

## DeepSeek-OCR 2: Visual Causal Flow

Haoran Wei, Yaofeng Sun, Yukun Li

DeepSeek-AI

### Abstract

We present DeepSeek-OCR 2 to investigate the feasibility of a novel encoder—DeepEncoder V2—capable of dynamically reordering visual tokens upon image semantics. Conventional vision-language models (VLMs) invariably process visual tokens in a rigid raster-scan order (top-left to bottom-right) with fixed positional encoding when fed into LLMs. However, this contradicts human visual perception, which follows flexible yet semantically coherent scanning patterns driven by inherent logical structures. Particularly for images with complex layouts, human vision exhibits causally-informed sequential processing. Inspired by this cognitive mechanism, DeepEncoder V2 is designed to endow the encoder with causal reasoning capabilities, enabling it to intelligently reorder visual tokens prior to LLM-based content interpretation. This work explores a novel paradigm: whether 2D image understanding can be effectively achieved through two-cascaded 1D causal reasoning structures, thereby offering a new architectural approach with the potential to achieve genuine 2D reasoning. Codes and model weights are publicly accessible at <http://github.com/deepseek-ai/DeepSeek-OCR-2>.

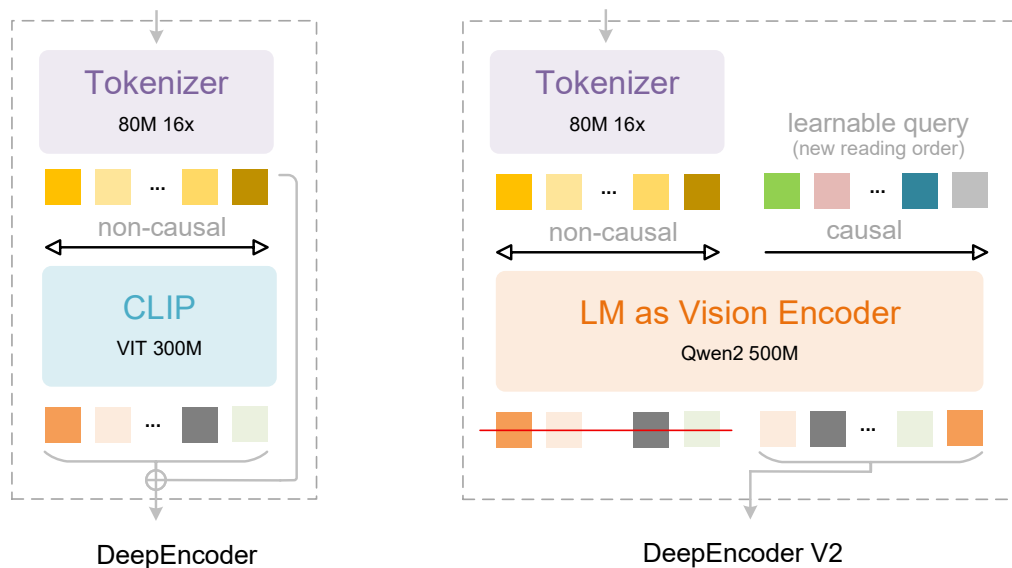


Figure 1 | We substitute the CLIP component in DeepEncoder with an LLM-style architecture. By customizing the attention mask, visual tokens utilize bidirectional attention while learnable queries adopt causal attention. Each query token can thus attend to all visual tokens and preceding queries, allowing progressive causal reordering over visual information.

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Related Works</b>	<b>4</b>
2.1	Parallelized Queries in Decoder . . . . .	4
2.2	Parallelized Queries in Projector . . . . .	4
2.3	LLM-based Multimodal Initialization . . . . .	4
<b>3</b>	<b>Methodology</b>	<b>5</b>
3.1	Architecture . . . . .	5
3.2	DeepEncoder V2 . . . . .	5
3.2.1	Vision tokenizer . . . . .	5
3.2.2	Language model as vision encoder . . . . .	6
3.2.3	Causal flow query . . . . .	6
3.2.4	Attention mask . . . . .	7
3.3	DeepSeek-MoE Decoder . . . . .	7
<b>4</b>	<b>Experimental Settings</b>	<b>8</b>
4.1	Data Engine . . . . .	8
4.2	Training Pipelines . . . . .	8
4.2.1	Training DeepEncoder V2 . . . . .	8
4.2.2	Query enhancement . . . . .	8
4.2.3	Continue-training LLM . . . . .	8
<b>5</b>	<b>Evaluation</b>	<b>9</b>
5.1	Main Results . . . . .	9
5.2	Improvement Headroom . . . . .	10
5.3	Practical Readiness . . . . .	10
<b>6</b>	<b>Discussion and Future Works</b>	<b>11</b>
6.1	Towards Genuine 2D Reasoning . . . . .	11
6.2	Towards Native Multimodality . . . . .	11
<b>7</b>	<b>Conclusion</b>	<b>11</b>

## 1. Introduction

The human visual system closely mirrors transformer-based vision encoders [14, 16]: foveal fixations function as visual tokens, locally sharp yet globally aware. However, unlike existing encoders that rigidly scan tokens from top-left to bottom-right, human vision follows a causally-driven flow guided by semantic understanding. Consider tracing a spiral—our eye movements follow inherent logic where each subsequent fixation causally depends on previous ones. By analogy, visual tokens in models should be selectively processed with ordering highly contingent on visual semantics rather than spatial coordinates.

This insight motivates us to fundamentally reconsider the architectural design of vision-language models (VLMs), particularly the encoder component. LLMs are inherently trained on 1D sequential data, while images are 2D structures. Directly flattening image patches in a predefined raster-scan order introduces unwarranted inductive bias that ignores semantic relationships. To address this, we present DeepSeek-OCR 2 with a novel encoder design—DeepEncoder V2—to advance toward more human-like visual encoding. Following DeepSeek-OCR [54], we adopt document reading as our primary experimental testbed. Documents present rich challenges including complex layout orders, intricate formulas, and tables. These structured elements inherently carry causal visual logic, demanding sophisticated reasoning capabilities that make document OCR an ideal platform for validating our approach.

Our main contributions are threefold:

First, we present DeepEncoder V2, featuring several key innovations: (1) we replace the CLIP [37] component in DeepEncoder [54] with a compact LLM [48] architecture, as illustrated in Figure 1, to achieve visual causal flow; (2) to enable parallelized processing, we introduce learnable queries [10], termed causal flow tokens, with visual tokens prepended as a prefix—through a customized attention mask, visual tokens maintain global receptive fields, while causal flow tokens can obtain visual token reordering ability; (3) we maintain equal cardinality between causal and visual tokens (with redundancy such as padding and borders) to provide sufficient capacity for re-fixation; (4) only the causal flow tokens—the latter half of the encoder outputs—are fed to the LLM [24] decoder, enabling cascade causal-aware visual understanding.

Second, leveraging DeepEncoder V2, we present DeepSeek-OCR 2, which preserves the image compression ratio and decoding efficiency of DeepSeek-OCR while achieving substantial performance improvements. We constrain visual tokens fed to the LLM between 256 and 1120. The lower bound (256) corresponds to DeepSeek-OCR’s tokenization of 1024×1024 images, while the upper bound (1120) matches Gemini-3 pro’s [44] maximum visual token budget. This design positions DeepSeek-OCR 2 as both a novel VLM architecture for research exploration and a practical tool for generating high-quality training data for LLM pretraining.

Finally, we provide preliminary validation for employing language model architectures as VLM encoders—a promising pathway toward unified omni-modal encoding. This framework enables feature extraction and token compression across diverse modalities (images, audio, text [28]) by simply configuring modality-specific learnable queries. Crucially, it naturally succeeds to advanced infrastructure optimizations from the LLM community, including Mixture-of-Experts (MoE) architectures, efficient attention mechanisms [26], and so on.

In summary, we propose DeepEncoder V2 for DeepSeek-OCR 2, employing specialized attention mechanisms to effectively model the causal visual flow of document reading. Compared to the DeepSeek-OCR baseline, DeepSeek-OCR 2 achieves 3.73% performance gains on OmniDocBench v1.5 [34] and yields considerable advances in visual reading logic.

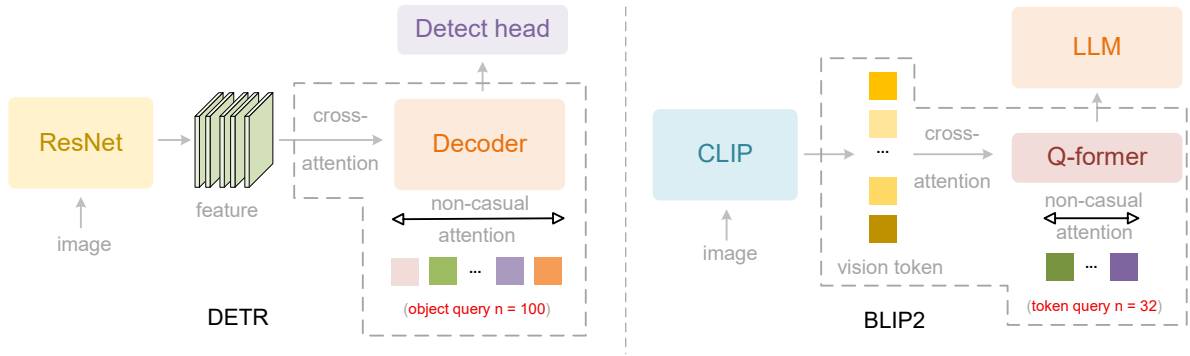


Figure 2 | This figure shows two computer vision models with parallelized queries: DETR’s decoder [10] for object detection and BLIP2’s Q-former [22] for visual token compression. Both employ bidirectional self-attention among queries.

## 2. Related Works

### 2.1. Parallelized Queries in Decoder

DETR [10] pioneered the integration of transformer architecture into object detection, fundamentally breaking away from traditional detection paradigms [38, 40]. To overcome the efficiency limitations of serial decoding in transformer blocks, DETR introduced preset parallelized learnable queries—a set of 100 object queries that encode object priors such as shape and position through training. These queries interact with feature maps [18] via cross-attention mechanisms, while simultaneously engaging in bidirectional information exchange among themselves through self-attention. DETR established a foundational paradigm that enables transformers to handle parallelized tokens. The object query design has since become the de facto standard architectural component in subsequent transformer-based detection methods [29, 56].

### 2.2. Parallelized Queries in Projector

In recent years, vision-language models [7, 9, 22, 50] have developed rapidly, with architectures converging toward the encoder-projector-LLM paradigm. The projector aligns visual tokens with the LLM’s embedding space, serving as a critical bridge that enables LLMs to understand visual content. Q-former, introduced in BLIP-2 [22], exemplifies an effective projector design that employs learnable queries for visual token compression. Adopting a BERT-like [15] architecture and drawing inspiration from DETR’s object queries [10], Q-former utilizes 32 learnable queries that interact with hundreds of CLIP [37] visual tokens through cross-attention. These compressed query representations are subsequently fed into the LLM, achieving effective mapping from visual to language space. The success of Q-former demonstrates that parallelized learnable queries are effective not only for feature decoding in detection tasks but also for token compression in multimodal alignment.

### 2.3. LLM-based Multimodal Initialization

LLMs trained on large-scale internet data have proven effective as initialization for multimodal models. Pang et al. [35] demonstrated that frozen LLM transformer layers enhance visual discriminative tasks. Moreover, encoder-free or lightweight-encoder models such as Fuyu [5] and Chameleon [43] in vision, as well as VALL-E [47] in speech, further validate the potential of LLM pretrained weights for multimodal initialization.

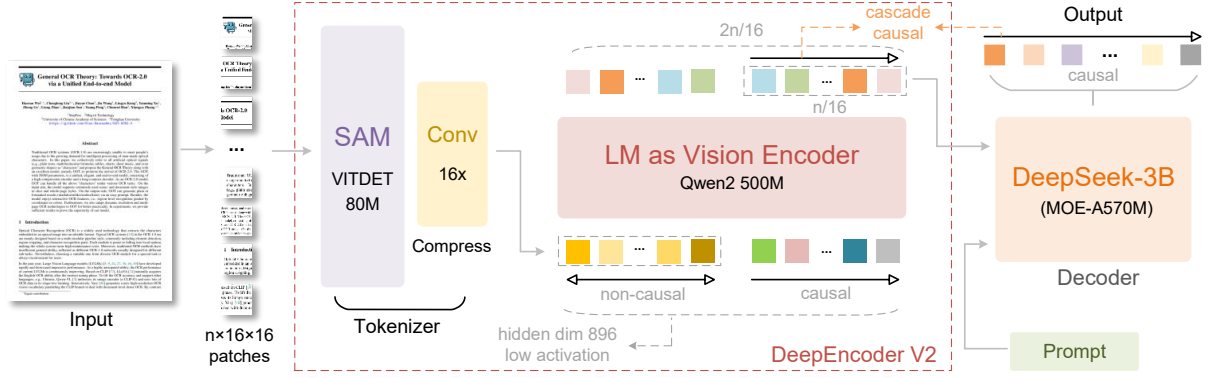


Figure 3 | DeepSeek-OCR 2 adopts the visual token compression mechanism from DeepEncoder, employing an 80M-parameter image compressor that reduces visual tokens by a factor of 16. DeepEncoder V2 differs by replacing DeepEncoder’s CLIP module with a compact language model architecture. Through customized attention masks, this LM-style vision encoder acquires CLIP’s knowledge compression capabilities while initiating causal modeling of visual sequences.

### 3. Methodology

#### 3.1. Architecture

As shown in Figure 3, DeepSeek-OCR 2 inherits the overall architecture of DeepSeek-OCR, which consists of an encoder and a decoder. The encoder discretizes images into visual tokens, while the decoder generates outputs conditioned on these visual tokens and text prompts. The key distinction lies in the encoder: we upgrade DeepEncoder to DeepEncoder V2, which retains all capabilities of its predecessor while introducing causal reasoning through a novel architectural design. We elaborate on the details of DeepSeek-OCR 2 in the following sections.

#### 3.2. DeepEncoder V2

The vanilla encoder serves as an important component that extracts and compresses image features through attention mechanisms, where each token attends to all others, achieving full-image receptive fields analogous to human foveal and peripheral vision. However, flattening 2D image patches into a 1D sequence imposes a rigid ordering bias through text-oriented positional encodings (e.g., RoPE [42]). This contradicts natural visual reading patterns, especially non-linear layouts in optical texts, forms and tables.

##### 3.2.1. Vision tokenizer

The first component of DeepEncoder V2 is a vision tokenizer. Following DeepEncoder, we employ an architecture combining an 80M-parameter SAM-base [21] along with two convolutional layers [50]. The output dimension of the final convolutional layer is reduced from 1024 in DeepEncoder to 896 to align with the subsequent pipeline. Note that this compression-based tokenizer is not mandatory and can be replaced with simple patch embedding. We retain it because it achieves 16 $\times$  token compression [19, 46, 51, 52] through window attention with minimal parameters, significantly reducing both computational cost and activation memory for the subsequent global attention module. Moreover, its parameter count ( 80M) remains comparable to the typical 100M parameters used for text input embeddings in LLMs.

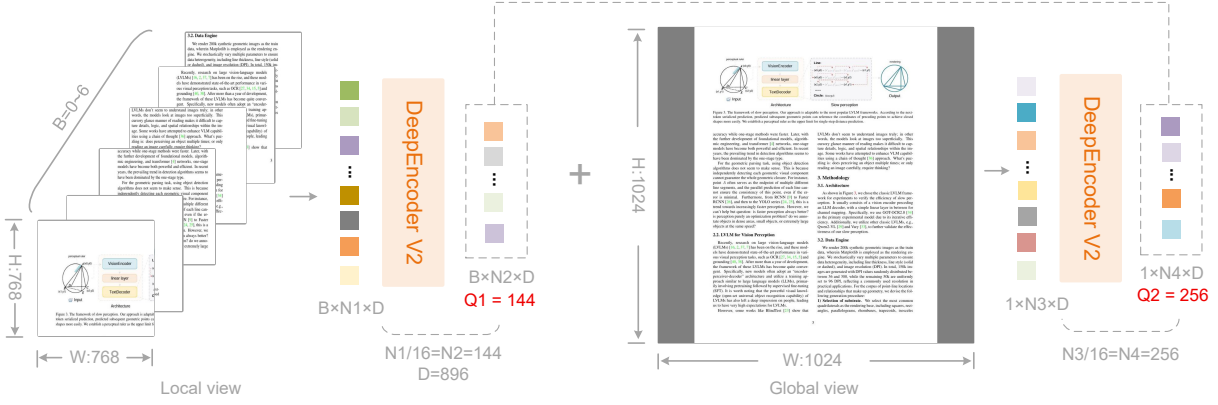


Figure 4 | Token count calculation in DeepEncoder V2. DeepEncoder V2 outputs 256–1120 tokens per image using a multi-crop strategy with 0–6 local views. With 0 local views, only the global view produces 256 tokens; with 6 local views, the count reaches 1120 ( $6 \times 144 + 256$ ).

### 3.2.2. Language model as vision encoder

In DeepEncoder, a CLIP ViT follows the vision tokenizer to compress visual knowledge. DeepEncoder V2 redesigns this component into an LLM-style architecture with a dual-stream attention mechanism. Visual tokens utilize bidirectional attention to preserve CLIP’s global modeling capability, while newly introduced causal flow queries employ causal attention. These learnable queries are appended after visual tokens as a suffix, where each query attends to all visual tokens and preceding queries. By maintaining equal cardinality between queries and visual tokens, this design imposes semantic ordering and distilling on visual features without altering token count. Finally, only the causal query outputs are fed to the LLM decoder.

We instantiate this architecture using Qwen2-0.5B [48], whose 500M parameters are comparable to CLIP ViT (300M) without introducing excessive computational overhead. The decoder-only architecture with prefix-concatenation of visual tokens proves crucial: extra experiments with cross-attention in an mBART-style [30] encoder-decoder structure fail to converge. We hypothesize this failure stems from insufficient visual token interaction when isolated in a separate encoder. In contrast, the prefix design keeps visual tokens active throughout all layers, fostering effective visual information exchange with causal queries.

This architecture actually establishes two-stage cascade causal reasoning: the encoder semantically reorders visual tokens through learnable queries, while the LLM decoder performs autoregressive reasoning over the ordered sequence. Unlike vanilla encoders that impose rigid spatial ordering through positional encodings, our causally-ordered queries adapt to smooth visual semantics while naturally aligning with the LLM’s unidirectional attention pattern. This design may bridge the gap between 2D spatial structure and 1D causal language modeling.

### 3.2.3. Causal flow query

As aforementioned, the number of causal query tokens equals the number of visual tokens, computed as  $\frac{W \times H}{16^2 \times 16}$ , where  $W$  and  $H$  denote the width and height of the image input to the encoder. To avoid maintaining multiple query sets for different resolutions, we adopt a multi-crop strategy with fixed query configurations at predefined resolutions.

Specifically, the global view uses a resolution of  $1024 \times 1024$ , corresponding to 256 query embeddings denoted as  $\text{query}_{\text{global}}$ . Local crops adopt a resolution of  $768 \times 768$ , with the

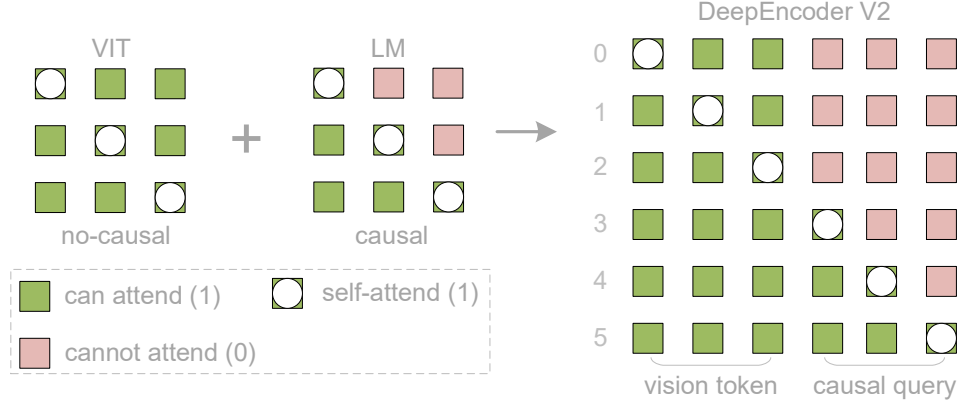


Figure 5 | Attention mask architecture of DeepEncoder V2. Concatenation of bidirectional mask (vision tokens, ViT-like) and causal triangular mask (flow tokens, LLM decoder-style).

number of crops  $k$  ranging from 0 to 6 (no cropping is applied when both image dimensions are smaller than 768). All local views share a unified set of 144 query embeddings, denoted as  $\text{query}_{\text{local}}$ . Therefore, the total number of reordered visual tokens fed to the LLM is  $k \times 144 + 256$ , ranging from [256, 1120]. This maximum token count (1120) is lower than DeepSeek-OCR’s 1156 (Gundam mode) and matches Gemini-3-Pro’s maximum visual token budget.

### 3.2.4. Attention mask

To better illustrate the attention mechanism of DeepEncoder V2, we visualize the attention mask in Figure 5. The attention mask is composed of two distinct regions. The left region applies bidirectional attention (similar to ViT) to original visual tokens, allowing full token-to-token visibility. The right region employs causal attention (triangular mask, identical to decoder-only LLMs) for causal flow tokens, where each token attends only to previous tokens. These two components are concatenated along the sequence dimension to construct DeepEncoder V2’s attention mask ( $\mathbf{M}$ ), as follows:

$$\mathbf{M} = \begin{bmatrix} \mathbf{1}_{m \times m} & \mathbf{0}_{m \times n} \\ \mathbf{1}_{n \times m} & \text{LowerTri}(n) \end{bmatrix}, \quad \text{where } n = m \quad (1)$$

where  $n$  is the number of causal query tokens,  $m$  represents vanilla visual tokens number, and LowerTri denotes a lower triangular matrix (with ones on and below the diagonal, zeros above).

### 3.3. DeepSeek-MoE Decoder

Since DeepSeek-OCR 2 primarily focuses on encoder improvements, we do not upgrade the decoder component. Following this design principle, we retain DeepSeek-OCR’s decoder – a 3B-parameter MoE structure with about 500M active parameters. The core forward pass of DeepSeek-OCR 2 can be formulated as:

$$\mathbf{O} = \mathcal{D}(\pi_Q(\mathcal{T}^L(\mathcal{E}(\mathbf{I}) \oplus \mathbf{Q}_0; \mathbf{M}))) \quad (2)$$

where  $\mathbf{I} \in \mathbb{R}^{H \times W \times 3}$  is the input image,  $\mathcal{E}$  is the vision tokenizer mapping images to  $m$  visual tokens  $\mathbf{V} \in \mathbb{R}^{m \times d}$ ,  $\mathbf{Q}_0 \in \mathbb{R}^{n \times d}$  are learnable causal query embeddings,  $\oplus$  denotes sequence concatenation,  $\mathcal{T}^L$  represents an  $L$ -layer Transformer with masked attention,  $\mathbf{M} \in \{0, 1\}^{2n \times 2n}$  is the block causal attention mask defined in Equation 1,  $\pi_Q$  is the projection operator that extracts the last  $n$  tokens (i.e.,  $\mathbf{Z} = \mathbf{X}_{m+1:m+n}$ ),  $\mathcal{D}$  is the language decoder, and  $\mathbf{O} \in \mathbb{R}^{n \times |\mathcal{V}|}$  is the output logits over LLM vocabulary.



## 4. Experimental Settings

### 4.1. Data Engine

DeepSeek-OCR 2 employs the same data sources as DeepSeek-OCR, comprising OCR 1.0, OCR 2.0 [11, 27, 53], and general vision data [54], with OCR data constituting 80% of the training mixture. We also introduce two modifications: (1) a more balanced sampling strategy for OCR 1.0 data, partitioning pages by content type (text, formulas, tables) with a 3:1:1 ratio, and (2) label refinement for layout detection by merging semantically similar categories (e.g., unifying "figure caption" and "figure title"). Given these minimal differences, we consider DeepSeek-OCR a valid baseline for comparison.

### 4.2. Training Pipelines

We train DeepSeek-OCR 2 in three stages: (1) encoder pretraining, (2) query enhancement, and (3) decoder specialization. The stage-1 enables the vision tokenizer and LLM-style encoder to acquire fundamental capabilities in feature extraction, token compression, and token reordering capabilities. The stage-2 further strengthens the token reordering capability of the encoder while enhancing visual knowledge compression. The stage-3 freezes the encoder parameters and optimizes only the decoder, enabling higher data throughput under the same FLOPs.

#### 4.2.1. Training DeepEncoder V2

Following DeepSeek-OCR and Vary [50], we train DeepEncoder V2 using a language modeling objective, coupling the encoder with a lightweight decoder [20] for joint optimization via next token prediction. We employ two dataloaders at resolutions of 768×768 and 1024×1024. The vision tokenizer is initialized from DeepEncoder, and the LLM-like encoder from Qwen2-0.5B-base [48]. After pretraining, only the encoder parameters are retained for subsequent stages. We use the AdamW [32] optimizer with cosine learning rate decay from 1e-4 to 1e-6, training on 160 A100 GPUs (20 nodes × 8 GPUs) with batch size 640 for 40k iterations (with sequence packing at 8K length, about 100M image-text pair samples).

#### 4.2.2. Query enhancement

After DeepEncoder V2 pretraining, we integrate it with DeepSeek-3B-A500M [24, 25] as our final pipeline. We freeze the visual tokenizer (SAM-conv structure) while jointly optimizing the LLM encoder and LLM decoder to enhance query representations. At this stage, we unify the two resolutions into a single dataloader via multi-crop strategy. We adopt 4-stage pipeline parallelism: vision tokenizer (PP0), LLM-style encoder (PP1), and DeepSeek-LLM layers (6 layers per stage on PP2-3). With 160 GPUs (40GB/per-GPU), we configure 40 data parallel replicas (4 GPUs per replica) and train with global batch size 1280 using the same optimizer and learning rate decay from 5e-5 to 1e-6 over 15k iterations.

#### 4.2.3. Continue-training LLM

To rapidly consume training data, we freeze all DeepEncoder V2 parameters in this stage and only update the DeepSeek-LLM parameters. This stage accelerates training (more than doubles the training speed under the same global batch size) while helping the LLM better understand DeepEncoder V2’s reordered visual tokens. Continuing from stage-2, we perform another learning rate decay from 1e-6 to 5e-8 training for 20k iterations in this stage.



Table 1 | Comprehensive evaluation of document reading on OmniDocBench v1.5. V-token<sup>max</sup> represents the maximum number of visual tokens used per page in this benchmark. R-order denotes reading order. Except for DeepSeek OCR and DeepSeek OCR 2, all other model results in this table are sourced from the OmniDocBench repository.

Model	V-token <sup>max</sup> ↓	Overall ↑	Text <sup>Edit</sup> ↓	Formula <sup>CDM</sup> ↑	Table <sup>TEDs</sup> ↑	Table <sup>TEDS<sub>s</sub></sup> ↑	R-order <sup>Edit</sup> ↑
<b>Pipeline</b>							
Marker-1.8.2 [1]	-	71.30	0.206	76.66	57.88	71.17	0.250
MinerU2-pp [45]	-	71.51	0.209	76.55	70.90	79.11	0.225
Dolphin [17]	-	74.67	0.125	67.85	68.70	77.77	0.124
Dolphin-1.5 [17]	-	83.21	0.092	80.78	78.06	84.10	0.080
PP-StructureV3 [13]	-	86.73	0.073	85.79	81.68	89.48	0.073
MonkeyOCR-pro-1.2B [23]	-	86.96	0.084	85.02	84.24	89.02	0.130
MonkeyOCR-3B [23]	-	87.13	0.075	87.45	81.39	85.92	0.129
MonkeyOCR-pro-3B [23]	-	88.85	0.075	87.25	86.78	90.63	0.128
MinerU2.5 [45]	-	90.67	0.047	88.46	88.22	92.38	0.044
PaddleOCR-VL [12]	-	92.86	0.035	91.22	90.89	94.76	0.043
<b>End-to-end Model</b>							
OCRFlux [4]	>6000	74.82	0.193	68.03	75.75	80.23	0.202
GPT-4o [33]	-	75.02	0.217	79.70	67.07	76.09	0.148
InternVL3 [55]	>7000	80.33	0.131	83.42	70.64	77.74	0.113
POINTS-Reader [31]	>6000	80.98	0.134	79.20	77.13	81.66	0.145
olmOCR [36]	>6000	81.79	0.096	86.04	68.92	74.77	0.121
InternVL3.5-241B [49]	>7000	82.67	0.142	87.23	75.00	81.28	0.125
MinerU2-VLM [45]	>7000	85.56	0.078	80.95	83.54	87.66	0.086
Nanonets-OCR-s [2]	>7000	85.59	0.093	85.90	80.14	85.57	0.108
Qwen2.5-VL-72B [9]	>6000	87.02	0.094	88.27	82.15	86.22	0.102
Gemini-2.5 Pro[6]	-	88.03	0.075	85.82	85.71	90.29	0.097
dots.ocr [39]	>6000	88.41	0.048	83.22	86.78	90.62	0.053
OCRVerse [3]	>6000	88.56	0.058	86.91	84.55	88.45	0.071
Qwen3-VL-235B [8]	>6000	89.15	0.069	88.14	86.21	90.55	0.068
DeepSeek-OCR (9-crops)	1156	87.36	0.073	84.14	85.25	89.01	0.085
DeepSeek-OCR 2	1120	91.09	0.048	90.31	87.75	92.06	0.057
	↓ 36	↑ 3.73	↓ 0.025	↑ 6.17	↑ 2.5	↑ 3.05	↓ 0.028

## 5. Evaluation

We select OmniDocBench v1.5 [34] as our primary benchmark for evaluation. This benchmark comprises 1,355 document pages spanning 9 major categories (including magazines, academic papers, research reports, and so on) in both Chinese and English. With its diverse test samples and robust evaluation criteria, OmniDocBench provides an effective framework for validating the performance of DeepSeek-OCR 2, particularly the effectiveness of DeepEncoder V2.

### 5.1. Main Results

As shown in Table 1, DeepSeek-OCR 2 achieves advanced performance of 91.09% while using the smallest upper limit of visual tokens (V-token<sup>max</sup>). Compared to the DeepSeek-OCR baseline, it demonstrates a 3.73% improvement under similar train data sources, validating the effectiveness of our newly designed architecture. Beyond the overall improvement, the Edit Distance (ED) for reading order (R-order) has also significantly decreased (from 0.085 to 0.057), indicating that the new DeepEncoder V2 can effectively select and arrange initial visual tokens based on image information. As illustrated in Table 2, DeepSeek-OCR 2 (0.100) achieves lower ED in document

Table 2 | Edit Distances for different categories of document-elements in OmniDocBench v1.5. V-token<sup>max</sup> denotes the lowest maximum number of visual tokens.

Model	V-token <sup>max</sup> ↓	Text <sup>Edit</sup> ↓	Formula <sup>Edit</sup> ↓	Table <sup>Edit</sup> ↓	R-order <sup>Edit</sup> ↓	Overall <sup>Edit</sup> ↓
Gemini-3 pro [44]	1120	-	-	-	-	0.115
Seed-1.8 [41]	5120	-	-	-	-	0.106
DeepSeek-OCR	1156	0.073	0.236	0.123	0.085	0.129
DeepSeek-OCR 2	1120	0.048	0.198	0.096	0.057	0.100

parsing compared to Gemini-3 Pro (0.115) under a similar visual token budget (1120), further demonstrating that our new model maintains high compression rates of visual tokens while ensuring superior performance, with exceptionally high potential.

## 5.2. Improvement Headroom

We conduct a detailed performance comparison between DeepSeek-OCR and DeepSeek-OCR 2 across 9 document types and found that DeepSeek-OCR 2 still has considerable room for improvement, as shown in Table 3. For text recognition Edit Distance (ED), DeepSeek-OCR 2 outperforms DeepSeek-OCR in most cases, but there are also notable weaknesses, such as newspapers, where it performs > 0.13 ED. We believe there are two main reasons: (1) the lower upper limit of visual tokens may affect the recognition of text-super-rich newspapers, which can be simply addressed in the future by increasing the number of local crops; (2) insufficient newspaper data – our training data contains only 250k relevant samples, which is inadequate for training DeepEncoder V2 for this class. Of course, for the reading order (R-order) metric, DeepSeek-OCR 2 consistently outperforms DeepSeek-OCR across the board, further validating the effectiveness of our visual causal flow encoder design.

Table 3 | Detailed comparison between DeepSeek-OCR 2 and DeepSeek-OCR across 9 document types. R-order denotes reading order. All metrics are Edit Distances, where lower is better.

Model	Edit ↓	PPT	Academic Paper	Book	Colorful Textbook	Exam Paper	Magazine	Newspaper	Note	Research Report
DS-OCR	Text	0.052	0.028	0.022	0.130	0.074	0.049	0.131	0.145	0.015
	R-order	0.052	0.021	0.040	0.125	0.083	0.101	0.217	0.089	0.016
DS-OCR 2	Text	0.031	0.013	0.033	0.053	0.047	0.026	0.139	0.068	0.008
	R-order	0.025	0.013	0.027	0.066	0.048	0.100	0.176	0.035	0.011

## 5.3. Practical Readiness

DeepSeek-OCR serves two primary production use cases: an online OCR service that reads image/documents for DeepSeek-LLMs, and a pretraining data pipeline that performs batch PDF processing. We compare the production performance between DeepSeek-OCR 2 and DeepSeek-OCR. Since ground truth is unavailable in production environments, we focus primarily on repetition rate as our key metric. As shown in Table 4, DeepSeek-OCR 2 demonstrates markedly improved practical readiness compared to its predecessor (DeepSeek-OCR), reducing the repetition rate from 6.25% to 4.17% for online user-log images, and from 3.69% to 2.88% for PDF data production. These results further validate the effectiveness of the DeepSeek-OCR 2 architecture, particularly its logical visual comprehension capabilities.

Table 4 | Production performance comparison between DeepSeek-OCR and DeepSeek-OCR 2. For OCR models serving LLM pipelines, ground truth is not accessible in production environments. Therefore, Repetition rate constitutes the primary observable quality metric.

Model	Metric	online-user-logs (image)	pretrain-data (PDF)
DeepSeek-OCR	Repeat ↓	6.25%	3.69%
DeepSeek-OCR 2		4.17% ↓ 2.08%	2.88% ↓ 0.81%

## 6. Discussion and Future Works

### 6.1. Towards Genuine 2D Reasoning

DeepSeek-OCR 2 presents a novel architectural paradigm with an LLM-style encoder cascaded with an LLM decoder. This cascade of two 1D causal reasoners holds promise for genuine 2D reasoning: the encoder performs reading logic reasoning (causally reordering visual information through query tokens), while the decoder executes visual task reasoning over these causally-ordered representations. Decomposing 2D understanding into two complementary/orthogonal 1D causal reasoning subtasks may represent a breakthrough toward genuine 2D reasoning. Of course, achieving this goal remains a long journey. For instance, to enable multiple re-examinations and multi-hop reordering of visual content, we may need substantially longer causal flow tokens than the original visual token sequence. We will continue to refine this architecture and explore its effectiveness on general visual reasoning tasks in future work.

### 6.2. Towards Native Multimodality

DeepEncoder V2 provides initial validation of the LLM-style encoder’s viability for visual tasks. More importantly, this architecture enjoys the potential to evolve into a unified omni-modal encoder: a single encoder with shared  $W_k, W_v$  projections, attention mechanisms, and FFNs can process multiple modalities through modality-specific learnable query embeddings. Such an encoder could compress text, extract speech features, and reorganize visual content within the same parameter space, differing only in the learned weights of their query embeddings. DeepSeek-OCR’s optical compression represents an initial exploration toward native multimodality, while we believe DeepSeek-OCR 2’s LLM-style encoder architecture marks our further step in this direction. We will also continue exploring the integration of additional modalities through this shared encoder framework in the future.

## 7. Conclusion

In this technical report, we present DeepSeek-OCR 2, a significant upgrade to DeepSeek-OCR, that maintains high visual token compression while achieving meaningfully performance improvements. This advancement is powered by the newly proposed DeepEncoder V2, which implicitly distills causal understanding of the visual world through the integration of both bidirectional and causal attention mechanisms, leading to causal reasoning capabilities in the vision encoder and, consequently, marked lifts in visual reading logic.

While optical text reading, particularly document parsing, represents one of the most practical vision tasks in the LLM era, it constitutes only a small part of the broader visual understanding landscape. Looking ahead, we will refine and adapt this architecture to more diverse scenarios, seeking deeper toward a more comprehensive vision of multimodal intelligence.

## References

- [1] Marker. URL <https://github.com/datalab-to/marker>.
- [2] Nanonets-ocr-s, 2025. URL <https://huggingface.co/nanonets/Nanonets-OCR-s>.
- [3] Ocrverse, 2025. URL <https://github.com/DocTron-hub/OCRVerse>.
- [4] Ocrflux, 2025. URL <https://github.com/chatdoc-com/OCRFlux>.
- [5] Adept. Fuyu-8b. <https://huggingface.co/adept/fuyu-8b>, 2023.
- [6] G. AI. Gemini 2.5-pro, 2025. URL <https://gemini.google.com/>.
- [7] J. Bai, S. Bai, S. Yang, S. Wang, S. Tan, P. Wang, J. Lin, C. Zhou, and J. Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. arXiv preprint arXiv:2308.12966, 2023.
- [8] S. Bai, Y. Cai, R. Chen, et al. Qwen3-vl technical report. arXiv preprint arXiv:2511.21631, 2025. URL <https://arxiv.org/abs/2511.21631>.
- [9] S. Bai, K. Chen, X. Liu, J. Wang, W. Ge, S. Song, K. Dang, P. Wang, S. Wang, J. Tang, H. Zhong, Y. Zhu, M. Yang, Z. Li, J. Wan, P. Wang, W. Ding, Z. Fu, Y. Xu, J. Ye, X. Zhang, T. Xie, Z. Cheng, H. Zhang, Z. Yang, H. Xu, and J. Lin. Qwen2.5-vl technical report. arXiv preprint arXiv:2502.13923, 2025.
- [10] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko. End-to-end object detection with transformers. In European conference on computer vision, pages 213–229. Springer, 2020.
- [11] J. Chen, L. Kong, H. Wei, C. Liu, Z. Ge, L. Zhao, J. Sun, C. Han, and X. Zhang. Onechart: Purify the chart structural extraction via one auxiliary token. In Proceedings of the 32nd ACM International Conference on Multimedia, pages 147–155, 2024.
- [12] C. Cui, T. Sun, S. Liang, et al. Paddleocr-vl: Boosting multilingual document parsing via a 0.9 b ultra-compact vision-language model. arXiv preprint arXiv:2510.14528, 2025.
- [13] C. Cui, T. Sun, M. Lin, T. Gao, Y. Zhang, J. Liu, X. Wang, Z. Zhang, C. Zhou, H. Liu, et al. Paddleocr 3.0 technical report. arXiv preprint arXiv:2507.05595, 2025.
- [14] M. Dehghani, J. Djolonga, B. Mustafa, P. Padlewski, J. Heek, J. Gilmer, A. Steiner, M. Caron, R. Geirhos, I. Alabdulmohsin, et al. Patch n’ pack: Navit, a vision transformer for any aspect ratio and resolution. Advances in Neural Information Processing Systems, 36:3632–3656, 2023.
- [15] J. Devlin, M.-W. Chang, K. Lee, and K. 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.
- [16] A. Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929, 2020.
- [17] H. Feng, S. Wei, X. Fei, W. Shi, Y. Han, L. Liao, J. Lu, B. Wu, Q. Liu, C. Lin, et al. Dolphin: Document image parsing via heterogeneous anchor prompting. arXiv preprint arXiv:2505.14059, 2025.

- [18] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778, 2016.
- [19] A. Huang, C. Yao, C. Han, F. Wan, H. Guo, H. Lv, H. Zhou, J. Wang, J. Zhou, J. Sun, et al. Step3-vl-10b technical report. arXiv preprint arXiv:2601.09668, 2026.
- [20] S. Iyer, X. V. Lin, R. Pasunuru, T. Mihaylov, D. Simig, P. Yu, K. Shuster, T. Wang, Q. Liu, P. S. Koura, et al. Opt-impl: Scaling language model instruction meta learning through the lens of generalization. arXiv preprint arXiv:2212.12017, 2022.
- [21] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, et al. Segment anything. arXiv preprint arXiv:2304.02643, 2023.
- [22] J. Li, D. Li, S. Savarese, and S. Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In International conference on machine learning, pages 19730–19742. PMLR, 2023.
- [23] Z. Li, Y. Liu, Q. Liu, Z. Ma, Z. Zhang, S. Zhang, Z. Guo, J. Zhang, X. Wang, and X. Bai. Monkeyocr: Document parsing with a structure-recognition-relation triplet paradigm. arXiv preprint arXiv:2506.05218, 2025.
- [24] A. Liu, B. Feng, B. Wang, B. Wang, B. Liu, C. Zhao, C. Deng, C. Ruan, D. Dai, D. Guo, et al. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. arXiv preprint arXiv:2405.04434, 2024.
- [25] A. Liu, B. Feng, B. Xue, B. Wang, B. Wu, C. Lu, C. Zhao, C. Deng, C. Zhang, C. Ruan, et al. Deepseek-v3 technical report. arXiv preprint arXiv:2412.19437, 2024.
- [26] A. Liu, A. Mei, B. Lin, B. Xue, B. Wang, B. Xu, B. Wu, B. Zhang, C. Lin, C. Dong, et al. Deepseek-v3. 2: Pushing the frontier of open large language models. arXiv preprint arXiv:2512.02556, 2025.
- [27] C. Liu, H. Wei, J. Chen, L. Kong, Z. Ge, Z. Zhu, L. Zhao, J. Sun, C. Han, and X. Zhang. Focus anywhere for fine-grained multi-page document understanding. arXiv preprint arXiv:2405.14295, 2024.
- [28] F. Liu and H. Qiu. Context cascade compression: Exploring the upper limits of text compression. arXiv preprint arXiv:2511.15244, 2025.
- [29] F. Liu, H. Wei, W. Zhao, G. Li, J. Peng, and Z. Li. Wb-detr: Transformer-based detector without backbone. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 2979–2987, 2021.
- [30] Y. Liu, J. Gu, N. Goyal, X. Li, S. Edunov, M. Ghazvininejad, M. Lewis, and L. Zettlemoyer. Multilingual denoising pre-training for neural machine translation. Transactions of the Association for Computational Linguistics, 8:726–742, 2020.
- [31] Y. Liu, Z. Zhao, L. Tian, et al. Points-reader: Distillation-free adaptation of vision-language models for document conversion. In Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 1576–1601, November 2025.
- [32] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. In ICLR, 2019.
- [33] OpenAI. Gpt-4 technical report, 2023.

- [34] L. Ouyang, Y. Qu, H. Zhou, J. Zhu, R. Zhang, Q. Lin, B. Wang, Z. Zhao, M. Jiang, X. Zhao, et al. Omnidocbench: Benchmarking diverse pdf document parsing with comprehensive annotations. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 24838–24848, 2025.
- [35] Z. Pang, Z. Xie, Y. Man, and Y.-X. Wang. Frozen transformers in language models are effective visual encoder layers. *arXiv preprint arXiv:2310.12973*, 2023.
- [36] J. Poznanski, A. Rangapur, J. Borchardt, J. Dunkelberger, R. Huff, D. Lin, C. Wilhelm, K. Lo, and L. Soldaini. olmocr: Unlocking trillions of tokens in pdfs with vision language models. *arXiv preprint arXiv:2502.18443*, 2025.
- [37] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.
- [38] J. Redmon and A. Farhadi. Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017.
- [39] Rednote. dots.ocr, 2025. URL <https://github.com/rednote-hilab/dots.ocr>.
- [40] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.
- [41] B. Seed. Seed1. 8 model card: Towards generalized real-world agency, 2025.
- [42] J. Su, Y. Lu, S. Pan, B. Wen, and Y. Liu. Roformer: Enhanced transformer with rotary position embedding. *arXiv preprint arXiv:2104.09864*, 2021.
- [43] C. Team. Chameleon: Mixed-modal early-fusion foundation models. *arXiv preprint arXiv:2405.09818*, 2024.
- [44] G. Team, R. Anil, S. Borgeaud, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, K. Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.
- [45] B. Wang, C. Xu, X. Zhao, L. Ouyang, F. Wu, Z. Zhao, R. Xu, K. Liu, Y. Qu, F. Shang, et al. Mineru: An open-source solution for precise document content extraction. *arXiv preprint arXiv:2409.18839*, 2024.
- [46] B. Wang, B. Wang, C. Wan, G. Huang, H. Hu, H. Jia, H. Nie, M. Li, N. Chen, S. Chen, et al. Step-3 is large yet affordable: Model-system co-design for cost-effective decoding. *arXiv preprint arXiv:2507.19427*, 2025.
- [47] C. Wang, S. Chen, Y. Wu, Z. Zhang, L. Zhou, S. Liu, Z. Chen, Y. Liu, H. Wang, J. Li, et al. Neural codec language models are zero-shot text to speech synthesizers. *arXiv preprint arXiv:2301.02111*, 2023.
- [48] P. Wang, S. Bai, S. Tan, S. Wang, Z. Fan, J. Bai, K. Chen, X. Liu, J. Wang, W. Ge, et al. Qwen2-vl: Enhancing vision-language model’s perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

- [49] W. Wang, Z. Gao, L. Gu, et al. Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. [arXiv preprint arXiv:2508.18265](#), 2025.
- [50] H. Wei, L. Kong, J. Chen, L. Zhao, Z. Ge, J. Yang, J. Sun, C. Han, and X. Zhang. Vary: Scaling up the vision vocabulary for large vision-language model. In [European Conference on Computer Vision](#), pages 408–424. Springer, 2024.
- [51] H. Wei, L. Kong, J. Chen, L. Zhao, Z. Ge, E. Yu, J. Sun, C. Han, and X. Zhang. Small language model meets with reinforced vision vocabulary. [arXiv preprint arXiv:2401.12503](#), 2024.
- [52] H. Wei, C. Liu, J. Chen, J. Wang, L. Kong, Y. Xu, Z. Ge, L. Zhao, J. Sun, Y. Peng, et al. General ocr theory: Towards ocr-2.0 via a unified end-to-end model. [arXiv preprint arXiv:2409.01704](#), 2024.
- [53] H. Wei, Y. Yin, Y. Li, J. Wang, L. Zhao, J. Sun, Z. Ge, X. Zhang, and D. Jiang. Slow perception: Let’s perceive geometric figures step-by-step. [arXiv preprint arXiv:2412.20631](#), 2024.
- [54] H. Wei, Y. Sun, and Y. Li. Deepseek-ocr: Contexts optical compression. [arXiv preprint arXiv:2510.18234](#), 2025.
- [55] J. Zhu, W. Wang, Z. Chen, Z. Liu, S. Ye, L. Gu, H. Tian, Y. Duan, W. Su, J. Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. [arXiv preprint arXiv:2504.10479](#), 2025.
- [56] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai. Deformable detr: Deformable transformers for end-to-end object detection. [arXiv preprint arXiv:2010.04159](#), 2020.