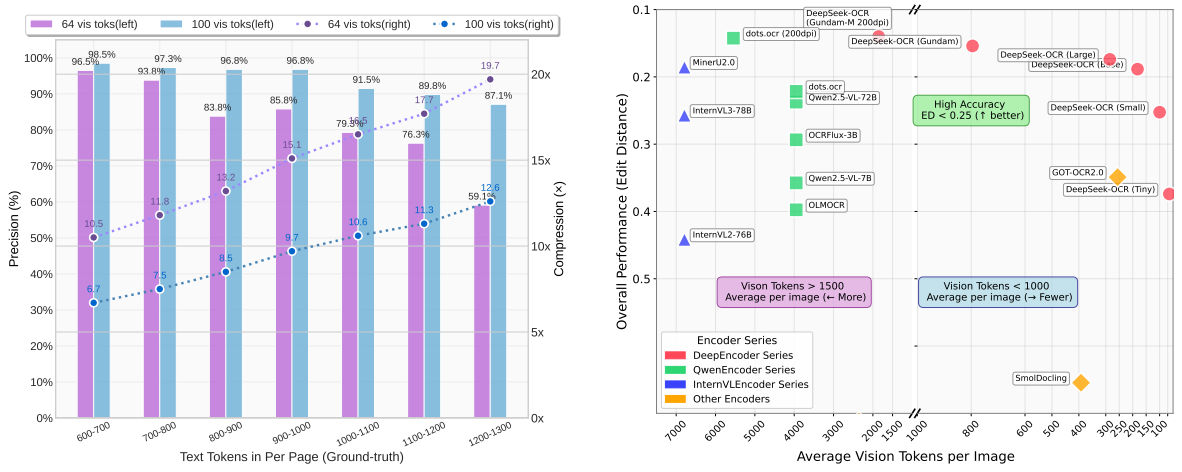
DeepSeek-OCR：上下文光学压缩

Haoran Wei, Yaofeng Sun, Yukun Li

DeepSeek-AI

Abstract

我们提出 DeepSeek-OCR，作为通过光学二维映射压缩长上下文的可行性初步探索。DeepSeek-OCR 由两部分组成：DeepEncoder 以及作为解码器的 DeepSeek3B-MoE-A570M。具体而言，DeepEncoder 作为核心引擎，旨在高分辨率输入下保持低激活状态，同时实现高压缩比，以确保视觉 token 数量既最优又易于管理。实验表明，当文本 token 数量在视觉 token 数量的 10 倍以内（即压缩比 $< 10\times$）时，模型可实现 97% 的解码（OCR）精度。即使在 $20\times$ 的压缩比下，OCR 准确率仍保持在 60% 左右。这为历史长上下文压缩以及大语言模型（LLM）中的记忆遗忘机制等研究领域展现了巨大潜力。除此之外，DeepSeek-OCR 还展现出极高的实用价值。在 OmniDocBench 基准测试中，它仅使用 100 个视觉 token 便超越了 GOT-OCR2.0（256 tokens/页），并在视觉 token 使用量少于 800 个的情况下优于 MinerU2.0（平均每页 6000+ tokens）。在实际生产环境中，DeepSeek-OCR 每天可为 LLMs/VLMs 生成超过 20 万页的训练数据（单张 A100-40G 显卡）。代码与模型权重已公开于 http://github.com/deepseek-ai/DeepSeek-OCR。



(a) Compression on Fox benchmark

(b) Performance on Omnidocbench

图 1 | 图 (a) 展示了在 Fox [21] 基准测试上的压缩比（真实文本 token 数/模型使用的视觉 token 数）；图 (b) 展示了在 OmniDocBench [27] 上的性能对比。DeepSeek-OCR 在使用最少视觉 token 的情况下，实现了端到端模型中的最先进性能。

目录

1. Introduction

当前的大语言模型（LLMs）在处理长文本内容时面临巨大的计算挑战，因为其计算成本随序列长度呈二次方增长。我们探索了一种潜在的解决方案：利用视觉模态作为文本信息的高效压缩媒介。一张包含文档文本的图像可以使用比等效数字文本少得多的 token 来表示丰富的信息，这表明通过视觉 token 进行的光学压缩有望实现更高的压缩比。

这一洞察促使我们从以 LLM 为中心的视角重新审视视觉语言模型（VLMs），重点关注视觉编码器如何提升 LLM 处理文本信息的效率，而非人类已擅长的基础 VQA [12, 16, 24, 32, 41] 任务。作为连接视觉与语言的中间模态，OCR 任务为这种视觉-文本压缩范式提供了理想的测试平台，因为它在视觉与文本表征之间建立了自然的压缩-解压映射，同时提供了定量评估指标。

因此，我们提出 DeepSeek-OCR，这是一个旨在作为高效视觉-文本压缩初步概念验证的 VLM。我们的工作主要贡献如下：首先，我们对视觉-文本 token 压缩比进行了全面的定量分析。在具有多样化文档布局的 Fox [21] 基准测试上，我们的方法在 9-10× 文本压缩下实现了 96%+ 的 OCR 解码精度，在 10-12× 压缩下达到 ~90%，在 20× 压缩下达到 ~60%（如图 1(a) 所示；若考虑输出与真实标签之间的格式差异，实际精度甚至更高）。结果表明，紧凑的语言模型能够有效学习解码压缩后的视觉表征，这表明更大的 LLM 可以通过适当的预训练设计轻松获得类似的能力。

其次，我们提出了 DeepEncoder，这是一种新颖的架构，即使在高分辨率输入下也能保持较低的激活内存和极少的视觉 token。它通过一个 16× 卷积压缩器将窗口注意力编码器组件与全局注意力编码器组件串行连接。该设计确保了窗口注意力组件能够处理大量视觉 token，同时压缩器在 token 进入密集全局注意力组件之前将其数量减少，从而实现了有效的内存和 token 压缩。

第三，我们基于 DeepEncoder 和 DeepSeek3B-MoE [19, 20] 开发了 DeepSeek-OCR。如图 1(b) 所示，它在 OmniDocBench 上以端到端模型中使用的最少视觉 token 实现了最先进的性能。此外，我们赋予该模型解析图表、化学公式、简单几何图形和自然图像的能力，以进一步提升其实用价值。在生产环境中，DeepSeek-OCR 仅需 20 个节点（每个节点配备 8 张 A100-40G GPU）即可每天为 LLM 或 VLM 生成 3300 万页数据。

综上所述，本文初步探索了将视觉模态作为 LLM 中文本信息处理的高效压缩媒介。通过 DeepSeek-OCR，我们证明了视觉-文本压缩可以为不同的历史上下文阶段实现显著的 token 减少（7-20×），为应对大语言模型的长上下文挑战提供了一个有前景的方向。我们的定量分析为 VLM token 分配优化提供了经验指导，而提出的 DeepEncoder 架构则展示了具备实际部署能力的可行性。尽管本文以 OCR 作为概念验证，但该范式为重新思考如何协同结合视觉与语言模态以提升大规模文本处理和智能体系统的计算效率开辟了新的可能性。

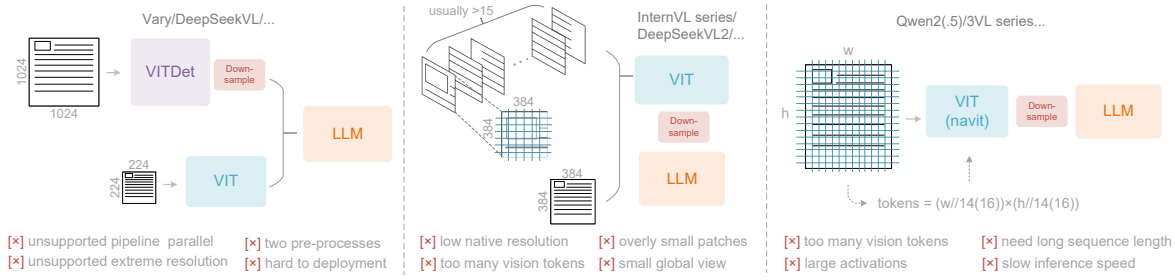


图 2 | 流行 VLM 中的典型视觉编码器。以下是当前开源 VLM 中常用的三种编码器类型，它们均存在各自的缺陷。

2. 相关工作

2.1. VLM 中的典型视觉编码器

如图 2 所示，当前的开源 VLM 主要采用三种类型的视觉编码器。第一种是以 Vary [36] 为代表的双塔架构，它利用并行的 SAM [17] 编码器来增加视觉词汇参数，以处理高分辨率图像。尽管该方法提供了可控的参数和激活内存，但存在显著缺点：它需要双重图像预处理，这使部署变得复杂，并在训练期间使编码器流水线并行变得困难。第二种是以 InternVL2.0 [8] 为代表的基于图块（tile）的方法，它通过将图像划分为小图块进行并行计算来处理图像，从而在高分辨率设置下减少激活内存。尽管该方法能够处理极高的分辨率，但由于其原生编码器分辨率通常较低（低于 512×512），导致大图像被过度碎片化，从而产生大量视觉 token。第三种是以 Qwen2-VL [35] 为代表的自适应分辨率编码，它采用 NaViT [10] 范式，通过基于 patch 的分割直接处理完整图像，而无需图块并行化。尽管该编码器能够灵活处理多种分辨率，但在处理大图像时面临巨大挑战，因为巨大的激活内存消耗可能导致 GPU 内存溢出，且序列打包在训练期间需要极长的序列长度。过长的视觉 token 会拖慢推理的预填充（prefill）和生成阶段。

2.2. 端到端 OCR 模型

OCR，尤其是文档解析任务，一直是图像到文本领域的一个高度活跃的话题。随着 VLM 的发展，大量端到端 OCR 模型应运而生，通过简化 OCR 系统，从根本上改变了传统的流水线架构（传统架构需要独立的检测和识别专家模型）。Nougat [6] 首次将端到端框架应用于 arXiv 学术论文的 OCR，展示了模型在处理密集感知任务方面的潜力。GOT-OCR2.0 [38] 将 OCR2.0 的范围扩展到包含更多合成图像解析任务，并设计了一个在性能与效率之间取得平衡的 OCR 模型，进一步凸显了端到端 OCR 研究的潜力。此外，Qwen-VL 系列 [35]、InternVL 系列 [8] 等通用视觉模型及其众多衍生模型不断增强其文档 OCR 能力，以探索密集视觉感知的边界。然而，当前模型尚未解决的一个关键研究问题是：*对于包含 1000 个单词的文档，解码至少需要多少个视觉 token？*这个问题对于研究“一图胜千言”这一原则具有重要意义。

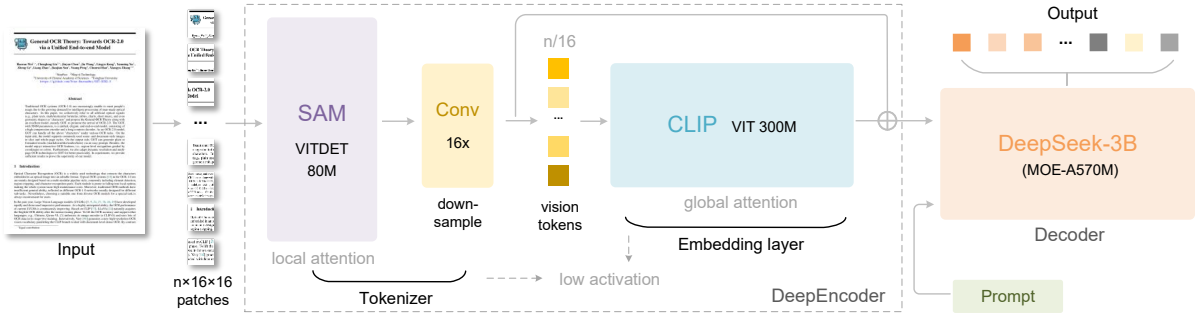


图 3 | DeepSeek-OCR 的架构。DeepSeek-OCR 由一个 DeepEncoder 和一个 DeepSeek-3B-MoE 解码器组成。DeepEncoder 是 DeepSeek-OCR 的核心，包含三个组件：一个以窗口注意力为主导用于感知的 SAM [17]，一个具有密集全局注意力用于知识的 CLIP [29]，以及一个连接两者的 16× token 压缩器。

3. 方法论

3.1. 架构

如图 3 所示，DeepSeek-OCR 采用统一的端到端 VLM 架构，由编码器和解码器组成。编码器（即 DeepEncoder）负责提取图像特征，并对视觉表征进行 token 化和压缩。解码器用于基于图像 token 和提示生成所需的结果。DeepEncoder 的参数规模约为 380M，主要由串联的 80M SAM-base [17] 和 300M CLIP-large [29] 组成。解码器采用 3B MoE [19, 20] 架构，激活参数为 570M。在接下来的段落中，我们将深入探讨模型组件、数据工程和训练技巧。

3.2. DeepEncoder

为了探索上下文光学压缩的可行性，我们需要一个具备以下特征的视觉编码器：1. 能够处理高分辨率；2. 高分辨率下激活内存低；3. 视觉 token 数量少；4. 支持多分辨率输入；5. 参数量适中。然而，如第 2.1 节所述，当前的开源编码器无法完全满足所有这些条件。因此，我们自行设计了一种新颖的视觉编码器，命名为 DeepEncoder。

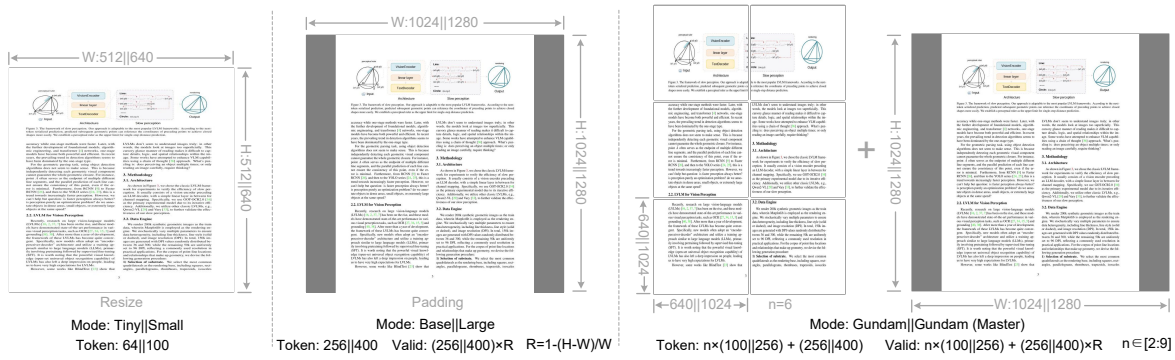


图 4 | 为了测试模型在不同压缩比（需要不同数量的视觉 token）下的性能，并增强 DeepSeek-OCR 的实用性，我们为其配置了多种分辨率模式。

3.2.1. DeepEncoder 的架构

DeepEncoder 主要由两个组件组成：一个以窗口注意力为主导的视觉感知特征提取组件，以及一个具有密集全局注意力的视觉知识特征提取组件。为了受益于先前工作的预训练成果，我们分别使用 SAM-base（patch 大小为 16）和 CLIP-large 作为这两个组件的主要架构。对于 CLIP，我们移除了第一个 patch 嵌入层，因为其输入不再是图像，而是来自前序流水线的输出 token。在两个组件之间，我们借鉴了 Vary [36] 的设计，使用一个 2 层卷积模块对视觉 token 进行 16× 下采样。每个卷积层的核大小为 3，步长为 2，填充为 1，通道数从 256 增加到 1024。假设我们输入一张 1024×1024 的图像，DeepEncoder 会将其分割为 1024/16×1024/16=4096 个 patch token。由于编码器的前半部分以窗口注意力为主且参数量仅为 80M，其激活内存是可以接受的。在进入全局注意力之前，这 4096 个 token 会经过压缩模块，token 数量变为 4096/16=256，从而使整体激活内存可控。

表 1 | DeepEncoder 的多分辨率支持。出于研究和应用的目的，我们为 DeepEncoder 设计了多种原生分辨率和动态分辨率模式。

模式	原生分辨率				动态分辨率	
	Tiny	Small	Base	Large	Gundam	Gundam-M
分辨率	512	640	1024	1280	640+1024	1024+1280
Token 数量	64	100	256	400	n×100+256	n×256+400
处理方式	缩放	缩放	填充	填充	缩放 + 填充	缩放 + 填充

3.2.2. 多分辨率支持

假设我们有一张包含 1000 个光学字符的图像，并希望测试解码需要多少个视觉 token。这要求模型支持可变数量的视觉 token。也就是说，DeepEncoder 需要支持多种分辨率。

我们通过位置编码的动态插值来满足上述要求，并设计了几种分辨率模式进行同步模型训练，以实现单个 DeepSeek-OCR 模型支持多种分辨率的能力。如图 4 所示，DeepEncoder 主要支持两种输入模式：原生分辨率和动态分辨率。每种模式均包含多个子模式。原生分辨率支持四种子模式：Tiny、Small、Base 和 Large，对应的分辨率和 Token 数量分别为 512×512（64）、640×640（100）、1024×1024（256）和 1280×1280（400）。由于 Tiny 和 Small 模式的分辨率相对较小，为避免浪费视觉 Token，图像通过直接缩放原始形状进行处理。对于 Base 和 Large 模式，为了保持原始图像的宽高比，图像会被填充至对应尺寸。填充后，有效视觉 Token 的数量小于实际视觉 Token 的数量，计算公式为：

$$N_{valid} = \lceil N_{actual} \times [1 - ((max(w,h) - min(w,h))/(max(w,h)))] \rceil \quad (1)$$

其中 w 和 h 分别表示原始输入图像的宽度和高度。

动态分辨率可由两种原生分辨率组合而成。例如，Gundam 模式由 n×640×640 的图块（局部视图）和一个 1024×1024 的全局视图组成。分块方法遵循 InternVL2.0 [8]。支持动态分辨率主

要是出于应用层面的考虑，特别是针对超高分辨率输入（如报纸图像）。分块是一种二次窗口注意力机制，能够进一步有效降低激活内存。值得注意的是，由于我们的原生分辨率相对较大，在动态分辨率下图像不会被过度碎片化（图块数量控制在 2 到 9 之间）。DeepEncoder 在 Gundam 模式下输出的视觉 Token 数量为：$n \times 100 + 256$，其中 n 为图块数量。对于宽高均小于 640 的图像，n 设为 0，即 Gundam 模式将退化为 Base 模式。

Gundam 模式与四种原生分辨率模式一起进行训练，以实现单个模型支持多种分辨率的目标。需要注意的是，Gundam-master 模式（1024×1024 局部视图 + 1280×1280 全局视图）是通过对已训练的 DeepSeek-OCR 模型进行继续训练获得的。这主要是出于负载均衡的考虑，因为 Gundam-master 的分辨率过大，若一起训练会拖慢整体训练速度。

3.3. MoE 解码器

我们的解码器采用 DeepSeekMoE [19, 20]，具体为 DeepSeek-3B-MoE。在推理阶段，模型从 64 个路由专家中激活 6 个，并激活 2 个共享专家，激活参数量约为 5.7 亿。3B 规模的 DeepSeekMoE 非常适合以特定领域（对我们而言是 OCR）为中心的 VLM 研究，因为它在获得 3B 模型表达能力的同时，享有 5 亿参数小模型的推理效率。

解码器从 DeepEncoder 压缩后的潜在视觉 Token 中重建原始文本表示，公式如下：

$$f_{\text{dec}} : \mathbb{R}^{n \times d_{\text{latent}}} \rightarrow \mathbb{R}^{N \times d_{\text{text}}}; \quad \hat{\mathbf{X}} = f_{\text{dec}}(\mathbf{Z}) \quad \text{where } n \leq N \tag{2}$$

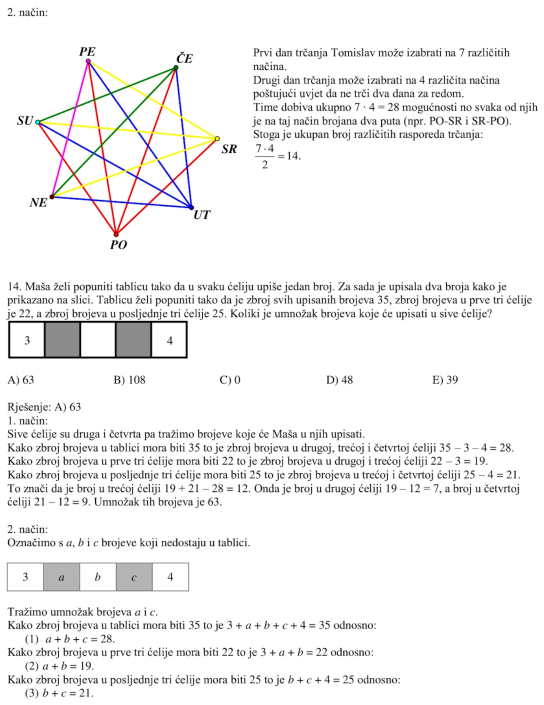
其中 $\mathbf{Z} \in \mathbb{R}^{n \times d_{\text{latent}}}$ 是来自 DeepEncoder 的压缩潜在（视觉）Token，$\hat{\mathbf{X}} \in \mathbb{R}^{N \times d_{\text{text}}}$ 是重建后的文本表示。函数 f_{dec} 表示一种非线性映射，紧凑的语言模型可以通过 OCR 风格的训练有效学习该映射。可以合理推测，LLM 通过专门的预训练优化，将展现出对此类能力更自然的融合。

3.4. 数据引擎

我们为 DeepSeek-OCR 构建了复杂且多样化的训练数据，包括 OCR 1.0 数据，主要包含场景图像 OCR 和文档 OCR 等传统 OCR 任务；OCR 2.0 数据，主要包含复杂人工图像的解析任务，如常见图表、化学公式和平面几何解析数据；通用视觉数据，主要用于向 DeepSeek-OCR 注入一定的通用图像理解能力，并保留通用视觉接口。

3.4.1. OCR 1.0 数据

文档数据是 DeepSeek-OCR 的首要任务。我们从互联网上收集了 3000 万页涵盖约 100 种语言的多样化 PDF 数据，其中中英文约占 2500 万页，其他语言占 500 万页。针对这些数据，我们创建了两类真实标签：粗标注和精细标注。粗标注直接使用 *fitz* 从完整数据集中提取，旨在教会模型识别光学文本，尤其是少数语言文本。精细标注包含中英文各 200 万页，使用先进的布局模型（如 PP-DocLayout [33]）和 OCR 模型（如 MinuerU [34] 和 GOT-OCR2.0 [38]）进行标注，以构建检测与识别交错的数据。对于少数语言，在检测部分，我们发现布局模型具备一定的



(a) Ground truth image


```
<|ref|>text<|/ref|><|det|>[[55, 43, 130, 60]]<|/det|>
2. način:

<|ref|>image<|/ref|><|det|>[[70, 93, 450, 360]]<|/det|>


<|ref|>text<|/ref|><|det|>[[460, 95, 896, 132]]<|/det|>
Prvi dan trčanja Tomislav može izabrati na 7 različitih načina.

<|ref|>text<|/ref|><|det|>[[460, 131, 880, 168]]<|/det|>
Drugi dan trčanja može izabrati na 4 različita načina poštujući uvjet da ne trči
dva dana za redom.

<|ref|>text<|/ref|><|det|>[[460, 166, 941, 220]]<|/det|>
Time dobiva ukupno \(7 \cdot 4 = 28\) mogućnosti no svaka od njih je na taj način
brojana dva puta (npr. PO-SR i SR-PO). Stoga je ukupan broj različitih rasporeda
trčanja:

<|ref|>equation<|/ref|><|det|>[[460, 217, 550, 256]]<|/det|>
\[ \frac{7 \cdot 4}{2} = 14. \]

<|ref|>text<|/ref|><|det|>[[55, 397, 931, 452]]<|/det|>
14. Maša želi popuniti tablicu tako da u svaku ćeliju upiše jedan broj. Za sada
je upisala dva broja kako je prikazano na slici. Tablicu želi popuniti tako da je
zbroj svih upisanih brojeva 35, zbroj brojeva u prve tri ćelije je 22, a zbroj
brojeva u posljednje tri ćelije 25. Koliki je umnožak brojeva koje će upisati u
sive ćelije?

<|ref|>table<|/ref|><|det|>[[57, 450, 360, 500]]<|/det|>
<table><tr><td>3</td><td></td><td></td><td></td><td>4</td></tr></table>

<|ref|>text<|/ref|><|det|>[[55, 515, 110, 534]]<|/det|>
A) 63

<|ref|>text<|/ref|><|det|>[[230, 515, 293, 534]]<|/det|>
B) 108

<|ref|>text<|/ref|><|det|>[[405, 515, 450, 534]]<|/det|>
C) 0

<|ref|>text<|/ref|><|det|>[[581, 515, 636, 534]]<|/det|>
D) 48
```


(b) Fine annotations with layouts

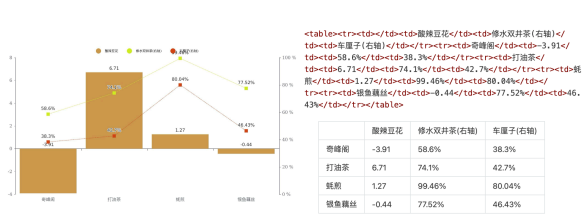
图 5 | OCR 1.0 精细标注展示。我们将真实标签格式化为布局与文本交错的格式，其中每段文本前均附有其在原始图像中的坐标和标签。所有坐标均归一化为 1000 个区间。

泛化能力。在识别部分，我们使用 *fitz* 创建小图块数据来训练一个 GOT-OCR2.0，然后使用训练好的模型在布局处理后对小图块进行标注，利用模型飞轮机制创建了 60 万条数据样本。在训练 DeepSeek-OCR 时，粗标签和精细标签通过不同的提示词进行区分。精细标注图文对的真实标签如图 5 所示。我们还收集了 300 万条 *Word* 数据，通过直接提取内容构建无布局的高质量图文对。该数据主要对公式和 HTML 格式表格带来收益。此外，我们选择了一些开源数据 [28, 37] 作为补充。

对于自然场景 OCR，我们的模型主要支持中英文。图像数据来源为 LAION [31] 和 Wukong [13]，使用 PaddleOCR [9] 进行标注，中英文各 1000 万条数据样本。与文档 OCR 类似，自然场景 OCR 也可以通过提示词控制是否输出检测框。

3.4.2. OCR 2.0 数据

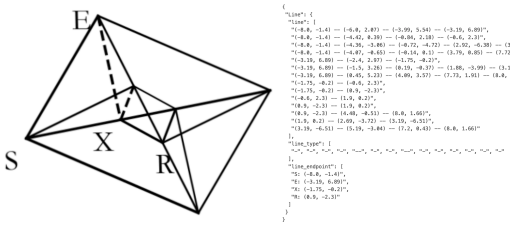
遵循 GOT-OCR2.0 [38]，我们将图表、化学公式和平面几何解析数据统称为 OCR 2.0 数据。对于图表数据，遵循 OneChart [7]，我们使用 pyecharts 和 matplotlib 渲染了 1000 万张图像，主要包括常用的折线图、柱状图、饼图和组合图。我们将图表解析定义为图像到 HTML 表格的转换任务，如图 6(a) 所示。对于化学公式，我们以 PubChem 的 SMILES 格式作为数据源，并使用 RDKit 将其渲染为图像，构建了 500 万对图文数据。对于平面几何图像，我们遵循 Slow Perception [39] 进行生成。具体而言，我们将 perception-ruler 大小设为 4 来对每条线段进行建模。为了增加渲染数据的多样性，我们引入了几何平移不变数据增强，即在同一原始图像中平移



<table><tr><td></td><td>醍辣豆花</td><td>修水双井茶(右轴)</td><td>车厘子(右轴)</td></tr><tr><td>奇峰阁</td><td>-3.91</td><td>58.6%</td><td>38.3%</td></tr><tr><td>打油茶</td><td>6.71</td><td>74.1%</td><td>42.7%</td></tr><tr><td>蚝煎</td><td>1.27</td><td>99.46%</td><td>80.04%</td></tr><tr><td>银鱼藤丝</td><td>-0.44</td><td>77.52%</td><td>46.43%</td></tr></table>

	醍辣豆花	修水双井茶(右轴)	车厘子(右轴)
奇峰阁	-3.91	58.6%	38.3%
打油茶	6.71	74.1%	42.7%
蚝煎	1.27	99.46%	80.04%
银鱼藤丝	-0.44	77.52%	46.43%

(a) Image-text ground truth of chart



{
"Line": {
"Line": [
"(-0.0, -1.4) — (-6.0, 2.87) — (-3.99, 5.54) — (-3.19, 6.09)",
"(-0.0, -1.4) — (-6.42, 6.39) — (-6.04, 2.18) — (-6.6, 2.17)",
"(-0.0, -1.4) — (-4.39, -3.06) — (-0.72, -4.72) — (2.92, -6.38) — (3.19, -6.51)",
"(-0.0, -1.4) — (-4.07, -6.65) — (-0.14, 8.1) — (3.79, 8.05) — (7.72, 1.6) — (8.0, 1.66)",
"(-3.19, 6.09) — (-2.4, 2.87) — (-1.75, -0.2)",
"(-3.19, 6.09) — (-1.3, 3.26) — (0.19, -0.37) — (1.88, -3.99) — (3.19, -6.51)",
"(-3.19, 6.09) — (0.45, 5.23) — (4.09, 3.57) — (7.72, 1.91) — (8.0, 1.66)",
"(-1.75, -0.2) — (0.9, 2.3)",
"(-1.75, -0.2) — (0.9, -2.3)",
"(-0.6, 2.3) — (1.9, 0.2)",
"(0.9, -2.3) — (1.9, 0.2)",
"(0.9, -2.3) — (4.48, -4.51) — (8.0, 1.66)",
"(1.9, 0.2) — (2.09, -3.72) — (3.19, -6.51)",
"(3.19, -6.51) — (5.19, -3.84) — (7.2, 0.43) — (8.0, 1.66)"
],
"Line_type": [
"-","-","-","-","-","-","-","-","-","-","-","-","-","-","-"
]
},
"Line_endpoint": {
"S": (-0.0, -1.4)",
"E": (-3.19, 6.09)",
"X": (-1.75, -0.2)",
"R": (0.9, -2.3)"
}
}

(b) Image-text ground truth of geometry

图 6 | 对于图表，我们未采用 OneChart [7] 的字典格式，而是使用 HTML 表格格式作为标签，这可以节省一定数量的 Token。对于平面几何，我们将真实标签转换为字典格式，其中字典包含线段、端点坐标、线段类型等键，以提高可读性。每条线段均采用 Slow Perception [39] 的方式进行编码。

相同的几何图像，对应于在坐标系中心位置绘制的相同真实标签。在此基础上，我们总共构建了 100 万条平面几何解析数据，如图 6(b) 所示。

3.4.3. 通用视觉数据

DeepEncoder 能够受益于 CLIP 的预训练成果，并拥有足够的参数来融入通用视觉知识。因此，我们也为 DeepSeek-OCR 准备了一些相应的数据。遵循 DeepSeek-VL2 [40]，我们生成了用于图像描述（caption）、检测和定位（grounding）等任务的相关数据。需要注意的是，DeepSeek-OCR 并非通用 VLM 模型，该部分数据仅占总数据的 20%。我们引入此类数据主要是为了保留通用视觉接口，以便对模型和通用视觉任务感兴趣的研究人员未来能够便捷地推进相关工作。

3.4.4. 纯文本数据

为确保模型的语言能力，我们引入了 10% 的内部纯文本预训练数据，所有数据均处理为 8192 个 Token 的长度，这也是 DeepSeek-OCR 的序列长度。综上所述，在训练 DeepSeek-OCR 时，OCR 数据占 70%，通用视觉数据占 20%，纯文本数据占 10%。

3.5. 训练流程

我们的训练流程非常简单，主要分为两个阶段：a) 独立训练 DeepEncoder；b) 训练 DeepSeek-OCR。需要注意的是，Gundam-master 模式是通过对预训练的 DeepSeek-OCR 模型使用 600 万条采样数据进行继续训练获得的。由于训练协议与其他模式相同，此处不再赘述。

3.5.1. 训练 *DeepEncoder*

遵循 Vary [36]，我们采用紧凑的语言模型 [15]，并使用下一个 Token 预测框架来训练 DeepEncoder。在此阶段，我们使用了前述的所有 OCR 1.0 和 2.0 数据，以及从 LAION [31] 数据集中采样的 1 亿条通用数据。所有数据训练 2 个 epoch，批次大小为 1280，使用带有余弦退火调度器 [22] 的 AdamW [23] 优化器，学习率为 5e-5。训练序列长度为 4096。

3.5.2. 训练 *DeepSeek-OCR*

DeepEncoder 准备就绪后，我们使用第 3.4 节中提到的数据来训练 DeepSeek-OCR，整个训练过程在 HAI-LLM [14] 平台上进行。整个模型采用流水线并行（PP），分为 4 个部分，其中 DeepEncoder 占 2 个部分，解码器占 2 个部分。对于 DeepEncoder，我们将 SAM 和压缩器视为视觉分词器，放置在 PP0 并冻结其参数，同时将 CLIP 部分视为输入嵌入层，放置在 PP1 并解冻权重进行训练。对于语言模型部分，由于 DeepSeek3B-MoE 有 12 层，我们在 PP2 和 PP3 上各放置 6 层。我们使用 20 个节点（每个节点配备 8 张 A100-40G GPU）进行训练，数据并行（DP）为 40，全局批次大小为 640。我们使用带有步长调度器的 AdamW 优化器，初始学习率为 3e-5。对于纯文本数据，训练速度为 900 亿 Token/天；对于多模态数据，训练速度为 700 亿 Token/天。

表 2 | 我们使用 Fox [21] 基准测试中所有包含 600-1300 个 Token 的英文文档，来测试 DeepSeek-OCR 的图文压缩率。Text tokens 表示使用 DeepSeek-OCR 的分词器对真实文本进行分词后的 Token 数量。Vision Tokens=64 或 100 分别表示将输入图像调整大小为 512×512 和 640×640 后，DeepEncoder 输出的视觉 Token 数量。

文本 Token 数	视觉 Token 数 =64		视觉 Token 数 =100		
	精确度	压缩率	精确度	压缩率	页数
600-700	96.5%	10.5×	98.5%	6.7×	7
700-800	93.8%	11.8×	97.3%	7.5×	28
800-900	83.8%	13.2×	96.8%	8.5×	28
900-1000	85.9%	15.1×	96.8%	9.7×	14
1000-1100	79.3%	16.5×	91.5%	10.6×	11
1100-1200	76.4%	17.7×	89.8%	11.3×	8
1200-1300	59.1%	19.7×	87.1%	12.6×	4

4. 评估

4.1. 图文压缩研究

我们选用 Fox [21] 基准测试来验证 DeepSeek-OCR 对富文本文档的压缩与解压能力，以初步探索上下文光学压缩的可行性与边界。我们使用 Fox 的英文文档部分，利用 DeepSeek-OCR 的分词器（词表大小约 129k）对真实文本进行分词，并选取包含 600-1300 个 Token 的文档进行测试，共计 100 页。由于文本 Token 数量不大，我们仅需在 Tiny 和 Small 模式下测试性能，其中 Tiny 模式对应 64 个 Token，Small 模式对应 100 个 Token。我们使用无布局的提示词：“<image>\nFree OCR.” 来控制模型的输出格式。尽管如此，输出格式仍无法与 Fox 基准测试完全匹配，因此实际性能会略高于测试结果。

如表 2 所示，在 10× 压缩率以内，模型的解码精确度可达到约 97%，这是一个非常令人鼓舞的结果。未来，或许可以通过文生图方法实现接近 10× 的无损上下文压缩。当压缩率超过 10× 时，性能开始下降，这可能有两个原因：一是长文档的布局变得更加复杂，二是长文本在

表 3 | 我们使用 OmniDocBench [27] 测试 DeepSeek-OCR 在真实文档解析任务上的性能。表中所有指标均为编辑距离，数值越小表示性能越好。“Tokens” 表示每页使用的平均视觉 Token 数量，而 “†200dpi” 表示使用 *fitz* 将原始图像插值到 200dpi。对于 DeepSeek-OCR 模型，“Tokens” 列中括号内的数值表示有效视觉 Token 数量，根据公式 1 计算得出。

模型	Token 数	英文					中文				
		整体	文本	公式	表格	顺序	整体	文本	公式	表格	顺序
流水线模型											
Dolphin [11]	-	0.356	0.352	0.465	0.258	0.35	0.44	0.44	0.604	0.367	0.351
Marker [1]	-	0.296	0.085	0.374	0.609	0.116	0.497	0.293	0.688	0.678	0.329
Mathpix [2]	-	0.191	0.105	0.306	0.243	0.108	0.364	0.381	0.454	0.32	0.30
MinerU-2.1.1 [34]	-	0.162	0.072	0.313	0.166	0.097	0.244	0.111	0.581	0.15	0.136
MonkeyOCR-1.2B [18]	-	0.154	0.062	0.295	0.164	0.094	0.263	0.179	0.464	0.168	0.243
PPstructure-v3 [9]	-	0.152	0.073	0.295	0.162	0.077	0.223	0.136	0.535	0.111	0.11
端到端模型											
Nougat [6]	2352	0.452	0.365	0.488	0.572	0.382	0.973	0.998	0.941	1.00	0.954
SmolDocling [25]	392	0.493	0.262	0.753	0.729	0.227	0.816	0.838	0.997	0.907	0.522
InternVL2-76B [8]	6790	0.44	0.353	0.543	0.547	0.317	0.443	0.29	0.701	0.555	0.228
Qwen2.5-VL-7B [5]	3949	0.316	0.151	0.376	0.598	0.138	0.399	0.243	0.5	0.627	0.226
OLMOCR [28]	3949	0.326	0.097	0.455	0.608	0.145	0.469	0.293	0.655	0.652	0.277
GOT-OCR2.0 [38]	256	0.287	0.189	0.360	0.459	0.141	0.411	0.315	0.528	0.52	0.28
OCRFlux-3B [3]	3949	0.238	0.112	0.447	0.269	0.126	0.349	0.256	0.716	0.162	0.263
GPT4o [26]	-	0.233	0.144	0.425	0.234	0.128	0.399	0.409	0.606	0.329	0.251
InternVL3-78B [42]	6790	0.218	0.117	0.38	0.279	0.095	0.296	0.21	0.533	0.282	0.161
Qwen2.5-VL-72B [5]	3949	0.214	0.092	0.315	0.341	0.106	0.261	0.18	0.434	0.262	0.168
dots.ocr [30]	3949	0.182	0.137	0.320	0.166	0.182	0.261	0.229	0.468	0.160	0.261
Gemini2.5-Pro [4]	-	0.148	0.055	0.356	0.13	0.049	0.212	0.168	0.439	0.119	0.121
MinerU2.0 [34]	6790	0.133	0.045	0.273	0.15	0.066	0.238	0.115	0.506	0.209	0.122
dots.ocr$^{\dagger 200dpi}$ [30]	5545	0.125	**0.032**	0.329	**0.099**	**0.04**	0.16	**0.066**	0.416	0.092	**0.067**
DeepSeek-OCR（端到端）											
Tiny	**64**	0.386	0.373	0.469	0.422	0.283	0.361	0.307	0.635	0.266	0.236
Small	100	0.221	0.142	0.373	0.242	0.125	0.284	0.24	0.53	0.159	0.205
Base	256(182)	0.137	0.054	0.267	0.163	0.064	0.24	0.205	0.474	0.1	0.181
Large	400(285)	0.138	0.054	0.277	0.152	0.067	0.208	0.143	0.461	0.104	0.123
Gundam	795	0.127	0.043	0.269	0.134	0.062	0.181	0.097	0.432	0.089	0.103
Gundam-M$^{\dagger 200dpi}$	1853	**0.123**	0.049	**0.242**	0.147	0.056	**0.157**	0.087	**0.377**	**0.08**	0.085

512×512 或 640×640 分辨率下会变得模糊。第一个问题可以通过将文本渲染到单一布局页面上来解决，而我们认为第二个问题将成为遗忘机制的一种特性。当将 Token 压缩近 20× 时，我们发现精确度仍能接近 60%。这些结果表明，上下文光学压缩是一个极具前景且值得研究的方向，并且该方法不会带来任何额外开销，因为它可以复用 VLM 的基础设施，而多模态系统本身就需要额外的视觉编码器。

4.2. OCR 实际性能

DeepSeek-OCR 不仅是一个实验性模型，它还具备强大的实际应用能力，可用于构建 LLM/VLM 预训练数据。为了量化 OCR 性能，我们在 OmniDocBench [27] 上测试了 DeepSeek-OCR，结

表 4 | OmniDocBench 中不同类别文档的编辑距离。结果表明，某些类型的文档仅需 64 或 100 个视觉 Token 即可取得良好性能，而其他类型则需要 Gundam 模式。

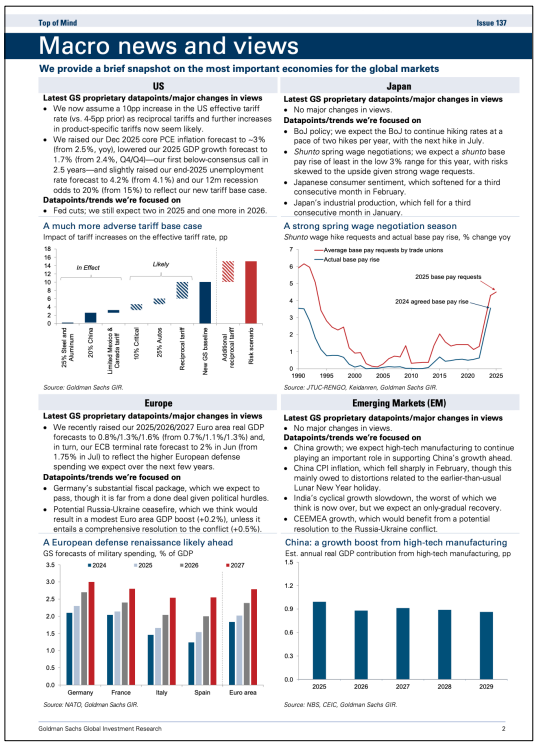
模式 \ 类型	书籍	幻灯片	财务报告	教科书	考试试卷	杂志	学术论文	笔记	报纸	整体
Tiny	0.147	0.116	0.207	0.173	0.294	0.201	0.395	0.297	0.94	0.32
Small	0.085	0.111	0.079	0.147	0.171	0.107	0.131	0.187	0.744	0.205
Base	0.037	0.08	0.027	0.1	0.13	0.073	0.052	0.176	0.645	0.156
Large	0.038	0.108	0.022	0.084	0.109	0.06	0.053	0.155	0.353	0.117
Gundam	0.035	0.085	0.289	0.095	0.094	0.059	0.039	0.153	0.122	0.083
Guandam-M	0.052	0.09	0.034	0.091	0.079	0.079	0.048	0.1	0.099	0.077

果如表 3 所示。仅需 100 个视觉 Token（640×640 分辨率），DeepSeek-OCR 便超越了使用 256 个 Token 的 GOT-OCR2.0 [38]；在使用 400 个 Token（285 个有效 Token，1280×1280 分辨率）时，其性能与该基准测试上的最先进模型持平。在使用不到 800 个 Token（Gundam 模式）的情况下，DeepSeek-OCR 的表现优于需要近 7,000 个视觉 Token 的 MinerU2.0 [34]。这些结果表明，我们的 DeepSeek-OCR 模型在实际应用中表现强劲，且由于具备更高的 Token 压缩率，它拥有更高的研究上限。As shown in Table 4, some categories of documents require very few tokens to achieve satisfactory performance, such as slides which only need 64 vision tokens. For book and report documents, DeepSeek-OCR can achieve good performance with only 100 vision tokens. Combined with the analysis from Section 4.1, this may be because most text tokens in these document categories are within 1,000, meaning the vision-token compression ratio does not exceed 10×. For newspapers, Gundam or even Gundam-master mode is required to achieve acceptable edit distances, because the text tokens in newspapers are 4-5,000, far exceeding the 10× compression of other modes. These experimental results further demonstrate the boundaries of contexts optical compression, which may provide effective references for researches on the vision token optimization in VLMs and context compression, forgetting mechanisms in LLMs.

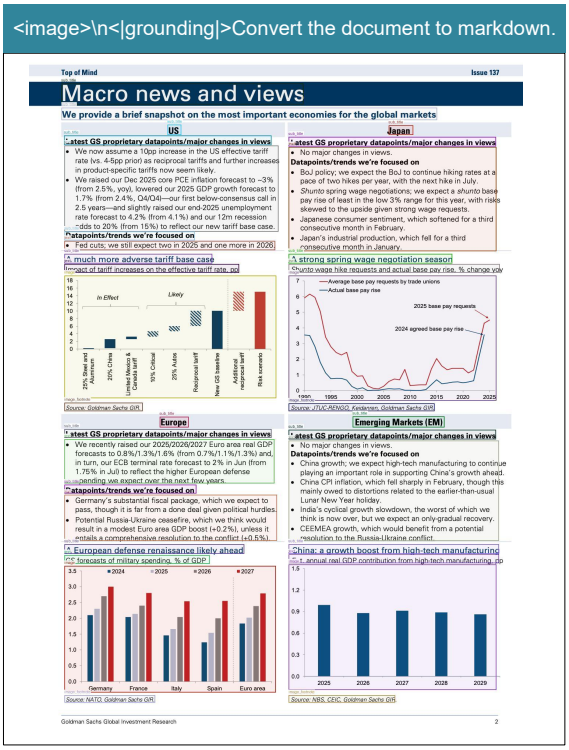
4.3. 定性研究

4.3.1. 深度解析

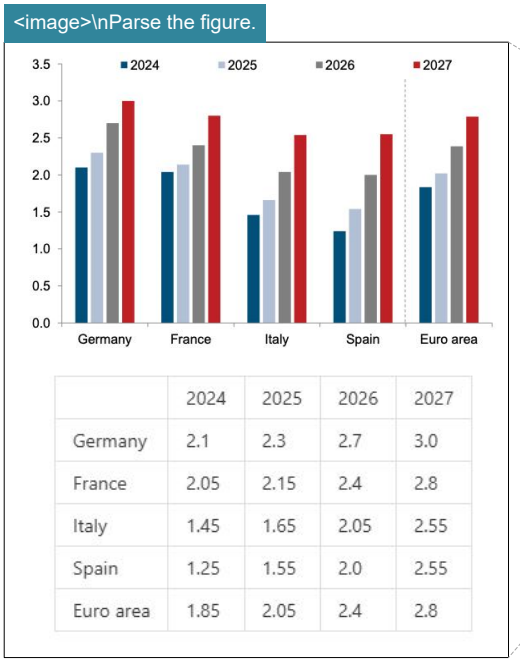
DeepSeek-OCR 兼具版式识别与 OCR 2.0 能力，能够通过二次模型调用进一步解析文档内的图像，我们将此功能称为 “深度解析”。如图 7、8、9、10 所示，我们的模型仅需统一的提示词，即可对图表、几何图形、化学公式甚至自然图像进行深度解析。



Input image

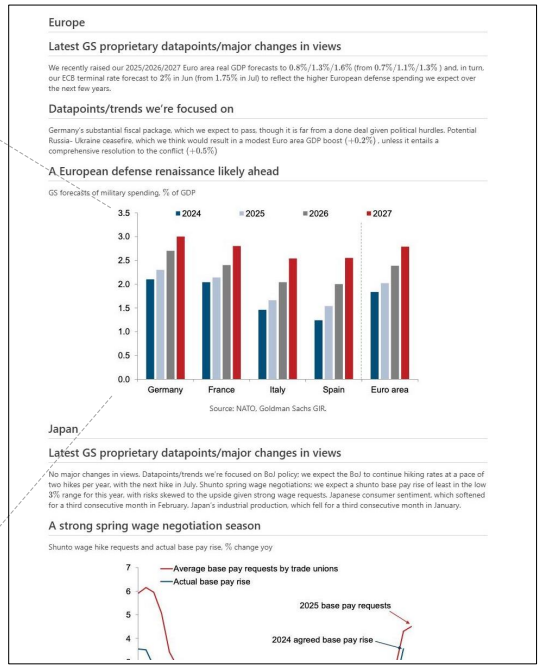


Result



	2024	2025	2026	2027
Germany	2.1	2.3	2.7	3.0
France	2.05	2.15	2.4	2.8
Italy	1.45	1.65	2.05	2.55
Spain	1.25	1.55	2.0	2.55
Euro area	1.85	2.05	2.4	2.8

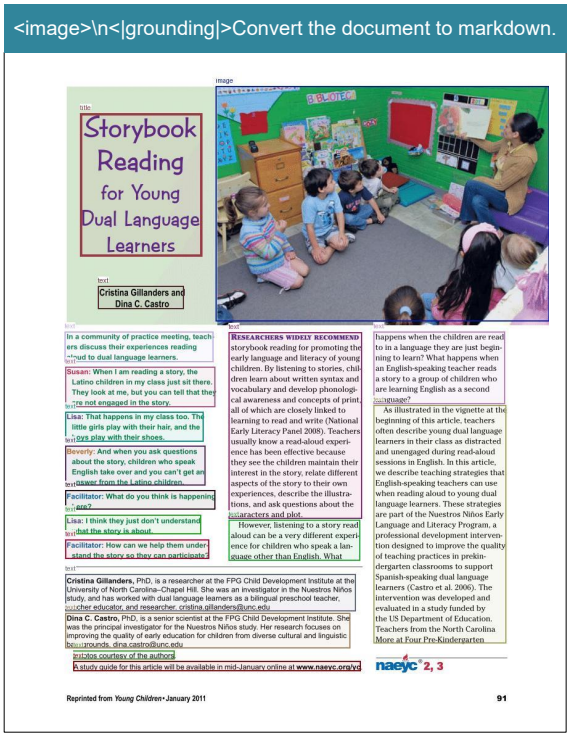
Deep Parsing



Rendering

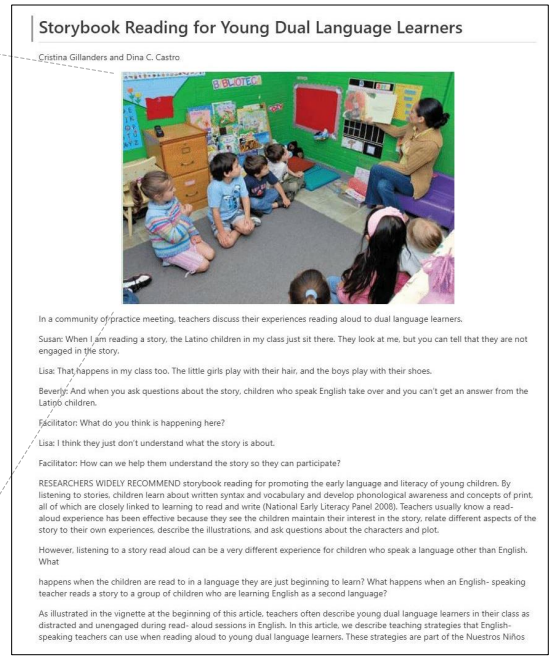
图 7 | 在金融研报领域，DeepSeek-OCR 的深度解析模式可用于获取文档内图表的结构化结果。图表是金融与科学领域重要的数据呈现形式，图表结构化提取是未来 OCR 模型不可或缺的能力。

Input image



Result

Deep Parsing



Rendering

图 8 | 对于书籍和文章，深度解析模式可为文档中的自然图像输出密集图注。仅需一个提示词，模型即可自动识别图像类型并输出所需结果。

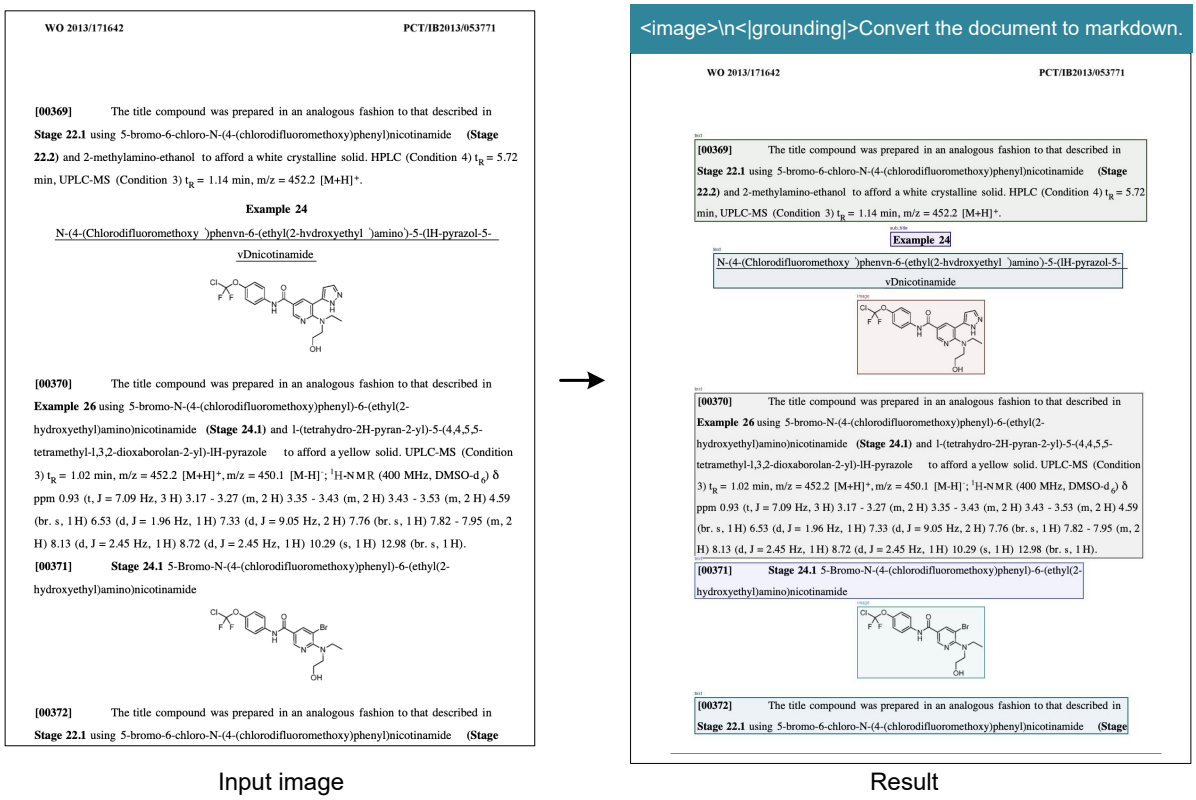
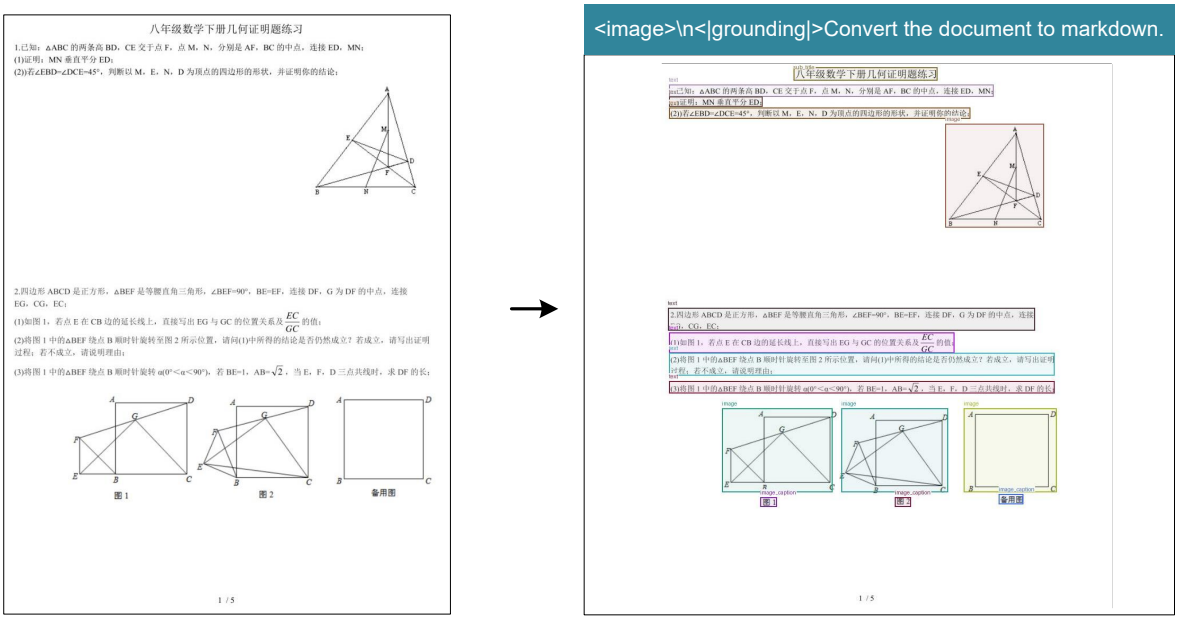


图 9 | 处于深度解析模式的 DeepSeek-OCR 还能识别化学文档中的化学公式并将其转换为 SMILES 格式。未来，OCR 1.0+2.0 技术有望在 STEM 领域的 VLM/LLM 发展中发挥重要作用。



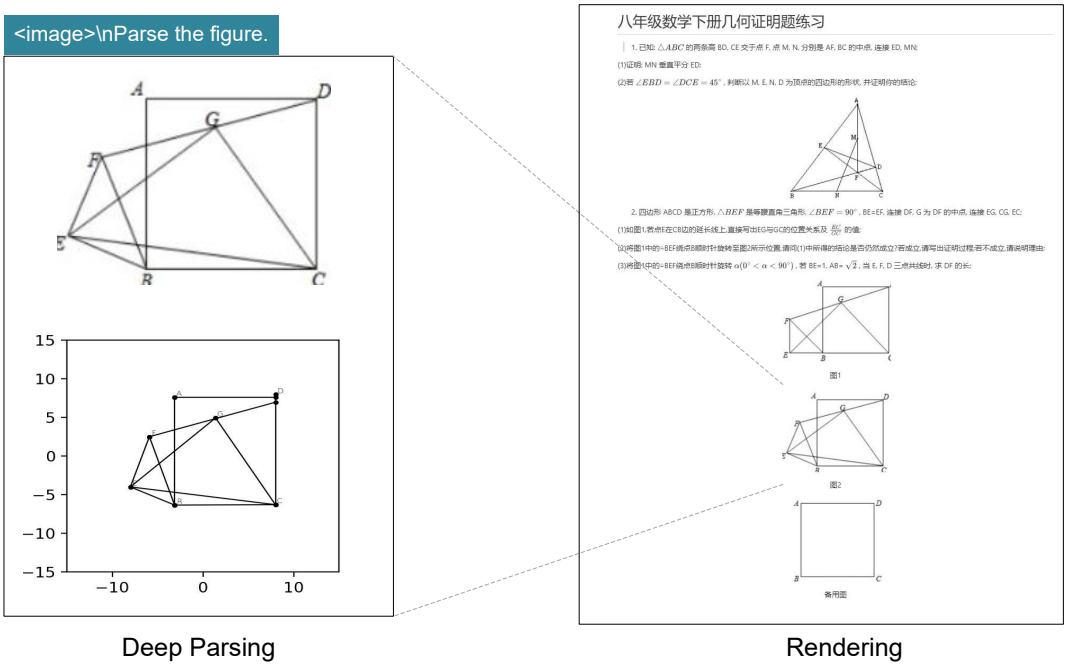


图 10 | DeepSeek-OCR 还具备复制（结构化）简单平面几何图形的能力。由于几何图形中线段之间存在复杂的相互依赖关系，几何解析任务极具挑战性，仍有漫长的探索之路。

4.3.2. 多语言识别

互联网上的 PDF 数据不仅包含中英文，还包含大量多语言数据，这对 LLM 的训练同样至关重要。针对 PDF 文档，DeepSeek-OCR 可处理近 100 种语言。与中英文文档类似，多语言数据也同时支持保留版式与纯文本的 OCR 格式。可视化结果如图 11 所示，我们选取阿拉伯语和僧伽罗语进行结果展示。

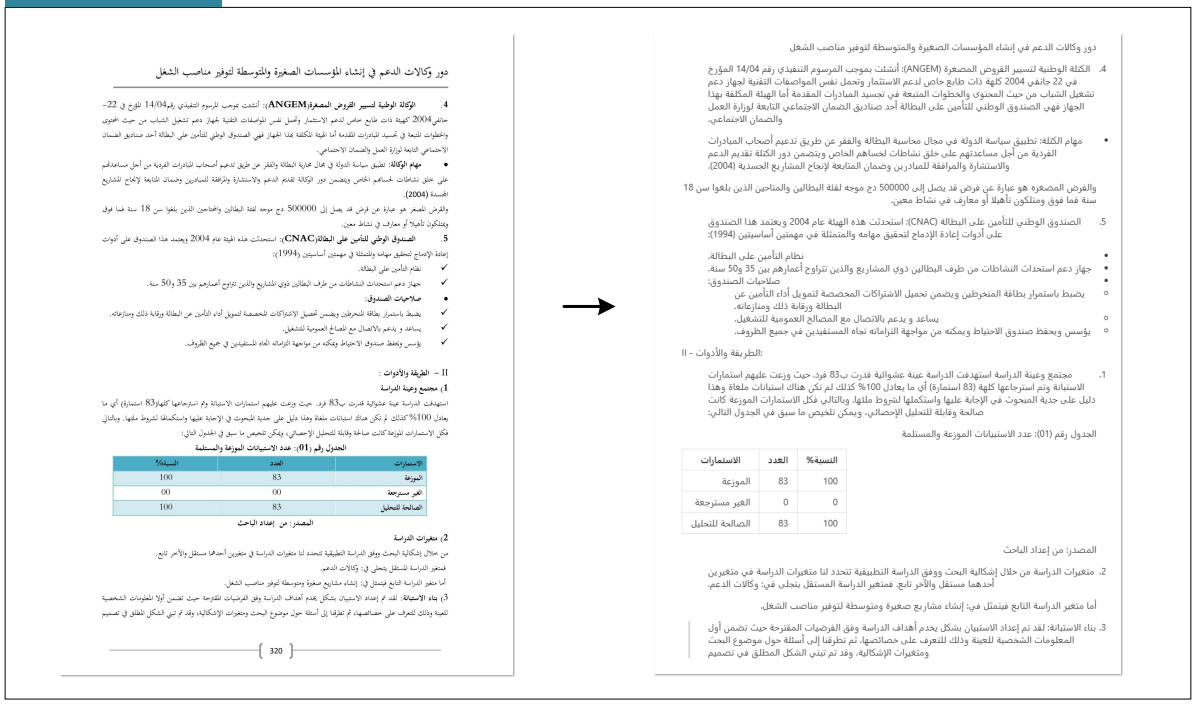


图 11 | 为赋予模型处理广泛爬取的 PDF（多语言数据）的能力，我们使用近 100 种语言的 OCR 数据对模型进行了训练。少数民族语言文档也可通过不同的提示词支持保留版式与纯文本的输出。

4.3.3. 通用视觉理解

我们还为 DeepSeek-OCR 赋予了一定程度的通用图像理解能力。相关可视化结果如图 12 所示。

图 12 | 我们保留了 DeepSeek-OCR 在通用视觉理解方面的能力，主要包括图像描述、目标检测、视觉定位（grounding）等。同时，由于训练数据中包含纯文本数据，DeepSeek-OCR 的语言能力也得以保留。需要注意的是，由于我们未包含 SFT（监督微调）阶段，该模型并非聊天机器人，部分能力需通过续写提示词来激活。

5. 讨论

我们的工作是对视觉-文本压缩边界的一次初步探索，旨在研究解码 N 个文本 Token 需要多少个视觉 Token。初步结果令人鼓舞：DeepSeek-OCR 在约 10× 压缩率下实现了近乎无损的 OCR 压缩，而在 20× 压缩下仍保留了 60% 的准确率。这些发现为未来应用指明了富有前景的方向，例如在多轮对话中对超过 k 轮的历史对话进行光学处理，以实现 10× 的压缩效率。

对于更早期的上下文，我们可以逐步缩小渲染图像的分辨率，以进一步降低 Token 消耗。这

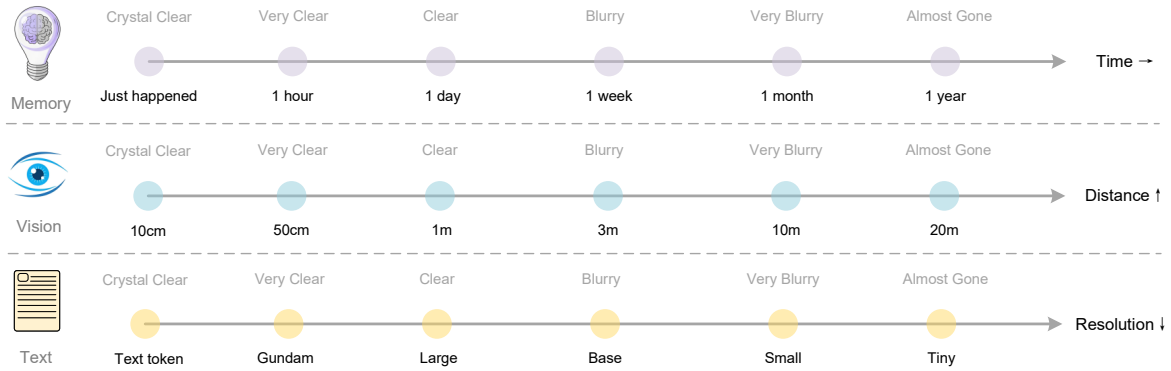


图 13 | 遗忘机制是人类记忆最基础的特征之一。上下文光学压缩方法可通过将前几轮的历史文本渲染为图像进行初始压缩，随后逐步缩小更早图像的分辨率以实现多级压缩，在此过程中 Token 数量逐渐减少且文本日益模糊，从而实现对文本的遗忘。

一假设的灵感来源于人类记忆随时间衰退与视觉感知随空间距离退化之间的自然类比——两者均表现出渐进式信息损失的相似模式，如图 13 所示。通过结合这些机制，上下文光学压缩方法实现了一种模拟生物遗忘曲线的记忆衰退形式：近期信息保持高保真度，而远期记忆则通过提高压缩率自然消退。

尽管我们的初步探索展示了可扩展超长上下文处理的潜力（近期上下文保持高分辨率，早期上下文消耗更少资源），但我们承认这仍处于早期阶段，需要进一步研究。该方法为构建理论上无限上下文架构提供了一条路径，该架构能在信息保留与计算约束之间取得平衡；不过，此类视觉-文本压缩系统的实际影响与局限性仍有待未来研究深入探讨。

6. 结论

在本技术报告中，我们提出了 DeepSeek-OCR，并通过该模型初步验证了上下文光学压缩的可行性，证明了该模型能够利用少量视觉 Token 有效解码数量超过其 10 倍的文本 Token。我们相信这一发现将促进未来 VLM 和 LLM 的发展。此外，DeepSeek-OCR 是一个实用性极强的模型，能够进行大规模预训练数据生产，是 LLM 开发过程中不可或缺的助手。当然，仅靠 OCR 尚不足以完全验证真正的上下文光学压缩，未来我们将开展数字-光学文本交错预训练、大海捞针测试及其他评估工作。从另一个角度来看，光学上下文压缩仍具有巨大的研究与改进空间，代表着一个充满前景的新方向。

参考文献

[1] Marker. URL https://github.com/datalab-to/marker.

[2] Mathpix. URL https://mathpix.com/.

[3] Ocrflux, 2025. URL https://github.com/chatdoc-com/OCRFlux.

[4] G. AI. Gemini 2.5-pro, 2025. URL https://gemini.google.com/.

[5] 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.

[6] L. Blecher, G. Cucurull, T. Scialom, and R. Stojnic. Nougat: Neural optical understanding for academic documents. arXiv preprint arXiv:2308.13418, 2023.

[7] 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.

[8] Z. Chen, W. Wang, H. Tian, S. Ye, Z. Gao, E. Cui, W. Tong, K. Hu, J. Luo, Z. Ma, et al. How far are we to gpt-4v? closing the gap to commercial multimodal models with open-source suites. arXiv preprint arXiv:2404.16821, 2024.

[9] 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.

[10] 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.

[11] 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.

[12] Y. Goyal, T. Khot, D. Summers-Stay, D. Batra, and D. Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 6904–6913, 2017.

[13] J. Gu, X. Meng, G. Lu, L. Hou, N. Minzhe, X. Liang, L. Yao, R. Huang, W. Zhang, X. Jiang, et al. Wukong: A 100 million large-scale chinese cross-modal pre-training benchmark. Advances in Neural Information Processing Systems, 35:26418–26431, 2022.

[14] High-flyer. HAI-LLM: Efficient and lightweight training tool for large models, 2023. URL https://www.high-flyer.cn/en/blog/hai-llm.

[15] 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-iml: Scaling language model instruction meta learning through the lens of generalization. arXiv preprint arXiv:2212.12017, 2022.

[16] S. Kazemzadeh, V. Ordonez, M. Matten, and T. Berg. Referitgame: Referring to objects in photographs of natural scenes. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pages 787–798, 2014.

[17] 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.

[18] 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.

[19] A. Liu, B. Feng, B. Wang, B. Wang, B. Liu, C. Zhao, C. Dengr, 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.

[20] 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.

[21] 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.

[22] I. Loshchilov and F. Hutter. Sgdr: Stochastic gradient descent with warm restarts. arXiv preprint arXiv:1608.03983, 2016.

[23] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. In ICLR, 2019.

[24] A. Masry, D. X. Long, J. Q. Tan, S. Joty, and E. Hoque. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. arXiv preprint arXiv:2203.10244, 2022.

[25] A. Nassar, A. Marafioti, M. Omenetti, M. Lysak, N. Livathinos, C. Auer, L. Morin, R. T. de Lima, Y. Kim, A. S. Gurbuz, et al. Smoldocling: An ultra-compact vision-language model for end-to-end multi-modal document conversion. arXiv preprint arXiv:2503.11576, 2025.

[26] OpenAI. Gpt-4 technical report, 2023.

[27] 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.

[28] 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.

[29] 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.

[30] Rednote. dots.ocr, 2025. URL https://github.com/rednote-hilab/dots.ocr.

[31] C. Schuhmann, R. Vencu, R. Beaumont, R. Kaczmarczyk, C. Mullis, A. Katta, T. Coombes, J. Jitsev, and A. Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. arXiv preprint arXiv:2111.02114, 2021.

[32] A. Singh, V. Natarajan, M. Shah, Y. Jiang, X. Chen, D. Batra, D. Parikh, and M. Rohrbach. Towards vqa models that can read. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 8317–8326, 2019.

[33] T. Sun, C. Cui, Y. Du, and Y. Liu. Pp-doclayout: A unified document layout detection model to accelerate large-scale data construction. arXiv preprint arXiv:2503.17213, 2025.

[34] 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.

[35] 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.

[36] 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.

[37] 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.

[38] 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.

[39] 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.

[40] Z. Wu, X. Chen, Z. Pan, X. Liu, W. Liu, D. Dai, H. Gao, Y. Ma, C. Wu, B. Wang, et al. Deepseek-vl2: Mixture-of-experts vision-language models for advanced multimodal understanding. arXiv preprint arXiv:2412.10302, 2024.

[41] W. Yu, Z. Yang, L. Li, J. Wang, K. Lin, Z. Liu, X. Wang, and L. Wang. Mm-vet: Evaluating large multimodal models for integrated capabilities. arXiv preprint arXiv:2308.02490, 2023.

[42] 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.