5	3.5	20.6
CascadeTab [12]	Marmot etc.	43.8	35.4	19	3.6	23.2
GTE [23]	FinTabNet	38.5	-	-	-	24.8
TabStruct-Net [14]	SciTSR	80.4	-	-	-	-
Ours	WTW	**80.8**	**51.1**	**31.9**	**11.2**	**40.0**

Table 6. Comparison with participants of ICDAR 19 Track B2 (Modern) F1-scores [3], here all the methods finetune on the ICDAR-2019 dataset, we just list their initial training datasets.

method, we significantly improve the performance of table structure parsing by large margins. Our proposed Cycle-CenterNet obtains the best performance by using the proposed pairing loss function.

Ablation study. In Sec. 3.2, we argue that the accuracy of cell regression has a great influence on table structure recognition, so we replace horizontal rectangle regression with arbitrary quadrilateral regression in CenterNet. Although the cell detection is only increased by 2.3%, we get 6.3% and 11.4% improvement on adjacency relation and TEDS. The cycle-pairing module can simultaneously detect and group tabular cells into structured digital tables, which makes the Cycle-CenterNet get 2.8% improvement on cell detection, 1.1% improvement on adjacency relation, and 4.2% improvement on TEDS. With the collaborative optimization of Pairing loss on center-vertice pair, the addition of Pairing loss makes Cycle-CenterNet significantly increase on the dedicated table structure evaluation matrix of TEDS by 9%.

Compared with our base model CenterNet, the Cycle-CenterNet has a 7.6% improvement on adjacency relation and 24.6% improvement on TEDS. Since the models in other papers are not designed for wild tables, our Cycle-CenterNet shows obvious advantages in all evaluation metrics.

Sub-category experiments. To verify the complexity of our WTW dataset, we analyze the model results on different types of tables separately. Results are shown in Tab. 4. Cycle-CenterNet achieves 99.3% in adjacency relation and 94.2% in TEDS on the simple subset of WTW, 97.7% in adjacency relation and, 90.6% in TEDS on the inclined subset. It means the Cycle-CenterNet can get a good result for ordinary tables. Relatively good results (91.9% at adjacency relation and 77.4% at TEDS) are obtained for tables with extreme aspect ratios for the vertex-to-center branch which can pull the bounding box to the vertex. Although our Cycle-Centernet has reached the state of the art, it still needs to be improved on some styles like curved, overlaid, etc. Our model just gets 53.3% at TEDS on occlusion and blur subset. Besides, multiple table superposition also brings great difficulties to the table recognition task. A table with especially dense cells, combined with any slightly complex feature makes the task drastically difficult. We will continue to seek solutions to these problems and look forward to more researchers joining in.

5.2. Evaluation on other Datasets

To evaluate the robustness and versatility of Cycle-CenterNet, we test our model on two mostly used datasets in table structure recognition, ICDAR-2013 [5] and ICDAR-2019 [3]. These two datasets consisting of both wired and wireless tables. For lacking wireless table in WTW, we finetune the Cycle-CenterNet on the ICDAR2019 training

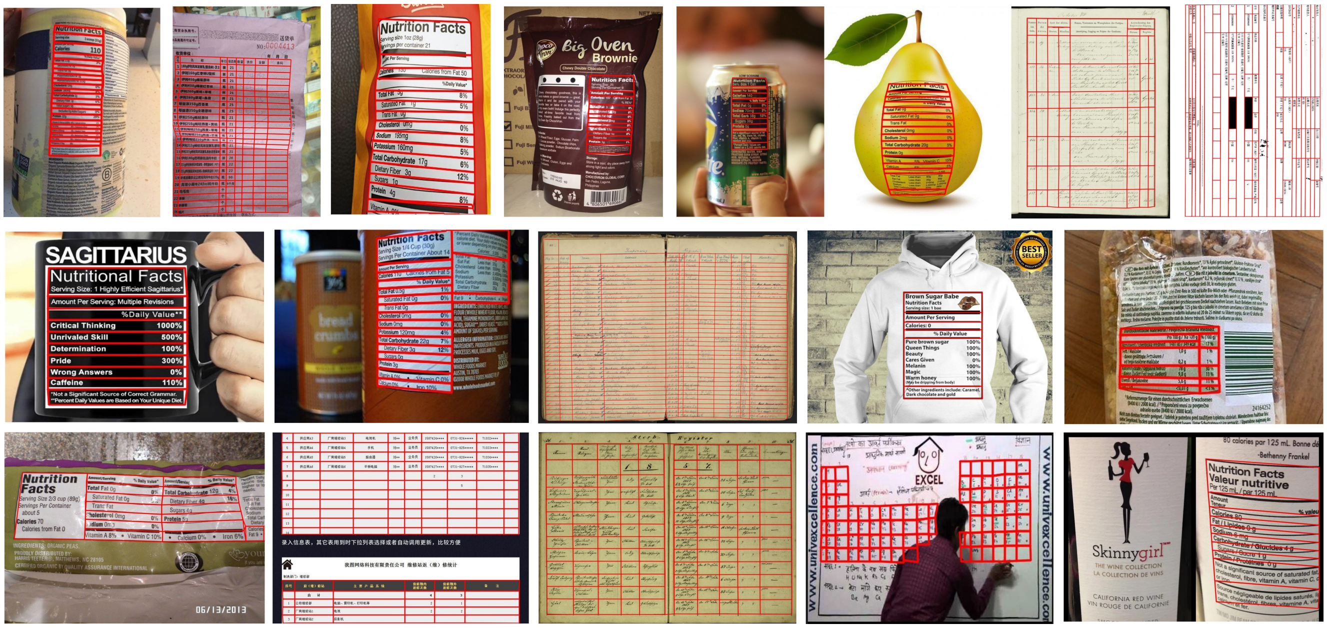


Figure 6. Qualitative results of Cycle-Centernet on different datasets

set and test on ICDAR 2019 Track B2 dataset. Since ICDAR2013 only has test set, we selected some wireless tables from ICDAR-2019 to finetune our model and test on ICDAR2013. Tab. 5 and 6 show the results on datasets ICDAR-2013 [5] and ICDAR-2019 [3]. Here the evaluation metric only includes the cell adjacency relation [4]: Recall (Rec.), Precision (Prec.), F1-metric op (F1).

Although Cycle-CenterNet is mainly designed for wired tables and trained on WTW, it still obtains 91.7% fscore on ICDAR2013 with wireless tables, and is only 1.3% lower than the first-ranked model of Split+Heuristic. This shows our model can also cover wireless tables. Similarly, our WTW mainly focuses on wild wired tables, but Cycle-CenterNet trained only on WTW and tested on ICDAR2013 wired tables can still achieve 98.0% fscore, indicating that WTW can also cover simple wired tables.

For ICDAR-2019 [3], we keep the same evaluation metric: ICDAR 19 Track B2 F1-scores [3], which is based on the adjacency relation evaluation [4]. Precision, Recall, and F1 scores are calculated with IoU thresholds 0.6, 0.7, 0.8, and 0.9 respectively. The Weighted-Average F1 (WAvg.) is calculated by assigning a weight to each F1 value of the corresponding IoU threshold. As shown in Tab. 6, compared with the highest result of the Weighted-Average F1 reported so far, Cycle-CenterNet has improved 15.2% and achieve the start of the art.

6. Conclusions and Future Work

In this paper, we tackle the problem of table structure parsing in the wild by proposing a new WTW dataset and a deep table structure parser, Cycle-CenterNet. On one hand, the proposed WTW dataset contains about 14k real-scene images that are taken in wild imaging conditions, which pushes the boundary of table structure parsing from the digital document images to the real-scene images. On another hand, we propose a new approach for wild-scene table structure recognition, called Cycle-CenterNet, which addressed the major weaknesses of the existing approaches including imprecise geometry prediction of instances with extremely physical distortion and defectiveness in extracting logical structures of misaligned tables. The comprehensive experiments demonstrated that the proposed approach resolves the mentioned issues in a principled way and achieves a new state-of-art for table structure parsing. We hope our proposed WTW dataset can further improve future research on table recognition.

Acknowledgement

This work was supported by the National Natural Science Foundation of China under Grant 61922065, Grant 61771350, Grant 41820104006, and National Post-Doctoral Program for Innovative Talents under Grant BX20200248. This work was also supported by Alibaba Group through Alibaba Innovative Research (AIR) program. The numerical calculations in this paper have been done on the Supercomputing System in the Supercomputing Center of Wuhan University. The views presented in this paper are those of the authors and should not be interpreted as representing any funding agencies.

References

[1] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6154–6162, 2018. 4

[2] Zewen Chi, Heyan Huang, Heng-Da Xu, Houjin Yu, Wanxuan Yin, and Xian-Ling Mao. Complicated table structure recognition. *arXiv preprint arXiv:1908.04729*, 2019. 2, 3

[3] Liangcai Gao, Yilun Huang, Hervé Déjean, Jean-Luc Meunier, Qinqin Yan, Yu Fang, Florian Kleber, and Eva Lang. Icdar 2019 competition on table detection and recognition (ctdar). In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1510–1515. IEEE, 2019. 2, 3, 7, 8

[4] Max Göbel, Tamir Hassan, Ermelinda Oro, and Giorgio Orsi. A methodology for evaluating algorithms for table understanding in pdf documents. In *Proceedings of the 2012 ACM symposium on Document engineering*, pages 45–48, 2012. 4, 6, 8

[5] Max Göbel, Tamir Hassan, Ermelinda Oro, and Giorgio Orsi. Icdar 2013 table competition. In *2013 12th International Conference on Document Analysis and Recognition*, pages 1449–1453. IEEE, 2013. 2, 3, 7, 8

[6] E Green and M Krishnamoorthy. Recognition of tables using table grammars. In *Proceedings of the Fourth Annual Symposium on Document Analysis and Information Retrieval*, pages 261–278, 1995. 1

[7] Katsuhiko Itonori. Table structure recognition based on text block arrangement and ruled line position. In *Proceedings of 2nd International Conference on Document Analysis and Recognition (ICDAR'93)*, pages 765–768. IEEE, 1993. 1

[8] Thomas G Kieninger. Table structure recognition based on robust block segmentation. In *Document Recognition V*, volume 3305, pages 22–32. International Society for Optics and Photonics, 1998. 1, 3

[9] Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, Ming Zhou, and Zhoujun Li. Tablebank: A benchmark dataset for table detection and recognition. *arXiv e-prints*, pages arXiv–1903, 2019. 2, 3

[10] Yanghao Li, Yuntao Chen, Naiyan Wang, and Zhaoxiang Zhang. Scale-aware trident networks for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6054–6063, 2019. 4

[11] Shubham Singh Paliwal, D Vishwanath, Rohit Rahul, Monika Sharma, and Lovekesh Vig. Tablenet: Deep learning model for end-to-end table detection and tabular data extraction from scanned document images. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 128–133. IEEE, 2019. 2, 3, 7

[12] 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. 3, 6, 7

[13] Shah Rukh Qasim, Hassan Mahmood, and Faisal Shafait. Rethinking table recognition using graph neural networks. pages 142–147, 2019. 2, 3

[14] Sachin Raja, Ajoy Mondal, and CV Jawahar. Table structure recognition using top-down and bottom-up cues. In *European Conference on Computer Vision*, pages 70–86. Springer, 2020. 2, 3, 7

[15] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(6):1137–1149, 2017. 4

[16] 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)*, volume 1, pages 1162–1167. IEEE, 2017. 2, 3, 7

[17] Asif Shahab, Faisal Shafait, Thomas Kieninger, and Andreas Dengel. An open approach towards the benchmarking of table structure recognition systems. In *Proceedings of the 9th IAPR International Workshop on Document Analysis Systems*, pages 113–120, 2010. 2, 3

[18] Ashwin Tengli, Yiming Yang, and Nian Li Ma. Learning table extraction from examples. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 987–993, 2004. 3

[19] 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. 1, 6, 7

[20] Scott Tupaj, Zhongwen Shi, C Hwa Chang, and Hassan Alam. Extracting tabular information from text files. *EECS Department, Tufts University, Medford, USA*, 1996. 1

[21] Yalin Wang, Ihsin T Phillips, and Robert M Haralick. Table structure understanding and its performance evaluation. *Pattern recognition*, 37(7):1479–1497, 2004. 3

[22] Fisher Yu, Dequan Wang, Evan Shelhamer, and Trevor Darrell. Deep layer aggregation. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 2403–2412, 2018. 5

[23] 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. 2, 3, 7

[24] Xu Zhong, Elaheh ShafieiBavani, and Antonio Jimeno Yepes. Image-based table recognition: data, model, and evaluation. *arXiv preprint arXiv:1911.10683*, 2019. 2, 3, 4, 6

[25] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019. 4, 5, 6