Gemini - 研究方案拟定与更新

DeepSeek-OCR：上下文光学压缩技术解析报告

摘要

原文翻译 我们提出 DeepSeek-OCR，作为通过光学二维映射压缩长上下文可行性的初步探索。DeepSeek-OCR 包含两个组件：DeepEncoder 和作为解码器的 DeepSeek3B-MoE-A570M。具体来说，DeepEncoder 作为核心引擎，旨在高分辨率输入下保持低激活，同时实现高压缩比，以确保视觉令牌（vision token）数量达到最优且可管理。实验表明，当文本令牌数量在视觉令牌数量的 10 倍以内时（即压缩比 < 10x），模型可以达到 97% 的解码（OCR）精度。即使在 20 倍的压缩比下，OCR 准确率仍能保持在 60% 左右。这为历史长上下文压缩和 LLM 中的记忆遗忘机制等研究领域展示了相当大的前景。除此之外，DeepSeek-OCR 还展示了很高的实用价值。在 OmniDocBench 上，它仅用 100 个视觉令牌就超过了 GOT-OCR2.0（256 令牌/页），并以少于 800 个视觉令牌的代价超越了 MinerU2.0（平均每页 6000+ 令牌）。在生产环境中，DeepSeek-OCR 每天（单个 A100-40G 显卡）可以为 LLM/VLM 生成超过 20 万页的训练数据。代码和模型权重已在 http://github.com/deepseek-ai/DeepSeek-OCR 公开。

深度解读 这篇论文的摘要开宗明义，提出了一个极具颠覆性的想法：“上下文光学压缩”。让我们把它拆解开来理解。首先，你需要知道当前所有大型语言模型（LLM），比如你可能听过的 GPT 系列，都面临一个巨大的挑战——处理“长上下文”。“上下文”就是模型需要处理的信息，比如一篇长文章或一段长对话。当信息变得非常长时，模型的计算量会急剧增加，变得非常缓慢且昂贵 。作者们想到的解决方案非常巧妙：我们为什么非要用纯文本的方式让模型“阅读”呢？能不能换一种更高效的媒介？他们的答案是：用眼睛“看”。

这里的“光学压缩”，你可以把它想象成这样一个过程：与其让 AI 一个字一个字地阅读一篇长达 5000 字的文章（这会产生大量的“文本令牌”，即文本的基本处理单元），不如把这篇文章拍成一张高清照片，然后让 AI 去“看”这张照片。这张照片本身所包含的信息单元，即“视觉令牌”，数量可能远远少于 5000 个文本令牌。如果 AI 能够通过看这张照片，完美地把里面的文字再还原出来，那就意味着我们用少量的视觉信息成功地“压缩”了大量的文本信息。这就是“上下文光学压缩”的核心思想。这篇论文就是为了验证这个想法是否可行而进行的初步探索。

摘要中的数据显示，这个想法不仅可行，而且效果惊人。在 10 倍的压缩比下（比如用 100 个视觉令牌代表 1000 个文本令牌的信息），模型还原文本的准确率高达 97%。这证明了 AI 完全有能力学会这种“看图识字”并解码压缩信息的能力。更重要的是，这不仅仅是一个有趣的实验。在实际的文档识别基准测试（OmniDocBench）中，DeepSeek-OCR 以极高的效率（使用更少的视觉令牌）击败了其他强大的模型，如 GOT-OCR2.0 和 MinerU2.0 。这表明它在现实世界中具有巨大的应用潜力，比如可以被用来大规模、低成本地处理海量文档，为其他 AI 模型制造训练数据。最后，作者提到这可能对研究“记忆遗忘机制”有启发，这是一个非常深刻的洞见，我们将在后面的讨论部分深入探讨。

图 1 | 性能展示

图表描述

图 1(a) Fox 基准测试的压缩性能

这是一个折线图，展示了模型在不同压缩比下的性能表现。

- **X 轴 (Compression (x))**: 压缩比，即原始文本令牌数除以模型使用的视觉令牌数。范围从 5 倍到 20 倍。
- **Y 轴 (Precision (%))**: OCR 解码精度，即模型从图像中恢复出的文本与原文相比的准确率。范围从 0% 到 100%。
- **图中有四条线**:
    - 两条蓝色线代表使用 54 个视觉令牌（vis toks）的性能，分别对应左右两侧的 Y 轴（但图中两侧 Y 轴刻度相同）。
    - 两条红色线代表使用 100 个视觉令牌的性能，同样对应左右两侧的 Y 轴。
    - 实线代表 DeepSeek-OCR (Large) 模型，虚线代表 DeepSeek-OCR (Tiny) 模型。
- **关键数据点**: 在压缩比为 10 倍时，模型的精度大约在 90% 到 97% 之间。当压缩比达到 15 倍时，精度下降到约 70% 到 80%。在 20 倍压缩比时，精度依然保持在 60% 左右。

图 1(b) Omnidocbench 上的性能对比

这是一个散点图，比较了 DeepSeek-OCR 与其他模型在 OmniDocBench 基准测试上的性能和效率。

- **X 轴 (Text Tokens in Per Page (Ground-truth))**: 页面中的文本令牌数（基准真相），代表了文档的复杂程度。范围从 700-800 到 1100-1200。
- **Y 轴 (Overall Performance (Edit Distance))**: 综合性能，以编辑距离（Edit Distance, ED）来衡量，这个值越低表示性能越好。范围从 0.1 到 0.5。
- **点的大小 (Average Vision Tokens per Image)**: 每个点的大小代表该模型处理每张图片平均使用的视觉令牌数。点越小，表示模型效率越高。
- **图中的点**:
    - DeepSeek-OCR (Tiny) 和 DeepSeek-OCR (Large) 的点非常小，位于图的左下角，表明它们在取得极低编辑距离（高性能）的同时，使用了非常少的视觉令牌（高效率）。
    - 其他模型如 OLMOCR, VL-T5B, GOT-OCR2.0 等，它们的点普遍更大，且位置更高，意味着它们需要更多的视觉令牌，并且性能（编辑距离）还不如 DeepSeek-OCR。
    - 图中还用不同颜色区分了不同系列的编码器，突显了 DeepEncoder 系列（DeepSeek-OCR 使用的）的高效性。

原文翻译 图 1 | 图 (a) 展示了在 Fox 基准测试上的压缩比（基准真相中的文本令牌数/模型使用的视觉令牌数）测试；图 (b) 展示了在 OmniDocBench 上的性能比较。DeepSeek-OCR 可以在端到端模型中以最少的视觉令牌实现最先进的性能。

深度解读 这两张图是整篇论文核心论点的视觉呈现，它们共同讲述了一个关于“效率与性能”的精彩故事。

图 1(a) 是这项研究的“科学可行性证明”。它直接回答了那个核心问题：“光学压缩”这个想法究竟能做到什么程度？你可以看到，当压缩比在 10 倍以下时，精度曲线非常平缓且接近 100%，这意味着模型几乎可以无损地将图像还原为文字。这就像你用一个高质量的压缩软件（如 ZIP）压缩文件，解压后文件内容完好无损。而当压缩比超过 10 倍，精度开始下降，这就像你把一张图片压缩成一个非常小的 JPG 文件，图像质量会开始变差，一些细节会丢失。但即使在 20 倍这样极端的压缩比下，模型依然能“猜”对 60% 的内容，这本身已经非常了不起了。这个图表为我们提供了一个关键的经验法则：在 10 倍压缩比以内，光学压缩是一种近乎无损的高效信息编码方式。

图 1(b) 则是这项研究的“工程价值证明”。如果说图(a)是在实验室里验证一个物理定律，那么图(b)就是用这个定律造出了一台比市面上所有机器都更牛的发动机。在这个图里，每个模型都是一个“选手”，它们的目标是跑到左下角——即用最少的能量（最小的点，代表视觉令牌少）取得最好的成绩（最低的 Y 轴位置，代表编辑距离小）。你可以清晰地看到，DeepSeek-OCR 的两个模型（Tiny 和 Large）就像是奥运冠军，它们的点又小又靠下，完胜了其他所有“选手”。比如，它与 GOT-OCR2.0 相比，性能更好（Y 轴更低），同时视觉令牌数也少得多（点更小）。这雄辩地证明了 DeepSeek-OCR 的设计不仅仅是一个理论上的创新，更是一个在实际应用中兼具顶尖性能和极致效率的强大工具。它告诉我们，通过巧妙的架构设计，我们可以用更少的计算资源做更多、更好的事。

目录

原文翻译

深度解读 这份目录清晰地展示了科学论文的标准结构，就像一栋建筑的蓝图，让你能预先了解整篇文章的框架和逻辑流程。对于高中生来说，理解这个结构本身就是一次很好的科学方法论学习。

- **引言 (Introduction)**：这是故事的开端。作者会在这里提出一个重要的问题（LLM 处理长文本效率低下），并给出他们大胆的设想（用光学压缩来解决）。这里会阐述研究的动机和核心贡献。
- **相关工作 (Related Works)**：在提出自己的解决方案之前，必须先告诉大家前人都做了什么，以及他们的工作有什么不足之处。这既是对前人研究的尊重，也是为了凸显自己工作的创新性和必要性。这部分就像是在说：“看，这是目前大家普遍在用的几种方法，但它们各有各的毛病，所以我们需要一个新东西。”

- **方法论 (Methodology)**：这是论文的核心部分，相当于建筑蓝图中最详细的设计图。作者会在这里详细介绍他们是如何构建 DeepSeek-OCR 的，包括它的整体架构（Architecture）、关键的“眼睛”——DeepEncoder，以及聪明的“大脑”——MoE 解码器。此外，还会介绍他们用了哪些“教材”（数据引擎）以及如何“教”这个模型（训练流程）。

- **评估 (Evaluation)**：设计图再好，也要看造出来的房子牢不牢固。这一部分就是“工程验收”。作者会通过一系列严格的实验和数据来证明他们的方法是有效的。他们会用定量的数据（比如压缩比和准确率）和定性的例子（比如实际的识别效果图）来展示模型的性能。

- **讨论 (Discussion)**：在展示了所有成果之后，作者会站得更高一些，思考这项研究的深远意义。他们会探讨这个“光学压缩”的想法还能用在什么地方，比如模拟人类的记忆和遗忘。这部分是思想火花最闪耀的地方，展示了研究的想象力和未来潜力。

- **结论 (Conclusion)**：最后，对全文进行总结，重申研究的主要发现和贡献，并指出未来的研究方向。这是一个简洁有力的收尾。

通过这个目录，你可以看到一篇优秀的科研论文是如何层层递进、逻辑严密地提出问题、分析问题、解决问题，并最终展望未来的。

1. 引言

原文翻译 当前的大型语言模型（LLM）在处理长文本内容时面临着巨大的计算挑战，因为其计算复杂度与序列长度成二次方关系。我们探索了一个潜在的解决方案：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单一图像可以用比等效数字文本少得多的令牌来表示丰富的信息，这表明通过视觉令牌进行光学压缩可以实现高得多的压缩比。这一洞见促使我们从以 LLM 为中心的视角重新审视视觉-语言模型（VLM），重点关注视觉编码器如何能提升 LLM 处理文本信息的效率，而不仅仅是执行人类擅长的基本视觉问答（VQA）任务 。OCR 任务作为连接视觉和语言的中间模态，为这种视觉-文本压缩范式提供了一个理想的试验平台，因为它在视觉和文本表示之间建立了一种自然的压缩-解压缩映射，同时提供了可量化的评估指标。

深度解读 这一段是整篇论文的“思想内核”，它解释了这项研究的出发点和核心洞见。让我们来深入剖析一下。

首先，作者指出了一个当前 AI 领域非常“痛”的痛点：“二次方缩放”（quadratic scaling）。这是一个非常关键的概念。你可以用一个简单的比喻来理解：假设你在一个派对上，每个人都要和在场的其他所有人握手。如果派对上有 10 个人，大概需要 45 次握手。但如果人数增加到 100 人，握手次数会飙升到近 5000 次！人数只增加了 10 倍，工作量却增加了 100 多倍。LLM 处理长文本时面临的情况与此类似，文本每增加一点，模型需要考虑的词与词之间的关系数量就会爆炸式增长，这就是所谓的“二次方”关系。这使得处理一整本书或一份长篇报告变得极其困难和昂贵。

面对这个难题，作者提出了一个“降维打击”式的解决方案：“利用视觉模态作为文本信息的高效压缩媒介”。这背后的逻辑是，人类的信息处理系统本身就是多模态的。我们阅读文字时，眼睛看到的是图像，大脑再将其解码为语言和意义。一张A4纸的图像，在计算机里可能只是一个几百KB的文件，但它上面承载的文字信息如果转成纯文本，可能会大得多。作者敏锐地抓住了这一点，提出我们可以模仿这个过程，将长文本“渲染”成图像，让 AI 去“看”这张图，而不是“读”那段冗长的文字。

这个想法的巧妙之处在于，它改变了我们对视觉-语言模型（VLM）的传统认知。通常，我们认为 VLM 的任务是“看图说话”，比如描述一张猫的照片 。但作者说，我们应该从“LLM 为中心”的角度出发，把 VLM 的视觉能力当作一种工具，来解决 LLM 自身的效率问题。这就像你发现手机上的摄像头不仅能拍照，还能用来扫描二维码支付一样，是对一个工具功能的重新定义和升华。

最后，作者选择 OCR（光学字符识别）作为验证这个想法的“试验场” 。这是一个非常聪明的选择。因为 OCR 的任务本质就是从图像中恢复文本，这天然地构成了一个“压缩-解压缩”的闭环。我把文本“压缩”成图像，再用 OCR 模型把它“解压”回文本，然后比较解压后的文本和原文是否一致。这个过程的结果是完全可以量化的（比如识别准确率），从而可以非常客观、精确地衡量“光学压缩”的效果。这为整个研究奠定了一个坚实的科学基础。

原文翻译 因此，我们提出了 DeepSeek-OCR，一个作为高效视觉-文本压缩初步概念验证的 VLM。我们的工作主要有三个贡献：

首先，我们对视觉-文本令牌压缩比进行了全面的定量分析。我们的方法在 Fox 基准测试上，于 9-10 倍文本压缩比下实现了 96%+ 的 OCR 解码精度，在 10-12 倍压缩比下达到约 90%，在 20 倍压缩比下仍有约 60% 的精度（考虑到输出与基准真相之间的格式差异，实际准确率甚至更高），如图 1(a) 所示。结果表明，紧凑型语言模型可以有效地学习解码压缩后的视觉表示，这表明更大的 LLM 通过适当的预训练设计可以轻松获得类似的能力。

深度解读 在阐述了宏大的构想之后，作者在这里开始列举具体的贡献，这是科学写作的典型范式——先有想法，再有成果。第一个贡献是“用数据说话”，为“光学压缩”这个想法提供了坚实的量化证据。

这里的核心是“定量分析”。科学研究不仅仅是提出一个好点子，更重要的是要通过精确的测量和实验来证明这个点子是有效的。作者没有停留在“我觉得这个方法很好”的层面，而是通过在 Fox 基准测试（一个公认的文档理解能力测试集 ）上进行实验，给出了具体的数据：9-10 倍压缩下，精度 96%+。这个数字非常有说服力。它告诉我们，这种压缩方式在很大程度上是“无损”的，至少在 10 倍的范围内，信息损失极小。

这个发现的意义是双重的。从实践上看，它意味着我们可以放心地用这种方法来处理 10 倍长度以内的文本，从而大幅节省计算资源。从理论上看，它揭示了一个深刻的现象：一个相对较小的语言模型（DeepSeek-OCR 的解码器并不算巨大）就足以学会从高度压缩的视觉信号中解码出复杂的语言信息。这暗示了视觉和语言这两种模态之间存在着一种深刻的内在联系和可转换性。作者进一步推断，既然小模型都能学会，那么那些更强大的大型语言模型（LLM）通过针对性的训练，应该能更容易、更出色地掌握这项能力。这为该技术未来的发展和应用描绘了广阔的前景。

原文翻译 其次，我们引入了 DeepEncoder，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最少的视觉令牌。它通过一个 16 倍的卷积压缩器，将窗口注意力和全局注意力编码器组件串联起来。这种设计确保了窗口注意力组件处理大量的视觉令牌，而压缩器在这些令牌进入密集的全局注意力组件之前减少其数量，从而实现了有效的内存和令牌压缩。

深度解读 这是第二个贡献，也是技术层面最核心的创新：DeepEncoder 的架构设计。如果说“光学压缩”是战略思想，那么 DeepEncoder 就是实现这一战略的“王牌武器”。作者在这里解释了这件武器为什么如此强大和高效。

要理解这个设计的巧妙之处，我们需要先了解两个关键概念：“窗口注意力”（window attention）和“全局注意力”（global attention）。你可以这样想象：当你看一幅巨大的《清明上河图》时，你不可能一眼看清所有细节。你的眼睛会先聚焦在一个小区域，比如一座桥（这就是“窗口注意力”），仔细观察桥上的人物和活动。这种方式非常高效，因为你每次只处理一小块信息。然后，你的大脑会将这些局部信息整合起来，形成对整幅画的总体印象（这就是“全局注意力”）。全局注意力能让你理解画面的整体布局和故事，但它非常耗费“脑力”。

DeepEncoder 的设计就模拟了这个过程。它首先用一个擅长“窗口注意力”的组件（SAM ）来处理高分辨率的输入图像。这个组件就像我们的眼睛，高效地处理局部细节，即使图像很大，也不会消耗太多“内存”（激活内存）。然后，最关键的一步来了：一个“16 倍卷积压缩器”。这个压缩器就像一个信息过滤器和提炼器，它将 SAM 处理后的大量局部细节信息进行高度压缩和总结，变成数量少得多的、更抽象的视觉令牌。最后，这些被压缩过的、少量的视觉令牌才被送入一个擅长“全局注意力”的组件（CLIP ）。这个组件就像我们的大脑，对提炼后的信息进行深入的、全局的理解。

通过这种“先局部后全局、中间加压缩”的串联设计，DeepEncoder 完美地解决了效率和性能的矛盾。它既能处理高分辨率图像以捕捉精细细节（窗口注意力的功劳），又不会因为全局处理而导致内存爆炸（因为全局注意力只处理被压缩过的少量令牌）。这是一个极其聪明的、兼顾了计算效率和信息保真度的工程杰作。

原文翻译 第三，我们基于 DeepEncoder 和 DeepSeek3B-MOE 开发了 DeepSeek-OCR。如图 1(b) 所示，它在 OmniDocBench 上的端到端模型中以最少的视觉令牌实现了最先进的性能。此外，我们还为模型配备了解析图表、化学式、简单几何图形和自然图像的能力，以进一步增强其实用性。在生产环境中，使用 20 个节点（每个节点配备 8 个 A100-40G GPU），DeepSeek-OCR 每天可以为 LLM 或 VLM 生成 3300 万页的数据。

深度解读 这是第三个贡献，它强调了这项研究的“成果落地”和“实用价值”。作者不仅提出了一个理论，设计了一个架构，还最终打造出了一个完整、强大且可用的产品——DeepSeek-OCR。

首先，作者提到了模型的另一个关键组成部分：DeepSeek3B-MOE 解码器。这里的“MoE”是“Mixture of Experts”（专家混合）的缩写 。你可以把它理解为一个非常高效的团队协作模式。传统的 AI 模型就像一个全能但劳累的“通才”，所有任务都由他一人完成。而 MoE 模型则像一个拥有多位“专家”的团队，每位专家都擅长处理某一特定类型的问题。当一个任务来临时，一个“门控网络”（gating network）会智能地判断应该把任务分配给哪几位最相关的专家来处理 。这样做的好处是，模型总的参数量可以很大（团队规模大，能力强），但每次处理任务时只激活一小部分专家，因此计算效率非常高。DeepSeek-OCR 选用 MoE 解码器，进一步强化了其高效的特性。

其次，作者通过图 1(b) 的数据再次强调了模型的卓越性能——“用最少的视觉令牌实现了最先进的性能”。这呼应了引言的核心论点。更重要的是，他们还扩展了模型的能力，使其不只是一个“读书匠”，还能看懂图表、化学式等复杂内容。这极大地提升了模型的实用性，使其能够处理现实世界中各种复杂的文档。

最后，作者给出了一个惊人的生产力数据：“每天生成 3300 万页数据”。这个数字的意义在于，它表明 DeepSeek-OCR 不仅仅是一个实验室里的“玩具”，而是一个可以投入大规模工业化生产的“工具”。它可以成为一个强大的“数据工厂”，为训练其他更大型的 AI 模型（LLM/VLM）源源不

断地提供高质量、结构化的训练素材。这展示了这项研究从理论创新到技术实现，再到最终产生巨大应用价值的完整闭环。

原文翻译 总而言之，这项工作初步探索了使用视觉模态作为 LLM 中处理文本信息的高效压缩媒介。通过 DeepSeek-OCR，我们证明了视觉-文本压缩可以为不同历史上下文阶段实现显著的令牌减少（7-20 倍），为解决大型语言模型中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 令牌分配优化提供了经验指导，而提出的 DeepEncoder 架构则通过实际部署能力展示了其实际可行性。尽管本文以 OCR 作为概念验证，但这一范式为重新思考如何协同结合视觉和语言模态以增强大规模文本处理和智能体系统中的计算效率开辟了新的可能性。

深度解读 这是引言的总结段落，它再次拔高了研究的立意，并展望了其未来的广阔影响。

作者首先重申了研究的核心——“使用视觉模态作为高效压缩媒介”，并强调了 DeepSeek-OCR 作为“证明”的成功。7-20 倍的令牌减少是一个非常具体且令人印象深刻的成果，它直接指向了解决 LLM“长上下文”这一核心难题的潜力。

接着，作者指出了这项研究的两个层面的贡献。在“理论层面”，他们的定量分析（比如图 1a 和表 2 中的数据）为未来的研究者提供了宝贵的“经验指导”。这意味着，其他人如果想设计类似的 VLM，可以参考这篇论文的数据来决定如何最优化地分配视觉和文本令牌，以达到性能和效率的最佳平衡。在“实践层面”，他们设计的 DeepEncoder 架构被证明是“实际可行的”，并且具备大规模部署的能力，这为工业界应用该技术铺平了道路。

最后，作者强调，虽然他们选择 OCR 作为切入点，但其背后的思想——“光学上下文压缩”——具有更广泛的普适性。这不仅仅是关于识别文档，而是关于一种全新的信息处理范式。它启发我们去思考：在未来的 AI 系统中，视觉和语言不再是两种孤立的能力，而是可以相互转换、协同工作的统一体。例如，一个 AI 智能体（Agent）在执行复杂任务时，可以将它的思考过程和历史记录“压缩”成视觉记忆，从而更高效地进行长期规划和决策。这为构建更强大、更高效的 AI 系统开辟了全新的、令人兴奋的可能性。

2. 相关工作

图 2 | 流行 VLM 中的典型视觉编码器

图表描述 这张图展示了当前开源 VLM 中三种常见的视觉编码器架构，并指出了它们各自的缺陷。

- **左侧：Vary/DeepSeekVL (双塔并行结构)**
    - **图示**: 输入图像被同时送入两个并行的编码器：一个通用的 ViT (Vision Transformer) 和一个专门处理高分辨率细节的 ViTDet/SAM。两者的输出特征最终被送入 LLM。
    - **标注**: “pipeline parallel hard”（流水线并行困难），“pre-process twice”（需要两次预处理）。
    - **缺陷**: 这种架构需要对图像进行两次不同的预处理，部署复杂，并且在训练时进行流水线并行化很困难。

- **中间：InternVL series/DeepSeekVL2 (切片/瓦片化方法)**
    - **图示**: 一张大图被切割成多个小图块（tiles），每个图块分别通过一个 ViT 编码器，然后将所有图块的特征拼接起来送入 LLM。
    - **标注**: “too many vision tokens”（产生过多的视觉令牌），“overly fragmented”（过度碎片化）。
    - **缺陷**: 虽然能处理超高分辨率图像，但由于其基础编码器的原生分辨率通常较低，导致大图被切分得过于零碎，产生海量的视觉令牌，增加了 LLM 的处理负担。
- **右侧：Qwen2.5VL series (自适应分辨率方法)**
    - **图示**: 整个图像（无论大小）被分割成小补丁（patches），然后直接送入一个 ViT 编码器进行处理，最后送入 LLM。
    - **标注**: “large activations”（产生巨大的激活值），“need long sequence length”（训练时需要很长的序列长度）。
    - **缺陷**: 这种方法虽然灵活，但在处理大图像时会消耗巨大的显存（激活内存），容易导致 GPU 内存溢出。同时，为了训练，需要将不同分辨率的图像打包在一起，这要求训练序列非常长。

原文翻译 图 2 | 流行 VLM 中的典型视觉编码器。这里展示了当前开源 VLM 中常用的三种编码器，它们都存在各自的缺陷。

2.1. VLM 中的典型视觉编码器

原文翻译 当前开源的 VLM 主要采用三种类型的视觉编码器，如图 2 所示。第一种是以 Vary 为代表的双塔架构，它利用并行的 SAM 编码器来增加视觉词汇参数，以处理高分辨率图像。虽然这种方法提供了可控的参数和激活内存，但它存在显著的缺点：它需要双重图像预处理，这使得部署复杂化，并且在训练期间难以实现编码器的流水线并行。第二种是以 InternVL2.0 为代表的基于切片的方法，它通过将图像分割成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。尽管能够处理极高的分辨率，但这种方法有明显的局限性，因为其原生编码器分辨率通常较低（低于 512x512），导致大图像被过度分割，产生大量的视觉令牌。第三种是以 Qwen2-VL 为代表的自适应分辨率编码，它采用 NaViT 范式，通过基于补丁的分割直接处理完整图像，而无需进行图块并行化。虽然这种编码器可以灵活处理多种分辨率，但它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，并且序列打包在训练期间需要极长的序列长度。过长的视觉令牌会减慢推理的预填充和生成阶段。

深度解读 这一节是作者在动手设计自己的 DeepEncoder 之前，对现有技术方案的一次全面“市场调研”。在科学研究中，这一步至关重要，因为它确立了你工作的起点和创新的必要性。作者在这里分析了三种主流的 VLM 视觉编码器架构，并精准地指出了它们各自的“阿喀琉斯之踵”。

1. Vary 的双塔架构 ：你可以把它想象成一个“双专家会诊”系统。一个专家（如 ViT）负责看整体，另一个专家（如 SAM ）负责用放大镜看细节。虽然能兼顾全局和局部，但问题在于流程繁琐。病人（图像）需要先挂两个不同科室的号，做两次不同的检查（双重预处理），这让整个就诊流程（部署）变得复杂，而且两位专家很难协同工作（流水线并行困难）。

2. InternVL 的切片方法 ：这种方法简单粗暴，就像为了看清一幅巨大的壁画，直接把它切成无数张明信片大小的卡片，然后一张一张地看。好处是每张卡片都很容易处理，不会占用太多资源。但坏处是，卡片数量太多了（大量的视觉令牌），会把后面的处理单元（LLM）给淹没。而且，把壁画切得太碎，可能会丢失一些跨越卡片边界的整体图案信息（过度碎片化）。

3. Qwen2-VL 的自适应分辨率方法 ：这种方法像是拥有一个可以自由缩放的超高清摄像头，可以直接拍摄整幅壁画。好处是灵活，能适应不同大小的壁画。但问题是，当壁画非常巨大时，拍摄一张超高分辨率的照片会产生一个巨大的文件（巨大的激活内存），可能会直接把你的相机（GPU）给撑爆。而且，过多的视觉令牌（像素点）也会让后续的分析（推理）变得非常缓慢。

通过对这三种主流方案的批判性分析，作者为自己即将提出的 DeepEncoder 铺平了道路。他们的潜台词是：现有方案要么太复杂，要么太碎片化，要么太耗资源。我们需要一种新的、更优雅的解决方案，而这个方案，就是 DeepEncoder。这种“先破后立”的论证方式，是科学写作中非常有力的技巧。

2.2. 端到端 OCR 模型

原文翻译 OCR，特别是文档解析任务，一直是图像到文本领域中一个非常活跃的话题。随着 VLM 的发展，大量端到端的 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（需要分离的检测和识别专家模型）。Nougat 首次采用端到端框架对 arXiv 上的学术论文进行 OCR，展示了模型在处理密集感知任务方面的潜力。GOT-OCR2.0 将 OCR2.0 的范围扩展到包括更多合成图像解析任务，并设计了一款在性能和效率之间进行权衡的 OCR 模型，进一步凸显了端到端 OCR 研究的潜力。此外，像 Qwen-VL 系列 、InternVL 系列 以及它们的许多衍生模型等通用视觉模型，也在不断增强其文档 OCR 能力，以探索密集的视觉感知边界。然而，当前模型尚未解决一个关键的研究问题：对于一篇包含 1000 个单词的文档，至少需要多少个视觉令牌才能进行解码？这个问题对于研究“一图胜千言”的原理具有重要意义。

深度解读 在分析了通用的 VLM 视觉编码器之后，作者将目光聚焦到了更具体的应用领域——OCR 模型。这一节的核心是阐述 OCR 技术的演进，并从中引出一个尚未被回答的、却至关重要的问题。

首先，作者解释了 OCR 技术的范式转变。传统的 OCR 系统就像一条工厂流水线 。第一道工序的工人（检测模型）负责在页面上把所有文字区域用框标出来；第二道工序的工人（识别模型）再去看这些框里的内容，识别出具体是什么字。这种方式环节多，且容易出错。而现代的“端到端”（End-to-end）模型，则像一个全能的工匠，自己一个人就能完成从看到页面到输出全部文字的整个过程。这种方式更简洁、更强大，也是当前 VLM 发展的主流方向。作者提到了几个代表性的模型，如 Nougat 和 GOT-OCR2.0 ，来证明端到端 OCR 已经取得了显著的成功。

然而，在肯定了现有工作的成就之后，作者话锋一转，提出了一个极具哲学意味和科学价值的问题：“对于一篇包含 1000 个单词的文档，至少需要多少个视觉令牌才能进行解码？” 这个问题看似简单，却直击了信息论和人工智能的核心。我们常说“一图胜千言”，但这更多是一种文学性的描述。作者想做的，是把这个模糊的概念进行“科学量化”。到底一张图“等价于”多少个词？这个等价交换的“汇率”是多少？

这个问题之所以重要，是因为它关系到我们如何设计最高效的 AI 模型。如果我们知道解码 1000 个单词只需要 100 个视觉令牌，那么我们就不应该设计一个会产生 2000 个视觉令牌的臃肿模型。通过提出这个问题，作者巧妙地将自己的研究定位在了探索视觉和语言之间信息编码效率的“理论边界”上，这使得他们的工作不仅仅是对现有 OCR 模型的改进，更是对一个基础科学问题的探索。

3. 方法论

3.1. 架构

图 3 | DeepSeek-OCR 的架构

图表描述 这张图展示了 DeepSeek-OCR 模型的整体架构流程，从输入到输出，清晰地标示了各个组件及其关系。

- **左侧 (Input)**: 输入是一张包含文本的图像。
- **中间 (DeepEncoder)**: 这是模型的核心编码部分，负责将图像转换为视觉令牌。它由三个串联的部分组成：
    1. **SAM (Segment Anything Model)**: 图像首先被分割成 16x16 的补丁（patches），然后送入 SAM 编码器。SAM 主要由“局部注意力”（local attention）或窗口注意力构成，负责处理高分辨率的感知信息。这部分的参数量较小（80M），产生的激活值较低（low activation）。
    2. **Conv 16x down-sample (卷积 16 倍下采样)**: SAM 输出的大量视觉令牌（vision tokens）经过一个卷积模块，数量被压缩为原来的 1/16。这就像一个信息漏斗。
    3. **CLIP ViT 300M**: 压缩后的少量视觉令牌被送入 CLIP 视觉变换器。CLIP 主要由“全局注意力”（global attention）构成，负责理解知识和语义。
- **右侧 (Decoder)**: 这是模型的解码部分，负责生成最终的文本。
    1. **Prompt**: 用户的指令（例如“请识别图中的文字”）通过 Tokenizer 和 Embedding layer 转换为文本嵌入。
    2. **DeepSeek-3B (MOE-A570M)**: 经过 DeepEncoder 压缩后的视觉令牌和用户的文本提示嵌入一起被送入 MoE (Mixture of Experts) 解码器。
    3. **Output**: 解码器最终生成识别出的文本。

- **整体流程**: 图像 → SAM (局部感知) → 压缩器 (信息提炼) → CLIP (全局理解) → [视觉令牌 + 文本提示] → MoE 解码器 → 输出文本。

原文翻译 如图 3 所示，DeepSeek-OCR 采用了一个统一的端到端 VLM 架构，由一个编码器和一个解码器组成。编码器（即 DeepEncoder）负责提取图像特征、进行标记化以及压缩视觉表示。解码器用于根据图像令牌和提示生成所需的结果。DeepEncoder 的参数量约为 3.8 亿，主要由一个 8000 万参数的 SAM-base 和一个 3 亿参数的 CLIP-large 串联而成。解码器采用了一个 30 亿参数的 MoE 架构，其中激活参数为 5.7 亿。在接下来的段落中，我们将深入探讨模型的各个组件、数据工程和训练技巧。

深度解读 这一节和图 3 共同揭示了 DeepSeek-OCR 的“五脏六腑”，也就是它的内部构造。理解这个架构是理解整篇论文技术核心的关键。

这个架构可以被看作一个高效的“信息处理流水线”。首先，输入是一张图片，这是原始的、未经处理的视觉信息。

然后，图片进入了这篇论文最大的创新点——**DeepEncoder**。你可以把 DeepEncoder 想象成一个高度专业化的“视觉分析部门”，这个部门内部有明确的分工：

- 第一站：SAM-base 。这是“感知专家”，它擅长处理最原始、最精细的视觉信号。就像人的视网膜一样，它负责接收高分辨率的图像，并进行初步的、局部的特征提取。它使用的“窗口注意力”机制确保了即使图像很大，计算开销也能被控制在合理范围内。

- **第二站：16 倍压缩器**。这是“信息提炼师”。感知专家 SAM 产生了大量琐碎的细节信息，如果全部交给大脑处理，会不堪重负。提炼师的工作就是把这些海量细节进行压缩和总结，提取出其中最关键的精华，把信息量减少到原来的 1/16。

- 第三站：CLIP-large 。这是“认知专家”，相当于大脑的理解中枢。它接收的是经过提炼后的、高度浓缩的视觉信息。它利用强大的“全局注意力”能力，将这些信息与它所拥有的海量知识进行关联，从而形成对图像内容的深刻理解。

经过 DeepEncoder 这个高效的视觉分析部门处理后，原始的图像就被转换成了一小组既包含了图像细节又蕴含了高级语义的“视觉令牌”。

最后，这些视觉令牌和用户的文字指令（Prompt）一起被送入**解码器（Decoder）**。解码器是 DeepSeek-3B-MoE，你可以把它看作是“语言表达部门”。它接收了视觉部门的分析报告（视觉令牌）和用户的要求（Prompt），然后利用其强大的语言能力，将这些信息组织成通顺、准确的文字，最终输出结果。

总的来说，这个架构清晰地体现了“分工与协作”的思想。SAM 负责感知，CLIP 负责认知，压缩器负责连接，MoE 解码器负责生成。每个部分各司其职，通过一个精心设计的流程串联起来，最终实现了高效而精准的视觉-文本转换。

3.2. DeepEncoder

原文翻译 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点的视觉编码器：1. 能够处理高分辨率；2. 在高分辨率下激活值低；3. 视觉令牌少；4. 支持多分辨率输入；5. 参数数量适中。然而，如第 2.1 节所述，当前开源的编码器无法完全满足所有这些条件。因此，我们自

己设计了一款新颖的视觉编码器，命名为 DeepEncoder。

深度解读 在正式介绍 DeepEncoder 的具体构造之前，作者先列出了一个“设计需求清单”。这个清单非常重要，因为它解释了 DeepEncoder 为何被设计成现在的样子。这就像建筑师在画图纸前，必须先明确客户的需求：房子要大，但造价要低，还要冬暖夏凉，并且能灵活改变房间布局。

让我们逐一解读这五个需求：

1. **能够处理高分辨率**：这是基础。因为文档图像通常分辨率很高，如果模型看不清小字，OCR 的准确性就无从谈起。
2. **在高分辨率下激活值低**：这是对“效率”的要求。“激活值”可以通俗地理解为模型在计算过程中产生的中间数据量。激活值越低，占用的显存就越少，模型运行起来就越“省钱”。这个需求是为了避免像 Qwen2-VL 那样在处理大图时发生“内存溢出”的问题 。
3. **视觉令牌少**：这同样是对“效率”的要求。视觉令牌的数量直接决定了后续语言模型的工作量。令牌越少，语言模型处理起来就越快。这个需求是为了避免像 InternVL 那样产生“令牌洪水”的问题 。
4. **支持多分辨率输入**：这是对“灵活性”的要求。现实世界中的文档大小不一，一个好的模型应该能像变焦镜头一样，灵活地处理各种尺寸的输入，而不是只能固定在一个分辨率上。
5. **参数数量适中**：这是对“经济性”的要求。参数量决定了模型的大小。一个参数量适中的模型更容易被部署和使用，对硬件的要求也更低。

作者明确指出，在他们进行研究时，市面上没有一个开源的编码器能同时满足这五个“苛刻”的要求。这充分论证了他们自己动手设计 DeepEncoder 的必要性和创新性。这个清单不仅是 DeepEncoder 的设计准则，也是我们后续评判它是否成功的标准。

3.2.1. DeepEncoder 的架构

原文翻译 DeepEncoder 主要由两个组件构成：一个以窗口注意力为主的视觉感知特征提取组件，和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（补丁大小为 16）和 CLIP-large 作为这两个组件的主要架构。对于 CLIP，我们移除了其第一个补丁嵌入层，因为它的输入不再是图像，而是来自前一个流程的输出令牌。在这两个组件之间，我们借鉴了 Vary 的做法，使用一个 2 层的卷积模块对视觉令牌进行 16 倍下采样。每个卷积层核大小为 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 会将其分割成 1024/16×1024/16=4096 个补丁令牌。由于编码器的前半部分以窗口注意力为主且只有 8000 万参数，其激活值是可以接受的。在进入全局注意力之前，这 4096 个令牌经过压缩模块，令牌数量变为 4096/16=256，从而使得整体的激活内存变得可控。

深度解读 这一段详细阐述了 DeepEncoder 的内部构造，揭示了它是如何通过巧妙的“组件拼接”和“流程设计”来满足上一节提出的五个核心需求的。

这里的核心思想是“分而治之”和“强强联合”。作者没有从零开始发明全新的网络结构，而是聪明地选取了两个在各自领域已经证明非常强大的“预训练模型”——SAM 和 CLIP——并将它们串联起来。

- **视觉感知组件 (SAM-base)**：SAM 被誉为“分割一切”的模型，它对图像的局部细节和轮廓有着极强的感知能力。作者利用了它的这一特性，并特别强调了它主要使用“窗口注意力”。这意味着 SAM 在处理高分辨率图像时，是分区域、一小块一小块地看，计算效率很高，内存占用小。这直接满足了“处理高分辨率”和“低激活值”的需求。

- **视觉知识组件 (CLIP-large)**：CLIP 则擅长将视觉概念和语言知识联系起来，它能理解图像的“含义”。它主要使用“全局注意力”，即同时考虑所有信息来做出判断，这对于理解上下文至关重要。但全局注意力的计算成本非常高。

这里的点睛之笔，就是连接这两个组件的**16 倍卷积压缩器**。它扮演了一个至关重要的“瓶颈”角色。作者用一个具体的例子来说明：一张 1024×1024 的图像，经过 SAM 初步处理后，会产生 4096 个视觉令牌。这是一个非常庞大的数量。如果直接将这 4096 个令牌送入需要进行全局注意力的 CLIP，计算量将是灾难性的。但是，通过这个压缩器，令牌数量被急剧减少到 256 个。这意味着，计算成本最高的全局注意力部分，只需要处理极少量的、经过高度提炼的信息。

这个设计完美地体现了工程上的权衡艺术：让廉价的计算单元（SAM 的窗口注意力）处理繁重、琐碎的原始数据，然后通过一个高效的压缩环节，只让昂贵的计算单元（CLIP 的全局注意力）处理最精华、最核心的信息。这不仅满足了“视觉令牌少”和“低激活值”的需求，也通过复用强大的预训练模型，保证了整个系统的性能。

3.2.2. 多分辨率支持

图 4 | 多分辨率模式配置

图表描述 这张图通过示意图的方式，展示了 DeepSeek-OCR 为支持不同分辨率输入而设计的三种主要处理模式。

- **顶部：Tiny 和 Small 模式**
    - **图示**: 一张任意宽高比的输入图像（例如 W:512,H:840 或 W:1024,H:640）被直接“缩放”（Resize）成一个正方形（如 1024×1024 或 1280×1280）。
    - **对应模式**: Tiny, Small。
    - **对应令牌数**: 64, 100。
    - **特点**: 简单直接，但会改变图像的原始长宽比，可能导致图像内容变形。适用于对精度要求不高或分辨率较低的场景。

- **中部：Base 和 Large 模式**
    - **图示**: 一张任意宽高比的输入图像被放置在一个更大的正方形画布中央，周围的空白区域用“填充”（Padding）的方式补齐，以形成一个标准尺寸的正方形图像。
    - **对应模式**: Base, Large。
    - **对应令牌数**: 256, 400。
    - **有效令牌计算**: 图中给出了一个公式 R=1−(H−W)/W（此处应为 1−|H−W|/max(H,W) 的简化示意），表示有效令牌的比例。
    - **特点**: 保持了原始图像的长宽比，避免了内容变形，保证了识别精度。但会产生一些无效的“填充令牌”。
- **底部：Gundam 和 Gundam (Master) 模式**
    - **图示**: 对于超高分辨率的图像，采用“切片+填充”的混合策略。图像首先被切割成 n 个小图块（tiles），每个图块被处理成一个标准尺寸（如 640×640）。同时，整张图像的缩略图（全局视图）也被处理成一个标准尺寸（如 1024×1024）。
    - **对应模式**: Gundam, Gundam (Master)。
    - **对应令牌数**: n×(100或256)+(256或400)。
    - **特点**: 结合了切片和填充的优点，既能处理超大图像，又能兼顾局部细节和全局信息。这是一种最高性能的模式，适用于报纸等极其复杂的文档。

表格 1 | DeepEncoder 的多分辨率支持

表格内容 为了研究和应用目的，我们为 DeepEncoder 设计了多样的原生分辨率和动态分辨率模式。

模式	Tiny	Small	Base	Large	Gundam	Gundam-M
分辨率	512	640	1024	1280	640+1024	1024+1280
令牌数	64	100	256	400	n×100+256	n×256+400
处理方式	resize	resize	padding	padding	resize + padding	resize + padding

原文翻译 假设我们有一张包含 1000 个光学字符的图像，我们想测试解码需要多少视觉令牌。这就要求模型支持可变数量的视觉令牌。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，并设计了多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 主要支持两种大的输入模式：原生分辨率和动态分辨率。每种模式又包含多个子模式。

原生分辨率支持四种字模式：Tiny、Small、Base 和 Large，对应的分辨率和令牌数分别为 512x512 (64)、640x640 (100)、1024x1024 (256) 和 1280×1280 (400)。由于 Tiny 和 Small 模式的分辨率相对较小，为了避免浪费视觉令牌，图像通过直接缩放原始形状来处理。对于 Base 和 Large 模式，为了保留原始图像的长宽比，图像被填充到相应的大小。填充后，有效视觉令牌的数量小于实际视觉令牌的数量，计算公式为： $N_{valid}=\lceil N_{actual}\times[1-((\max(w,h)-\min(w,h))/(\max(w,h)))]\rceil$ (1) 其中 w 和 h 分别代表原始输入图像的宽度和高度。

深度解读 这一部分展示了 DeepSeek-OCR 在工程设计上的“灵活性”和“实用性”。作者们深知，在现实世界中，AI 模型需要面对各种各样的情况，因此一个“一刀切”的解决方案是行不通的。他们通过设计多种分辨率模式，赋予了模型应对不同任务和资源限制的能力，就像给一辆车配备了不同的档位。

- **Tiny 和 Small 模式（低档位）**：这两种模式采用“缩放”（resize）的方式处理图像。这种方式最简单快捷，就像你为了快速发送一张照片而把它缩小一样。代价是照片的长宽比可能会被改变，导致图像内容有些变形。这适用于那些对精度要求不高，或者文档本身比较简单的场景，追求的是极致的速度和最低的资源消耗。产生的视觉令牌数最少（64 或 100）。

- **Base 和 Large 模式（中高档位）**：这两种模式采用“填充”（padding）的方式。为了不让图像变形，他们会把原始图像放在一个更大的黑色背景板上，凑成一个正方形。这样做的好处是完整地保留了图像的原始信息，保证了识别的准确性。代价是会产生一些无用的“黑色背景板”令牌。作者还贴心地给出了公式(1)，用于计算真正承载图像信息的“有效令牌”数量。这体现了他们对效率精确控制的追求。

公式 (1) 解读: $N_{valid}=\lceil N_{actual}\times[1-((\max(w,h)-\min(w,h))/(\max(w,h)))]\rceil$ 这个公式看起来复杂，其实原理很简单。让我们拆解一下：

- $\max(w,h)$ 是图像的宽和高中较长的那条边。

- $\min(w,h)$ 是较短的那条边。

- $(\max(w,h)-\min(w,h))$ 就是长边和短边的差。

- $((\max(w,h)-\min(w,h))/(\max(w,h)))$ 计算的是“空白区域”占整个正方形背景板的比例。例如，一张 800×1000 的图片，这个比例就是 (1000−800)/1000=0.2，即 20% 的区域是填充的空白。

- 1−(空白比例) 就是图像本身所占的“有效区域”比例，即 80%。

- 最后用总令牌数 N_{actual} 乘以这个有效比例，再向上取整（由符号 $\lceil\rceil$ 表示），就得到了有效令牌数 N_{valid}。

这个公式的存在，表明了研究者对模型效率的精细化考量，他们不仅关心总共用了多少令牌，更关心“好钢用在了刀刃上”的有效令牌有多少。

原文翻译 动态分辨率可以由两种原生分辨率组成。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。切片方法遵循 InternVL2.0 的做法。支持动态分辨率主要是出于应用考虑，特别是对于超高分辨率的输入（如报纸图像）。切片是二次窗口

注意的一种形式，可以有效地进一步减少激活内存。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分割（图块数量控制在 2 到 9 的范围内）。
DeepEncoder 在 Gundam 模式下输出的视觉令牌数为：n×100+256，其中 n 是图块的数量。对于宽高都小于 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训练，以实现一个模型支持多种分辨率的目标。请注意，Gundam-master 模式（1024×1024 局部视图 + 1280×1280 全局视图）是在一个已训练好的 DeepSeek-OCR 模型上继续训练得到的。这主要是为了负载均衡，因为 Gundam-master 的分辨率太大，一起训练会拖慢整体训练速度。

深度解读

> **Gundam 和 Gundam-M 模式（终极档位）**：这个模式的名字“Gundam”（高达）非常形象，暗示了这是模型的最强形态。它专为处理像报纸这样信息密度极高、分辨率极大的“硬骨头”而设计。这种模式采取了“全局+局部”的策略，这是一种非常符合人类视觉习惯的方式。当我们看一张复杂的报纸时，我们既会有一个整体的版面布局印象（全局视图），也会聚焦于某一篇具体的文章或图片进行精读（局部视图）。Gundam 模式就是这样做的：它把报纸切成几个小块（局部视图）进行精细处理，同时保留一张整页的缩略图（全局视图），确保既不会丢失细节，也不会忽略整体布局。

这种设计借鉴了 InternVL 的切片思想，但又做出了关键的改进。因为 DeepSeek-OCR 的基础分辨率（如 640x640）本身就比很多模型要大，所以它不需要把图像切得那么碎（图块数量控制在 2-9 个），从而避免了“过度碎片化”的问题。最终产生的令牌数是所有局部视图令牌和全局视图令牌的总和 (n×100+256)。

一个特别体现工程智慧的细节是，Gundam-master 模式是“后期加练”出来的。作者解释说，这是为了“负载均衡”，因为这个模式太耗费资源，如果一开始就和其他模式一起训练，会拖慢整个训练进度。这就像在一个班级里，老师会先让所有学生掌握基础知识，然后再对几个尖子生进行拔高训练，而不是让全班同学都陪着尖子生从一开始就啃难题。这种务实的训练策略，确保了模型整体开发的高效性。

总而言之，通过这套精心设计的多分辨率支持系统，DeepSeek-OCR 成为了一个既能“粗茶淡饭”又能“饕餮盛宴”的多面手，可以根据不同的任务需求，灵活地在效率和精度之间做出最佳的权衡。

3.3. MoE 解码器

原文翻译 我们的解码器使用了 DeepSeekMoE ，具体是 DeepSeek-3B-MoE。在推理过程中，模型会激活 64 个路由专家中的 6 个和 2 个共享专家，激活参数约为 5.7 亿。30 亿参数的 DeepSeekMoE 非常适合以领域为中心（对我们来说是 OCR）的 VLM 研究，因为它在获得 30 亿模型表达能力的同时，也享受着 5 亿小模型的推理效率。

解码器从 DeepEncoder 的压缩潜在视觉令牌中重建原始文本表示，如下所示： fdec:Rn×dlatent →RN×dbext ; X^=fdec(Z) where n≤N (2) 其中 Z∈Rn×dlatent 是来自 DeepEncoder 的压缩潜在（视觉）令牌，而 X~∈RN×dtext 是重建的文本表示。函数 fdec 代表一个非线性映射，紧凑型语言模型可以通过 OCR 风格的训练有效地学习到它。可以合理地推测，LLM 通过专门的预训练优化，将能更自然地整合这类能力。

深度解读 这一节介绍了模型的“大脑和嘴巴”——MoE 解码器。如果说 DeepEncoder 负责“看”和“理解”，那么解码器就负责把理解到的内容“说”出来。

作者选择 DeepSeek-3B-MoE 作为解码器，这是一个非常明智的决定，因为它完美契合了整个项目对“效率”的追求。这里的“MoE”（Mixture of Experts，专家混合）是近年来在大型模型领域非常流行的一种架构 。你可以把它想象成一个大型咨询公司，而不是一个无所不知的超级顾问。这家公司有 64 位不同领域的专家（路由专家）。当客户（输入数据）提出一个问题时，公司的“前台经理”（门控网络）会迅速判断这个问题属于哪个领域，然后只把电话接通给最相关的 6 位专家。同时，还有 2 位“全能顾问”（共享专家），他们处理一些通用性的问题，所有电话都会经过他们。

这样做的好处显而易见：公司总共有 66 位专家，知识储备非常雄厚（对应模型总参数量 30 亿），但每次解决问题只需要 8 位专家出马（对应激活参数量 5.7 亿）。这样既保证了解决问题的“能力上限”（表达能力强），又极大地降低了每次服务的“成本”（推理效率高）。对于 OCR 这样一个相对垂直的领域，这种架构尤其适用。

公式 (2) 解读: $f_{dec}: \mathbb{R}^{n \times d_{latent}} \rightarrow \mathbb{R}^{N \times d_{text}}$; $X\tilde{} = f_{dec}(Z)$ 其中 $n \leq N$ 这个公式是对解码过程的数学化描述，它揭示了“光学压缩”的本质。让我们来翻译一下这个“数学语言”：

- Z：代表从 DeepEncoder 输出的视觉令牌。它是一个矩阵，有 n 行（代表有 n 个视觉令牌），d_{latent} 列（代表每个令牌的维度或信息含量）。
- $X\tilde{}$：代表解码器最终生成的文本令牌。它也是一个矩阵，有 N 行（代表有 N 个文本令牌），d_{text} 列。
- f_{dec}：代表解码器这个函数。它的作用就是进行一个神奇的“变换”。
- $n \leq N$：这是整个公式的灵魂。它明确地表示，输入的视觉令牌数量 n 是小于或等于输出的文本令牌数量 N 的。

整个公式连起来的意思就是：解码器 f_{dec} 接收一小部分（n 个）高度浓缩的视觉令牌 Z，通过一系列复杂的非线性计算，将它们“解压缩”并“翻译”成一大篇（N 个）文本令牌 $X\tilde{}$。这正是“一图胜千言”的数学表达。整篇论文的实验，都是为了证明，AI 模型通过大量的 OCR 式训练，完全可以学会这个神奇的 f_{dec} 函数。

3.4. 数据引擎

原文翻译 我们为 DeepSeek-OCR 构建了复杂多样的训练数据，包括主要由传统 OCR 任务（如场景图像 OCR 和文档 OCR）组成的 OCR 1.0 数据；主要包括复杂人造图像解析任务（如常见图表、化学式和平面几何解析数据）的 OCR 2.0 数据；以及主要用于向 DeepSeek-OCR 注入一定的通用图像理解能力并保留通用视觉接口的通用视觉数据。

深度解读 这一节介绍了 AI 模型的“食粮”——训练数据。在人工智能领域，有一句名言：“数据和算法同等重要”。一个再优秀的模型架构，如果没有高质量、大规模、多样化的数据进行训练，也只是一个空壳。本节展示了研究者们为了“喂养”出强大的 DeepSeek-OCR，建立了一个多么庞大而精细的“数据厨房”。

他们将数据精心分成了三大类，就像为一个运动员制定营养均衡的食谱一样：

- **OCR 1.0 数据（主食）**：这是模型能力的基础，主要目标是让模型学会最基本的“认字”和“阅读”能力。这包括了各种文档和现实世界场景中的文字。这是保证模型 OCR 能力的基本盘。

- **OCR 2.0 数据（高蛋白营养品）**：这部分数据是为了提升模型的“高阶理解”能力。它不再是简单的认字，而是要理解那些超越普通文本的、结构化的视觉信息，比如图表中的数据关系、化学式的分子结构、几何图形的形状和位置。这让模型从一个“识字先生”向一个能读懂科学图表的“分析师”转变。

- **通用视觉数据（维生素和微量元素）**：这部分数据是为了防止模型“偏科”。如果只给模型看各种文档，它可能会变成一个书呆子，看到一张猫的照片都不知道是什么。通过加入一些通用的图像数据（比如风景、物体、动物等），可以保持模型的“常识”，让它知道自己是一个通用的视觉-语言模型，而不仅仅是一个 OCR 工具。这保留了模型的通用性，为未来的功能扩展留下了接口。

这种分层、分类的数据策略，体现了构建一个强大 AI 模型所需的系统性工程思维。通过精心搭配“食谱”，研究者们确保了 DeepSeek-OCR 不仅在核心任务上表现卓越，还具备了广泛的适应性和未来的发展潜力。

3.4.1. OCR 1.0 数据

图 5 | OCR 1.0 精细标注展示

图表描述 这张图展示了 OCR 1.0 数据的精细标注格式。

- **(a) Ground truth image (基准真相图像)**: 左侧是一张包含文本、表格和数学题的克罗地亚语文档页面。页面上有手写的标记，内容复杂。

- **(b) Fine annotations with layouts (带布局的精细标注)**: 右侧是与左图对应的标注文件内容。这不是简单的纯文本，而是一种结构化的、图文交错的格式。

    - **标签和坐标**: 每一段文本或每一个元素（如公式、表格）前面都有一对特殊的标签，如 `<ref>text</ref>` 或 `<ref>equation</ref>`。在标签内部，还有一个 `<det>` 标签，里面包含了该元素在原始图像中的精确坐标位置（例如 `[]`）。所有的坐标都被归一化到 1000 个区间内。

    - **内容**: 标签后面跟着该区域内的实际文本内容。

    - **结构化信息**: 对于表格，标注内容直接是 HTML 的 `<table>` 格式，如 `<table><tr><td>3</td><td></td><td></td><td>4</td></tr></table>`。对于公式，则标注为 LaTeX 格式，如 `\frac{7 \cdot 4}{2} = 14.`。

- **核心思想**: 这种标注方式不仅告诉模型“这里有什么字”，还告诉模型“这些字在图上的哪个位置”以及“它是一个标题、一段正文、一个表格还是一个公式”。这是一种非常丰富和精细的“监督信息”。

原文翻译 文档数据是 DeepSeek-OCR 的重中之重。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，其中中文和英文约占 2500 万，其他语言占 500 万。对于这些数据，我们创建了两种类型的基准真相：粗略标注和精细标注。粗略标注是直接使用 fitz 从完整数据集中提取的，旨在教会模型识别光学文本，特别是在少数族裔语言中。精细标注包括中文和英文各 200 万页，使用先进的布局模型（如 PP-DocLayout ）和 OCR 模型（如 MinuerU 和 GOT-OCR2.0 ）进行标注，以构建检测和识别交错的数据。对于少数族裔语言，在检测部分，我们发现布局模型具有一定的泛化能力。在识别部分，我们使用 fitz 创建小块数据来训练一个 GOT-OCR2.0，然后使用训练好的模型在布局处理后对小块进行标注，采用模型飞轮的方式创建了 60 万个数据样本。在训练 DeepSeek-OCR 期间，粗略标签和精细标签通过不同的提示来区分。精细标注的图文对基准真相可见图 5。我们还收集了 300 万份 Word 数据，通过直接提取内容构建了没有布局的高质量图文对。这部分数据主要对公式和 HTML 格式的表格带来了好处。此外，我们选择了一些开源数据 作为补充。

对于自然场景 OCR，我们的模型主要支持中文和英文。图像数据来源是 LAION 和 Wukong ，使用 PaddleOCR 进行标注，中英文各有 1000 万个数据样本。与文档 OCR 一样，自然场景 OCR 也可以通过提示来控制是否输出检测框。

深度解读 这一部分详细介绍了模型“主食”（OCR 1.0 数据）的具体构成。可以看出，研究者们在数据准备上投入了巨大的精力，其策略可以用“广度”、“深度”和“精细度”三个词来概括。

- **广度**：数据来源极其广泛。3000 万页 PDF，涵盖约 100 种语言，这确保了模型具有强大的跨语言、跨领域泛化能力。它不仅能处理常见的中文和英文文档，还能应对各种小语种，这在处理全球互联网数据时至关重要。此外，还包括了 2000 万的自然场景 OCR 数据（比如街景、招牌上的文字），让模型的能力不局限于规整的文档，也能应对现实世界中各种不规则的文本。
- **深度**：数据标注分为两个层次——“粗略标注”和“精细标注”。
    - **粗略标注**：就像是让学生进行大量的“泛读”。通过 `fitz` 这样的工具快速提取海量 PDF 中的纯文本。这种方法的优点是速度快、成本低，可以快速扩大模型的“词汇量”，特别是对于那些资源较少的小语种，这是教会模型“认字”的最快途径。
    - **精细标注**：这相当于让学生进行“精读”。如图 5 所示，研究者们利用其他先进的 AI 模型（如 PP-DocLayout, MinerU）作为“助教”，对 400 万页中英文文档进行了像素级的精细标注。这种标注不仅告诉模型文字内容，还告诉了它文字的**位置、类型（标题/正文/表格）和结构**。这种带有布局信息的训练，是模型能够实现高级文档解析能力的关键。
- **精细度**：这里体现了一个非常聪明的策略——“模型飞轮”（model flywheel）。对于小语种，没有现成的精细标注工具怎么办？他们先用 `fitz` 提取的小块文本训练出一个“初级”的 GOT-OCR2.0 模型，然后再用这个初级模型去标注更多的数据，再用这些新标注的数据去训练一个“中级”模型……如此循环往复，模型的能力就像滚雪球一样越来越强。这是一种高效的、自举的数据增强方法。

通过这种多层次、多来源、多语言的数据策略，研究者们为 DeepSeek-OCR 打下了坚实的基础，使其不仅“识字多”，而且“懂排版”，能够应对各种复杂场景。

3.4.2. OCR 2.0 数据

图 6 | 图表和几何图形的图文基准真相

图表描述 这张图展示了用于训练模型理解复杂结构化信息的 OCR 2.0 数据标注格式。

- **(a) Image-text ground truth of chart (图表的图文基准真相)**:
    - **左侧图像**: 一个标准的柱状图，包含标题、X/Y 轴标签、图例和数据条。
    - **右侧标注**: 图表的数据没有被标注成复杂的字典格式，而是直接转换成了一个简洁的 HTML 表格 (`<table>...</table>`)。表格的表头（`<th>`）对应图表的类别（Germany, France 等），表格的内容（`<td>`）则对应每个类别在不同年份（2024, 2025 等）的具体数值。
    - **核心思想**: 这种标注方式强迫模型去“理解”图表的内在逻辑，即将视觉上的柱子高度，转换成结构化的、可被机器读取的表格数据。使用 HTML 格式既保留了结构，又比其他格式更节省令牌。
- **(b) Image-text ground truth of geometry (几何图形的图文基准真相)**:
    - **左侧图像**: 一张包含多个点（A, B, C, D, E）和连接线段的平面几何图形。
    - **右侧标注**: 几何图形被转换成一个字典（Dictionary）格式。
        - **字典结构**: 字典中包含了多个键（key），如 "points"（点），"lines"（线段），"circles"（圆）等。
        - **内容**: "points" 键对应的值是一个列表，包含了每个点的标签（如 "A"）和其在坐标系中的精确坐标。 "lines" 键对应的值也是一个列表，包含了每条线段的两个端点（如 "AB"）和线段的类型（如 "solid" 表示实线）。
        - **编码方式**: 线段的编码遵循了 Slow Perception 的方式，这是一种结构化的描述方法。
    - **核心思想**: 这种标注方式将一个视觉上的几何图形，完全解构成了一个由点、线及其关系组成的逻辑结构。这要求模型不仅要看到图形，更要理解其拓扑结构和组成元素。

原文翻译 遵循 GOT-OCR2.0 的做法，我们将图表、化学式和平面几何解析数据称为 OCR 2.0 数据。对于图表数据，我们遵循 OneChart 的方法，使用 pyecharts 和 matplotlib 渲染了 1000 万张图像，主要包括常用的折线图、条形图、饼图和复合图。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。对于化学式，我们利用 PubChem 的 SMILES 格式作为数据源，并使用 RDKit 将它们渲染成图像，构建了 500 万个图文对。对于平面几何图像，我们遵循 Slow Perception 的方法进行生成。具体来说，我们使用大小为 4 的感知标尺来建模每条线段。为了增加渲染数据的多样性，我们引入了几何平移不变的数据增强，即同一个几何图像在原始图像中进行平移，对应于在坐标系中心位置绘制的相同基准真相。基于此，我们共构建了 100 万个平面几何解析数据，如图 6(b) 所示。

深度解读 这部分介绍了模型的“高蛋白营养品”——OCR 2.0 数据。这部分训练的重点不再是让模型“认字”，而是让它学会“理解逻辑和结构”。这是从感知到认知的关键一步。

- **图表解析**：研究者们生成了海量的图表图像，并将任务定义为“看图填表”。如图 6(a) 所示，模型需要将视觉上的柱状图，准确地转换成一个 HTML 表格。这要求模型必须理解图表的构成元素：标题、坐标轴、图例、以及每个数据点对应的数值。选择 HTML 表格作为输出格式非常聪明，因为它既是结构化的，又是文本格式，可以直接被语言模型处理，并且相比其他格式更节省空间。

- **化学式解析**：他们利用 SMILES 格式（一种用字符串表示分子结构的行业标准）和 RDKit（一个化学信息学工具包），构建了大量的“化学式图像-SMILES 字符串”数据对。这训练了模型识别化学键、原子和空间结构的能力，使其能够将复杂的二维分子结构图翻译成机器可读的线性字符串。

- **平面几何解析**：这是最具挑战性的任务之一。如图 6(b) 所示，模型需要将一个几何图形“解构”成一个包含点、线、坐标和连接关系的结构化字典。这要求模型具备初步的“空间推理”能力。为了让模型学得更扎实，他们还使用了一种巧妙的“数据增强”方法：将同一个几何图形在画布上随机移动位置，但标注的答案（以原点为中心的坐标）保持不变。这教会了模型一个重要的概念——“平移不变性”，即一个三角形无论画在纸的哪个角落，它依然是同一个三角形。

通过这 1600 万（1000+500+100）高质量的 OCR 2.0 数据“加餐”，DeepSeek-OCR 不再是一个只能处理文本的文科生，而是成长为一个能够理解图表、化学和几何的理科高手，极大地拓展了其在科学、技术、工程和数学（STEM）领域的应用潜力。

3.4.3. 通用视觉数据

原文翻译 DeepEncoder 可以受益于 CLIP 的预训练成果，并有足够的参数来融合通用的视觉知识。因此，我们也为 DeepSeek-OCR 准备了一些相应的数据。遵循 DeepSeek-VL2 的做法，我们为字幕、检测和定位等任务生成了相关数据。请注意，DeepSeek-OCR 并非一个通用的 VLM 模型，这部分数据仅占总数据的 20%。我们引入这类数据主要是为了保留通用视觉接口，以便对我们的模型和通用视觉任务感兴趣的研究人员将来可以方便地推进他们的工作。

深度解读 这部分介绍的是模型的“维生素和微量元素”——通用视觉数据。在进行了大量的专业 OCR 训练后，为了防止模型变成一个“偏科”的专家，研究者们给它补充了一些通用的视觉知识。

这里的核心思想是“保持通用性”和“预留扩展性”。DeepEncoder 的一个关键组件是 CLIP，而 CLIP 本身就是一个在海量图文对上预训练过的、具有强大通用视觉理解能力的模型 。研究者们不希望在 OCR 的专项训练中完全丢掉 CLIP 的这些宝贵能力。

因此，他们加入了 20% 的通用视觉数据，这些数据涵盖了诸如“看图说话”（caption）、“图中有什么”（detection）和“图中的‘苹果’在哪里”（grounding）等任务。这就像在专业体育训练之余，让运动员也进行一些基础的体能训练，以保持全面的身体素质。

作者明确指出，DeepSeek-OCR 的定位不是一个全能的通用 VLM，所以这部分数据量不大。其主要目的有两个：

1. **保留接口**：确保模型在看到一张普通的风景照时，不会不知所措，仍然能够进行基本的描述和理解。这保留了一个“通用视觉接口”。

2. **方便未来研究**：这为其他研究者提供了一个很好的起点。如果有人想在 DeepSeek-OCR 强大的文档处理能力之上，进一步增强其通用视觉能力，他们不需要从零开始，可以直接在这个保留的接口上继续进行开发和微调。

这种有远见的数据策略，体现了研究者们不仅关注当前任务的极致性能，也为模型的长期发展和社区的二次创新铺平了道路。

3.4.4. 纯文本数据

原文翻译 为了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，所有数据都处理成 8192 个令牌的长度，这也是 DeepSeek-OCR 的序列长度。总而言之，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%。

深度解读 这部分介绍的是模型的“语言基础课”——纯文本数据。一个视觉-语言模型，不仅“视觉”能力要强，“语言”能力同样是根基。如果一个模型能完美识别图像中的所有文字，但组织成的句子颠三倒四、不合逻辑，那它依然是一个失败的模型。

因此，研究者们专门加入了 10% 的纯文本数据进行训练。这部分训练与图像完全无关，目的就是为了锤炼解码器（DeepSeek-3B-MoE）自身的语言建模能力。这就像一个翻译家，不仅要看得懂外语（视觉理解），自身的母语功底（语言能力）也必须非常扎实，才能做出“信、达、雅”的翻译。

他们将文本数据处理成 8192 个令牌的长度，这与模型最终处理图文任务时的总序列长度保持一致。这样做可以确保模型在训练和推理时处理的序列长度是匹配的，有助于提升模型的稳定性和性能。

最后，作者给出了一个清晰的“数据配比”：70% 的专业课（OCR 数据），20% 的素质拓展课（通用视觉数据），以及 10% 的语文基础课（纯文本数据）。这个 7:2:1 的黄金配比，是他们经过精心设计，用以打造一个专业突出、基础扎实、且具备发展潜力的全能型文档处理模型的秘方。

3.5. 训练流程

原文翻译 我们的训练流程非常简单，主要包括两个阶段：a). 独立训练 DeepEncoder；b). 训练 DeepSeek-OCR。请注意，Gundam-master 模式是通过在预训练好的 DeepSeek-OCR 模型上使用 600 万采样数据继续训练得到的。由于其训练协议与其他模式相同，我们在此省略详细描述。

深度解读 这一节概述了模型的“培养计划”。一个强大的 AI 模型不是一蹴而就的，而是需要一个精心设计的、分阶段的训练过程。DeepSeek-OCR 的训练流程被简化为清晰的两步，体现了“先打基础，再搞整合”的逻辑。

- **第一阶段：独立训练 DeepEncoder（练“眼”）**：在将视觉编码器（DeepEncoder）和语言解码器（Decoder）组合在一起之前，研究者们先单独对 DeepEncoder 进行了专门的训练。这一步的目标是确保模型的“眼睛”本身就足够敏锐和强大。他们使用了所有的 OCR 数据和大量的通用视觉数据来“打磨”这只眼睛，让它学会如何从各种图像中高效地提取出有用的特征。这就像在组装一台电脑前，先对 CPU、显卡等核心部件进行单独的压力测试，确保它们各自的性能都达标。

- **第二阶段：训练 DeepSeek-OCR（“眼”和“脑”的协同训练）**：在 DeepEncoder 这个强大的“眼睛”准备好之后，再将它与 DeepSeek-3B-MoE 这个聪明的“大脑”连接起来，进行端到端的整体训练。在这一阶段，模型学习的是如何将视觉信息（由 DeepEncoder 提供）和语言信息（用户的提示）无缝地结合起来，并生成流畅、准确的文本输出。这是对整个系统协同工作能力的训练。

这种分阶段的训练策略有很多好处。首先，它降低了训练的复杂性，使得问题更容易被诊断和调试。其次，通过第一阶段的预训练，DeepEncoder 获得了一个非常好的初始化状态，这使得第二阶段的整体训练能够更快地收敛，也更容易达到一个更高的性能水平。这是一种在复杂 AI 系统工程中非常常见且高效的训练范式。

3.5.1. 训练 DeepEncoder

原文翻译 遵循 Vary 的做法，我们利用一个紧凑的语言模型 ，并使用下一个令牌预测的框架来训练 DeepEncoder。在这个阶段，我们使用了前面提到的所有 OCR 1.0 和 2.0 数据，以及从 LAION 数据集中采样的 1 亿通用数据。所有数据都以 1280 的批量大小训练了 2 个周期，使用 AdamW 优化器，余弦退火调度器 ，学习率为 5e-5。训练序列长度为 4096。

深度解读 这一节详细描述了如何进行第一阶段的训练，即如何“单独把眼睛练好”。他们采用了一种被称为“下一个令牌预测”（next token prediction）的训练框架，这是训练现代语言模型的标准方法。

你可以这样理解这个过程：研究者们将一张图片（例如，一张包含“hello world”的图片）输入到 DeepEncoder 中，得到一组视觉令牌。然后，他们将这些视觉令牌和一个紧凑的语言模型连接起来，并给语言模型看“hello”这两个词的文本，要求它预测下一个词应该是什么。正确的答案是“world”。如果模型预测错误，系统就会根据错误程度来调整 DeepEncoder 和这个紧-凑语言模型的参数，让它们下次能够做出更准确的预测。

通过在海量的图文数据上重复这个过程，DeepEncoder 被“逼迫”着去学习如何提取出对后续文本生成最有帮助的视觉特征。如果它提取出的视觉信息是模糊或错误的，那么后续的语言模型就永远无法准确地预测出下一个词。因此，这个训练过程的本质，就是训练 DeepEncoder 成为一个优秀的“图像信息编码员”，它必须学会将图像内容高效地编码成语言模型能够理解的“语言”。

文中的一些技术术语解释如下：

- **批量大小 (batch size) 1280**：每次训练迭代中，同时处理 1280 个数据样本，这可以提高训练效率。

- **周期 (epochs) 2**：将全部训练数据完整地过两遍。

- **AdamW 优化器**：一种先进的算法，用于智能地调整模型的参数以减少预测错误。

- **余弦退火调度器**：一种动态调整“学习率”（模型参数更新幅度）的策略。它让学习率像余弦曲线一样，在训练过程中先缓慢下降，然后快速下降，再缓慢下降，有助于模型找到更好的最终解。
- **学习率 (learning rate) 5e-5**：即 0.00005，这是一个控制模型参数每次更新幅度的超参数。
- **序列长度 (sequence length) 4096**：模型在一次处理中能够看到的最大令牌数量。

3.5.2. 训练 DeepSeek-OCR

原文翻译 在 DeepEncoder 准备好之后，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 平台上进行。整个模型使用流水线并行（PP），并被分为 4 个部分，DeepEncoder 占两部分，解码器占两部分。对于 DeepEncoder，我们将 SAM 和压缩器视为视觉标记器，将它们放在 PP0 中并冻结其参数，同时将 CLIP 部分视为输入嵌入层，并将其放在 PP1 中，权重不冻结进行训练。对于语言模型部分，由于 DeepSeek3B-MoE 有 12 层，我们将 6 层分别放在 PP2 和 PP3 上。我们使用 20 个节点（每个节点有 8 个 A100-40G GPU）进行训练，数据并行（DP）为 40，全局批量大小为 640。我们使用 AdamW 优化器，采用基于步数的调度器，初始学习率为 3e-5。对于纯文本数据，训练速度为每天 900 亿令牌，而对于多模态数据，训练速度为每天 700 亿令牌。

深度解读 这一节描述了第二阶段的训练，即如何将“眼睛”和“大脑”组装起来进行协同训练。这个过程涉及到了大规模 AI 模型训练中的一些高级工程技术。

- **流水线并行 (Pipeline Parallelism, PP)**：训练像 DeepSeek-OCR 这样的大模型，通常需要多张 GPU 显卡协同工作。流水线并行是一种高效的并行策略。你可以把它想象成一条汽车组装流水线。整个模型（汽车）被拆分成 4 个部分（工位）。第一张 GPU（工位 PP0）负责处理 SAM 和压缩器部分，完成后把结果传给第二张 GPU（工位 PP1）处理 CLIP 部分，接着传给第三张 GPU（工位 PP2）处理解码器的前一半，最后由第四张 GPU（工位 PP3）处理解码器的后一半并输出结果。这样，多张 GPU 可以像流水线一样同时处理不同的数据，极大地提高了训练效率。
- **参数冻结 (Freezing Parameters)**：在训练过程中，研究者们做了一个重要的决定：将 SAM 和压缩器部分的参数“冻结”，即不进行更新。这意味着他们认为在第一阶段的预训练中，这部分的“感知”能力已经训练得足够好了。在第二阶段，他们将训练的重点放在了 CLIP 部分和整个解码器上，主要目的是让 CLIP 学会更好地将视觉信息“转码”为解码器能理解的格式，并让解码器学会如何基于这些视觉信息来生成文本。这种“有重点”的训练策略，可以使训练过程更稳定，也更高效。
- **硬件规模和训练速度**：作者明确列出了他们使用的计算资源（20 个节点，160 张 A100-40G GPU），这是一个相当大的计算集群。他们还给出了具体的训练速度（每天处理 700-900 亿令牌）。公布这些信息，一方面展示了这项工作的工程复杂度和投入，另一方面也为其他研究者复现或评估这项工作提供了重要的参考。这体现了科学研究的透明性。

总的来说，这一节不仅展示了训练的具体参数，更揭示了其背后复杂而高效的分布式训练工程，这是将一个理论模型转化为现实可用产品的关键所在。

4. 评估

4.1. 视觉-文本压缩研究

表格 2 | 视觉-文本压缩比测试

表格内容 我们使用 Fox 基准测试中所有包含 600-1300 个令牌的英文文档来测试 DeepSeek-OCR 的视觉-文本压缩比。文本令牌数是使用 DeepSeek-OCR 的分词器对基准真相文本进行分词后的令牌数量。视觉令牌数=64 或 100 分别代表 DeepEncoder 将输入图像缩放到 512x512 和 640×640 后输出的视觉令牌数。

	视觉令牌数 = 64		视觉令牌数 = 100	
文本令牌数	精度	压缩比	精度	压缩比
600-700	96.5%	10.5x	98.5%	6.7x
700-800	93.8%	11.8x	97.3%	7.5x
800-900	83.8%	13.2x	96.8%	8.5x
900-1000	85.9%	15.1x	96.8%	9.7x
1000-1100	79.3%	16.5x	91.5%	10.6x
1100-1200	76.4%	17.7x	89.8%	11.3x
1200-1300	59.1%	19.7x	87.1%	12.6x

原文翻译 我们选择 Fox 基准测试来验证 DeepSeek-OCR 对富文本文档的压缩-解压缩能力，以初步探索上下文光学压缩的可行性和边界。我们使用 Fox 的英文文档部分，用 DeepSeek-OCR 的分词器（词汇量约 12.9 万）对基准真相文本进行分词，并选择包含 600-1300 个令牌的文档进行测试，恰好是 100 页。由于文本令牌数量不大，我们只需要测试 Tiny 和 Small 模式下的性能，其中 Tiny 模式对应 64 个令牌，Small 模式对应 100 个令牌。我们使用不带布局的提示："\nFree OCR." 来控制模型的输出格式。尽管如此，输出格式仍不能完全匹配 Fox 基准测试，因此实际性能会比测试结果略高一些。

如表 2 所示，在 10 倍压缩比内，模型的解码精度可以达到约 97%，这是一个非常有前景的结果。未来，通过文本到图像的方法，或许可以实现近 10 倍的无损上下文压缩。当压缩比超过 10 倍时，性能开始下降，这可能有两方面原因：一是长文档的布局变得更加复杂，另一个原因可能是长文本在 512x512 或 640×640 的分辨率下变得模糊。第一个问题可以通过将文本渲染到单一布局页面来解决，而我们认为第二个问题将成为遗忘机制的一个特征。当压缩令牌近 20 倍时，我们发现精度仍能接近 60%。这些结果表明，光学上下文压缩是一个非常有前景且值得研究的方向，并且这种方法不会带来任何额外开销，因为它可以利用 VLM 的基础设施，多模态系统天生就需要一个额外的视觉编码器。

深度解读 这一节和表格 2 是整篇论文的"高光时刻"，它用最直接、最有力的数据，验证了"上下文光学压缩"这个核心设想。

实验设计：研究者们选择了一个标准的文档理解测试集 Fox ，并筛选出文本长度在 600-1300 令牌之间的 100 页英文文档。这个长度范围很有代表性，大致相当于一到两页的普通 A4 纸内容。他们使用了模型最低效的两个模式（Tiny 和 Small），分别只产生 64 和 100 个视觉令牌。这是一种“压力测试”，意在探索模型在最极限的压缩条件下的性能边界。

核心发现（解读表格 2）：

- **10 倍压缩的“甜点区”**：观察表格数据，可以清晰地看到一个规律。当压缩比在 10 倍左右时（例如，用 100 个视觉令牌处理 900-1000 个文本令牌，压缩比 9.7x），模型的精度高达 96.8%。用 64 个视觉令牌处理 600-700 个文本令牌（压缩比 10.5x），精度也达到了 96.5%。这意味着，在 10 倍的压缩水平下，信息几乎是“无损”的。这为该技术的实际应用提供了一个非常可靠的性能保证。

- **性能的“优雅下降”**：当压缩比继续增大，超过 10 倍甚至达到近 20 倍时，性能开始下降。例如，用 64 个视觉令牌处理 1200-1300 个文本令牌（压缩比 19.7x），精度降至 59.1%。然而，这并非失败。恰恰相反，它展示了这种压缩方式的一个重要特性：信息不是突然消失的，而是逐渐变得模糊。即使在极高的压缩下，模型依然能“抓住”大部分核心信息。

- **原因分析**：作者对性能下降给出了两个合理的解释。一是长文档布局更复杂，低分辨率图像无法承载所有排版细节。二是文字本身在低分辨率下会变得模糊不清，就像你看一张被严重压缩的图片。这个“模糊化”的过程，作者敏锐地将其与人类的“遗忘机制”联系起来，这是一个非常深刻的洞见，我们将在讨论部分再次看到。

深远意义：这个实验的结论是革命性的。它证明了我们可以用一种截然不同的方式来处理文本。传统的思路是不断扩大 LLM 的“内存”（上下文窗口），而这条路计算成本高昂。DeepSeek-OCR 提出的新思路是，我们可以把文本信息“编码”到视觉空间中，从而用极小的“视觉内存”来承载海量的文本信息。更妙的是，这种方法几乎是“免费”的，因为它完全复用了 VLM 本身就必须具备的视觉编码器，没有增加任何额外的硬件或架构负担。这为解决 LLM 的长上下文问题开辟了一条全新的、极具成本效益的技术路径。

4.2. OCR 实用性能

表格 3 | OmniDocBench 性能测试

表格内容 我们使用 OmniDocBench 来测试 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，值越小表示性能越好。“Tokens”代表每页平均使用的视觉令牌数，“200dpi”表示使用 fitz 将原始图像插值到 200dpi。对于 DeepSeek-OCR 模型，“Tokens”列括号中的值代表根据公式 1 计算的有效视觉令牌。

模型	Tokens	英文 (overall)	中文 (overall)
流水线模型 (Pipline Models)			
Dolphin	-	0.356	0.44
Marker	-	0.609	0.497

模型	Tokens	英文 (overall)	中文 (overall)
Mathpix	-	0.108	0.364
MinerU-2.1.1	-	0.162	0.244
...	...	...	...
端到端模型 (End-to-end Models)			
Nougat	2352	0.382	0.973
InternVL2-76B	6790	0.44	0.443
Qwen2.5-VL-7B	3949	0.316	0.399
GOT-OCR2.0	256	0.141	0.411
GPT4o	-	0.128	0.399
Gemini2.5-Pro	-	0.148	0.212
MinerU2.0	6790	0.133	0.238
dots.ocr 200dpi	5545	0.125	0.16
DeepSeek-OCR (end2end)			
Tiny	64	0.283	0.361
Small	100	0.221	0.284
Base	256(182)	**0.054**	0.24
Large	400(285)	0.138	0.208
Gundam	795	0.062	0.181
Gundam-M 200dpi	1853	0.147	0.157

(注：为简洁起见，表格仅显示了部分模型和 overall 指标。英文 overall 最优值已加粗。)

表格 4 | OmniDocBench 不同类别文档的编辑距离

表格内容 结果显示，某些类型的文档仅需 64 或 100 个视觉令牌即可达到良好性能，而其他类型则需要 Gundam 模式。

模式	书籍	幻灯片	财务报告	教科书	试卷	杂志	学术论文	笔记	报纸	总体
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

模式	书籍	幻灯片	财务报告	教科书	试卷	杂志	学术论文	笔记	报纸	总体
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
Gundam-M	0.052	0.09	0.034	0.091	0.079	0.079	0.048	0.1	0.099	0.077

原文翻译 DeepSeek-OCR 不仅仅是一个实验模型；它具有强大的实用能力，可以为 LLM/VLM 的预训练构建数据。为了量化 OCR 性能，我们在 OmniDocBench 上测试了 DeepSeek-OCR，结果如表 3 所示。仅需 100 个视觉令牌（640×640 分辨率），DeepSeek-OCR 就超过了使用 256 个令牌的 GOT-OCR2.0 ；使用 400 个令牌（285 个有效令牌，1280×1280 分辨率），它在该基准测试上达到了与最先进技术相当的性能。使用少于 800 个令牌（Gundam 模式），DeepSeek-OCR 的性能优于需要近 7000 个视觉令牌的 MinerU2.0 。这些结果表明，我们的 DeepSeek-OCR 模型在实际应用中非常强大，并且由于其更高的令牌压缩率，它享有更高的研究上限。

如表 4 所示，某些类别的文档只需很少的令牌即可达到令人满意的性能，例如幻灯片仅需 64 个视觉令牌。对于书籍和报告类文档，DeepSeek-OCR 仅用 100 个视觉令牌就能取得良好性能。结合第 4.1 节的分析，这可能是因为这些文档类别中的大多数文本令牌都在 1000 以内，意味着视觉-令牌压缩比没有超过 10 倍。对于报纸，需要 Gundam 甚至 Gundam-master 模式才能达到可接受的编辑距离，因为报纸中的文本令牌有 4000-5000，远远超过了其他模式的 10 倍压缩比。这些实验结果进一步证明了上下文光学压缩的边界，这可能为 VLM 中的视觉令牌优化和 LLM 中的上下文压缩、遗忘机制的研究提供有效参考。

深度解读 在证明了理论可行性之后，这一节旨在证明 DeepSeek-OCR 在“真实世界”中的战斗力。研究者们将其放在了 OmniDocBench 这个“综合格斗场”上，与业界各路高手进行了一场公开对决。OmniDocBench 是一个非常全面的文档解析基准测试，它包含了学术论文、教科书、财务报告、甚至是手写笔记和报纸等九种不同类型的文档，能够全面地考察一个模型的综合能力。

表格 3 的核心信息：以弱胜强，效率为王 这张表格就像一场比赛的最终记分牌。指标是“编辑距离”（Edit Distance），可以理解为“改错字数”，分数越低越好。

- **越级挑战成功**：DeepSeek-OCR (Small) 模式仅用 100 个视觉令牌，其性能（英文 0.221）就显著优于使用 256 个令牌的 GOT-OCR2.0（英文 0.141，此处原文描述与数据有出入，但趋势是可比的）。更惊人的是，DeepSeek-OCR (Gundam) 模式用不到 800 个令牌，就全面超越了需要近 7000 个令牌的强大对手 MinerU2.0 （英文 0.062 vs 0.133）。这就像一个轻量级拳击手击败了一个重量级选手，充分展示了其技术的先进性和效率的巨大优势。
- **比肩顶级商业模型**：DeepSeek-OCR (Base) 模式在英文文档上的表现（0.054）甚至优于像 GPT-4o（0.128）和 Gemini-2.5-Pro（0.148）这样的顶级闭源商业模型。这证明了其性能已经达到了世界一流水平。

- **效率是关键**：贯穿整个表格最亮眼的一点，就是 DeepSeek-OCR 在“Tokens”这一列的数字总是远远小于同等性能水平的对手。这再次印证了 DeepEncoder 架构在令牌压缩上的巨大成功。

表格 4 的核心信息：因材施教，按需分配 这张表格则展示了 DeepSeek-OCR 的“智能化”和“灵活性”。它告诉我们，不是所有任务都需要动用“牛刀”。

- **简单任务用小模型**：对于像“幻灯片”（Slides）这样布局简单、文字较少的文档，Tiny 模式（64 令牌）就足够了，其性能（0.116）和更强的 Base 模式（0.08）相差不大。这在实际应用中意味着巨大的成本节约。

- **复杂任务用大模型**：而对于“报纸”（Newspaper）这种信息密度极高、排版极其复杂的文档，Tiny 和 Small 模式几乎无法工作（编辑距离高达 0.94 和 0.744），必须使用 Gundam 模式（0.122）甚至 Gundam-M 模式（0.099）才能获得可接受的结果。

- **10 倍压缩比的再次验证**：作者的分析一针见血——为什么书籍、报告用小模型就行，而报纸不行？因为前者的文本量通常在 1000 令牌以内，用 100 个视觉令牌处理，压缩比低于 10 倍，处于“甜点区”。而报纸的文本量高达四五千，用小模型处理，压缩比远超 10 倍，信息损失严重，自然效果不佳。

综合来看，这两张表格不仅证明了 DeepSeek-OCR 的强大性能和极致效率，还为我们揭示了“上下文光学压缩”技术的使用边界和最佳实践，为未来的研究和应用提供了宝贵的经验数据。

4.3. 定性研究

4.3.1. 深度解析

图 7-10 | 深度解析能力展示

图表描述 这一系列图片展示了 DeepSeek-OCR 的“深度解析”能力，即通过二次调用模型，对文档内的复杂元素进行结构化解析。

- **图 7 (金融研报中的图表解析)**:
    - **输入**: 一页金融研究报告，其中包含一个复杂的柱状图。
    - **常规 OCR 结果 (Result)**: 模型首先将整个页面的文本内容（包括图表上方的标题和下方的文字）识别出来，并转换为 Markdown 格式。
    - **深度解析 (Deep Parsing)**: 当使用特定提示（如 `Parse the figure`）时，模型会聚焦于图表本身，并将其“翻译”成一个结构化的 HTML 表格。这个表格精确地提取了图表中的所有数据点。
    - **渲染 (Rendering)**: 将深度解析出的 HTML 表格再渲染成图像，可以直观地验证提取结果的准确性。

- **图 8 (书籍中的自然图像描述)**:
    - **输入**: 一页书籍，其中包含一张照片（一群孩子和老师在教室里）。
    - **常规 OCR 结果 (Result)**: 模型识别出页面上的标题等文字信息。
    - **深度解析 (Deep Parsing)**: 针对页面中的图片，模型生成了一段非常详细、密集的文字描述（dense caption）。描述内容不仅包括了画面中的人物、物体、颜色、布局，甚至还根据墙上的文字“BIBLIOTECA”（图书馆）推断出场景可能是一个图书馆。
    - **渲染 (Rendering)**: 渲染出原始的书籍页面。
- **图 9 (化学专利中的化学式解析)**:
    - **输入**: 一页化学领域的专利文档，其中包含多个化学分子结构图。
    - **常规 OCR 结果 (Result)**: 模型识别出页面上的所有文本，包括段落编号、化学名称等。
    - **深度解析 (Deep Parsing)**: 模型自动识别出文档中的化学分子结构图，并将其转换成了 SMILES 格式的字符串。SMILES 是一种用文本表示分子结构的标准化语言。
    - **渲染 (Rendering)**: 将解析出的 SMILES 字符串再渲染回分子结构图，用于验证。
- **图 10 (数学练习题中的几何图形解析)**:
    - **输入**: 一页包含几何证明题的数学练习册，其中有一个带坐标轴的几何图形。
    - **常规 OCR 结果 (Result)**: 模型识别出页面上的标题文字。
    - **深度解析 (Deep Parsing)**: 模型尝试解析几何图形，并将其结构化地输出。虽然作者承认几何解析非常困难，但这展示了模型具备初步的几何结构理解能力。
    - **渲染 (Rendering)**: 渲染出原始的数学题页面。

原文翻译 DeepSeek-OCR 同时具备布局和 OCR 2.0 的能力，使其能够通过二次模型调用进一步解析文档内的图像，我们称之为“深度解析”。如图 7、8、9、10 所示，我们的模型可以对图表、几何、化学式甚至自然图像进行深度解析，且仅需统一的提示。

深度解读 在展示了惊人的 OCR 性能数据之后，研究者们通过这一系列的“定性研究”案例，向我们展示了 DeepSeek-OCR 更令人兴奋的一面：它不仅仅是一个“阅读者”，更是一个“理解者”和“分析者”。这种能力被称为“深度解析”。

这里的核心概念是“二次调用”。你可以这样理解：第一次调用模型时，我们让它对整个页面进行一次“宏观扫描”，识别出所有的文本和布局，就像一个图书管理员对一本书进行编目。但是，当我们发现书中有一张复杂的插图（如图表、化学式）时，我们可以再次调用模型，并用一个更具体的指令，比如“请详细分析这张图”，让模型对这个特定区域进行一次“微观精读”。

- **从像素到数据（图 7）**：在金融报告的例子中，模型将视觉上的彩色柱子，直接转换成了精确的、可用于数据分析的数字表格。这是从非结构化的视觉信息到结构化数据的跨越，对于自动化数据提取和分析领域具有革命性的意义。
- **从图像到故事（图 8）**：对于书中的一张普通照片，模型不再是简单地说“这是一张图”，而是生成了一段包含丰富细节和逻辑推理的“小作文”。它不仅描述了场景，还通过细节（墙上的文字）进行了推理，展示了其强大的通用视觉理解和常识推理能力。
- **从结构到符号（图 9）**：在化学文档中，模型将复杂的二维分子结构图，准确地翻译成了化学家们通用的、一维的 SMILES 语言。这使得机器能够“阅读”和“理解”化学文献中的核心信息，为科学知识的自动化处理和发现开辟了道路。
- **从形状到逻辑（图 10）**：尽管作者谦虚地表示几何解析仍有很长的路要走，但模型能够尝试去结构化地理解几何图形，这本身就是一个巨大的进步。它表明模型正在从识别“像素的集合”向理解“点、线、面之间的逻辑关系”迈进。

这些例子共同证明了 DeepSeek-OCR 的能力已经超越了传统的 OCR 范畴。通过 OCR 1.0 和 2.0 数据的协同训练，它已经成为了一个强大的、多才多艺的“文档理解专家”，能够在统一的框架下，处理从简单文本到复杂科学图形的各种信息，这正是其“深度解析”能力的价值所在。

4.3.2. 多语言识别

图 11 | 多语言识别能力展示

图表描述 这张图展示了 DeepSeek-OCR 对非拉丁语系语言的识别能力。

- **左侧图像对**:
  - **上图**: 一页阿拉伯语的文档。
  - **下图**: DeepSeek-OCR 对该页面的识别结果。可以看到，模型准确地识别出了从右到左书写的阿拉伯语文本，并基本保留了段落格式。
- **右侧图像对**:
  - **上图**: 一页僧伽罗语（斯里兰卡官方语言之一）的文档，其文字具有非常独特的圆形特征。
  - **下图**: DeepSeek-OCR 对该页面的识别结果。模型同样成功地识别出了这种非常见的小语种文字。
- **提示 (Prompt)**: 图中展示了两种不同的提示。`<image>\nFree OCR.` 用于输出纯文本，而 `<image>\n<grounding>Convert the document to markdown.` 则用于输出带布局的 Markdown 格式。这表明多语言文档同样支持这两种输出模式。

原文翻译 互联网上的 PDF 数据不仅包含中文和英文，还包含大量的多语言数据，这在训练 LLM 时也至关重要。对于 PDF 文档，DeepSeek-OCR 可以处理近 100 种语言。与中英文文档一样，多语言数据也支持带布局和不带布局的 OCR 格式。可视化结果如图 11 所示，我们选择了阿拉伯语和僧伽罗语来展示结果。

深度解读 这个例子展示了 DeepSeek-OCR 的“世界性”和“包容性”。在构建大型语言模型（LLM）的时代，一个关键的挑战是如何让 AI 理解和处理全球范围内的多样化信息，而不是仅仅局限于英语或中文。互联网上的知识宝库是以数百种语言存在的，一个真正强大的文档处理工具必须具备跨语言的能力。

研究者们在训练数据中包含了来自近 100 种语言的 500 万页文档，这一巨大的投入换来了丰厚的回报。如图 11 所示，模型不仅能够处理像阿拉伯语这样书写方向与英语相反的语言，还能处理像僧伽罗语这样形态独特、在公开数据集中非常罕见的小语种。

这背后体现了深度学习模型强大的“模式识别”能力。模型并不是通过学习每一种语言的语法规则来识别文字的，而是通过观察海量的数据，学会了从像素层面识别不同语言文字的视觉特征和组合规律。无论是横着写、竖着写，还是从右往左写，对模型来说，都只是一种需要学习的视觉模式。

此外，作者还强调了多语言处理的灵活性。用户可以通过不同的提示（prompt），来控制模型是只输出纯文本（`Free OCR`），还是输出保留了原始排版格式的 Markdown 文本。这种灵活性使得 DeepSeek-OCR 能够适应不同的下游应用需求。例如，如果只是想提取文本内容进行分析，纯文本格式就足够了；如果想在网页上完美地复现原始文档的版式，那么 Markdown 格式就更有用。

总而言之，强大的多语言能力，使得 DeepSeek-OCR 成为了一个真正意义上的全球化文档处理工具，能够为构建更具包容性和知识广度的下一代 LLM 提供关键的数据支持。

4.3.3. 通用视觉理解

图 12 | 通用视觉理解能力展示

图表描述 这张图通过六个小例子，展示了 DeepSeek-OCR 在处理非文档、通用图像时的多方面能力。

- **左上 (定位与计算)**:
  - **输入**: 一张包含多个数学算式的图片。
  - **提示**: `Locate <ref>11-2=</ref> in the image.` (在图中定位 `11-2=`)
  - **输出**: 模型在图中的 "11-2=" 算式周围画出了一个精确的定位框。这展示了模型的“定位”（grounding）能力。
- **中上 (图像描述)**:
  - **输入**: 一张豆瓣酱的商品图。
  - **提示**: `Describe this image in detail.` (详细描述这张图片)
  - **输出**: 模型生成了一段非常详尽的英文描述，不仅识别了瓶子上的中文字样（如“豆瓣酱”、“六月香”），还解释了它们的含义，并描述了包装的设计、颜色、材质等细节。这展示了模型的“看图说话”（caption）和跨语言理解能力。

- **右上 (指代理解)**:
    - **输入**: 一张老师和学生在教室里的图片。
    - **提示**: `Locate <ref>the teacher</ref> in the image.` (在图中定位老师)
    - **输出**: 模型在图中的老师身上画出了一个定位框。这展示了模型能够理解抽象的指代词（“the teacher”），并将其与图像中的具体人物对应起来。
- **左下 (物体检测)**:
    - **输入**: 一张包含多个卡通人物的图片。
    - **提示**: `Identify all objects in the image and output them in bounding boxes.` (识别图中所有物体并用边界框输出)
    - **输出**: 模型在图中的两个主要人物周围画出了检测框。这展示了其基本的物体检测能力。
- **中下 (中文描述与情感理解)**:
    - **输入**: 一张给消防栓画上笑脸的创意图片。
    - **提示**: `这是一张...` (这是一个...，引导模型进行补全)
    - **输出**: 模型用流畅的中文描述了这张图片，并捕捉到了其“非常友好和治愈”的情感氛围。这展示了其强大的中文语言能力和一定的情感理解能力。
- **右下 (古诗词 OCR)**:
    - **输入**: 一张写有《将进酒》部分诗句的书法图片。
    - **提示**: `OCR the image.` (识别图中文字)
    - **输出**: 模型准确地识别出了图片中的行书书法文字，尽管有一些字符因模糊未能完全识别。这展示了其在艺术化、非标准字体上的 OCR 拓展能力。

原文翻译 我们还为 DeepSeek-OCR 提供了一定程度的通用图像理解能力。相关的可视化结果如图 12 所示。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、物体检测、定位等。同时，由于加入了纯文本数据，DeepSeek-OCR 的语言能力也得到了保留。请注意，由于我们没有包括 SFT（监督微调）阶段，该模型不是一个聊天机器人，一些能力需要通过补全提示来激活。

深度解读 这一部分展示了 DeepSeek-OCR 的“多才多艺”。虽然它的核心任务是文档处理，但得益于其全面的训练数据（包含了 20% 的通用视觉数据和 10% 的纯文本数据），它在处理普通照片和执行通用视觉任务时，同样表现出色。

这些例子共同说明了几个关键点：

1. **视觉与语言的深度融合**：模型不仅仅是“看到”图像，而是能将视觉信息与语言概念紧密地联系起来。例如，它能理解“老师”这个词，并在图像中找到对应的人物；它能看到“豆瓣酱”三个汉字，并知道这是一种调味品。这就是视觉-语言模型（VLM）的核心能力。

2. **定位与检测能力 (Grounding & Detection)**：模型能根据文本提示，在图像中用边界框精确地标出对应的物体或区域。这种“指哪打哪”的能力在许多交互式应用中至关重要，比如图像编辑、智能监控等。

3. **强大的语言能力**：无论是生成详细的英文产品描述，还是富有情感的中文场景描写，亦或是识别高难度的中文书法，都展示了其背后解码器强大的语言功底。这得益于训练数据中 10% 的纯文本数据，确保了模型在“说”的方面同样出色。

4. **提示工程的重要性**：作者特别提到，模型不是一个“聊天机器人”，需要通过特定的“补全提示”（completion prompts）来激活某些能力。例如，通过给出“这是一张...”的开头，来引导模型生成一段描述。这提示我们，与现代 AI 模型交互，如何巧妙地设计提示（Prompt Engineering）是一项重要的技巧。

总的来说，图 12 向我们证明，DeepSeek-OCR 虽然是一个专注于 OCR 的“专才”，但它同时也保留了成为一个“通才”的坚实基础。这使得它不仅是一个强大的工具，更是一个充满潜力的、可供未来进一步开发和扩展的优秀模型平台。

5. 讨论

原文翻译 我们的工作代表了对视觉-文本压缩边界的初步探索，研究解码 N 个文本令牌需要多少个视觉令牌。初步结果令人鼓舞：DeepSeek-OCR 在大约 10 倍的压缩比下实现了近乎无损的 OCR 压缩，而 20 倍的压缩仍然保留了 60% 的准确率。这些发现为未来的应用指明了有前景的方向，例如，在多轮对话中对超出 k 轮的历史对话进行光学处理，以实现 10 倍的压缩效率。

深度解读 在展示了所有实验数据和结果之后，这篇论文进入了“讨论”部分。这部分不再纠结于具体的技术细节或性能指标，而是站得更高，从更宏观、更具启发性的角度来思考这项研究的意义和未来。

作者首先将他们的工作定位为一次对“边界的探索”。他们试图回答一个非常基础但重要的问题：视觉和文本这两种信息模态之间的“汇率”到底是多少？一个视觉信息单元最多能“兑换”多少个文本信息单元？他们的实验给出了一个初步的答案：在近乎无损的情况下，这个“汇率”大约是 1:10；而在允许一定信息损失的情况下，甚至可以达到 1:20。

这个发现本身就极具价值，但作者的思考并未止步于此。他们立刻将这个发现与一个非常实际的应用场景联系起来——“多轮对话”。我们知道，像 ChatGPT 这样的聊天机器人，其记忆力是有限的。当对话变得非常长时，它会忘记最开始聊了些什么。目前的解决方法是不断增大模型的“上下文窗口”，但这会导致计算成本急剧上升。

作者在这里提出了一个全新的、优雅的解决方案：我们可以用“光学压缩”来管理对话历史。比如，一个聊天机器人可以把最近的几轮对话（需要精确记忆）以纯文本形式保存在“工作记忆”中。而对于更早的、已经不太重要的历史对话，机器人可以把它们“截图”保存成一张图片。这

张图片只占用很少的视觉令牌，但却压缩了大量的对话历史。当需要回忆时，机器人只需“看一眼”这张截图，就能大致记起当时聊了什么。这样，机器人的记忆在理论上就可以变得无限长，同时计算成本又保持在一个很低的水平。这是一个极具想象力和实用价值的构想。

图 13 | 遗忘机制模拟

图表描述 这张图通过三组平行的类比，非常直观地阐释了“上下文光学压缩”如何模拟人类的遗忘机制。

- **顶部：人类记忆随时间衰减 (Memory vs. Time)**
  - **X 轴**: 时间 (Time)，从“刚刚发生”到“1 年”。
  - **Y 轴**: 记忆清晰度 (Crystal Clear -> Almost Gone)。
  - **曲线**: 记忆的清晰度随时间的流逝而迅速下降，从晶莹剔透变得模糊不清，最终几乎消失。这是著名的艾宾浩斯遗忘曲线的示意。
- **中部：人类视觉随距离衰减 (Vision vs. Distance)**
  - **X 轴**: 距离 (Distance)，从 10 厘米到 20 米。
  - **Y 轴**: 视觉清晰度 (Crystal Clear -> Almost Gone)。
  - **曲线**: 物体的视觉清晰度随距离的增加而下降。近处的物体清晰锐利，远处的物体则模糊不清。
- **底部：文本信息随压缩（分辨率降低）衰减 (Text vs. Resolution)**
  - **X 轴**: 分辨率/模式 (Resolution)，从 Gundam 模式到 Tiny 模式。
  - **Y 轴**: 文本清晰度 (Crystal Clear -> Almost Gone)。
  - **曲线**: 通过光学压缩的文本信息，其清晰度随着分辨率的降低（即压缩比的增高）而下降。在 Gundam 模式下，文本清晰可辨；到了 Tiny 模式，文本变得模糊，只能看个大概。
- **核心类比**: 图中用箭头将三者联系起来，揭示了它们之间惊人的相似性。**时间对记忆的侵蚀**，就像**距离对视觉的削弱**，而这两种自然现象，又可以被**降低图像分辨率这种计算操作**来完美模拟。

原文翻译 对于更早的上下文，我们可以逐步缩小渲染图像的尺寸，以进一步减少令牌消耗。这个假设的灵感来自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相似性——两者都表现出相似的渐进式信息丢失模式，如图 13 所示。通过结合这些机制，上下文光学压缩方法能够实现一种模仿生物遗忘曲线的记忆衰退形式，其中最近的信息保持高保真度，而遥远的记忆通过增加压缩比自然地消退。

虽然我们的初步探索显示了可扩展的超长上下文处理的潜力，其中最近的上下文保留高分辨率，而较早的上下文消耗更少的资源，但我们承认这是早期阶段的工作，需要进一步的研究。该方法为实现理论上无限上下文的架构指明了一条道路，这种架构在信息保留和计算约束之间取得了平

衡，尽管这种视觉-文本压缩系统的实际影响和局限性值得在未来的研究中进行更深入的探讨。

深度解读 这一段和图 13 是整篇论文思想最深刻、最富启发性的部分。作者在这里完成了一次从“技术”到“哲学”的升华。

核心洞见：遗忘是一种功能，而非缺陷。在 AI 领域，我们常常追求无限大、无损的记忆。但作者提醒我们，人类的智能并非如此。遗忘是人类大脑一项至关重要的功能。它帮助我们过滤掉海量不重要的信息，从而让我们能将宝贵的认知资源集中在当前最重要的事情上。一个什么都忘不掉的人，生活可能会是一场灾难。

作者通过图 13，建立了一个绝妙的类比：

- **时间的流逝**会让我们的**记忆**变得模糊。
- **空间的距离**会让我们的**视觉**变得模糊。
- 而**光学压缩（降低分辨率）会让文本信息**变得模糊。

这三者在“信息保真度随某种维度增加而衰减”这一模式上，是完全同构的。这意味着，我们可以利用一种非常简单、计算成本极低的计算机操作——“降低图像分辨率”——来模拟一种非常复杂的生物智能现象——“遗忘”。

应用构想：AI 的分层记忆系统。基于这个洞见，作者描绘了一个未来 AI 记忆系统的蓝图。这个系统是分层的、动态的：

- **高保真区（近期记忆）**：最近的对话、最重要的信息，被渲染成高分辨率的图像（例如 Gundam 模式），以极高的精度被保存下来，随时可以精确调用。
- **中保真区（中期记忆）**：稍早一些的信息，被渲染成中等分辨率的图像（例如 Base 模式），细节有所丢失，但核心内容依然清晰。
- **低保真区（远期记忆）**：非常久远的历史信息，被渲染成极低分辨率的图像（例如 Tiny 模式），只剩下一些模糊的轮廓和主旨。

通过这种方式，AI 可以像人一样，拥有一个“有重点”的记忆系统。它既能记住海量的信息（理论上无限），又不会被信息的洪流所淹没，因为大部分信息都以一种高度压缩、低保真度的“模糊”形式存在。这不仅极大地节省了计算资源，更可能让 AI 变得更“专注”、更“智能”。

尽管作者承认这还只是一个初步的构想，但它无疑为解决超长上下文问题，乃至构建更接近人类智能的 AI，提供了一个全新的、极具吸引力的视角。

6. 结论

原文翻译 在这份技术报告中，我们提出了 DeepSeek-OCR，并通过这个模型初步验证了上下文光学压缩的可行性，证明了模型可以从少量视觉令牌中有效地解码出超过其数量 10 倍的文本令牌。我们相信这一发现将有助于未来 VLM 和 LLM 的发展。此外，DeepSeek-OCR 是一个高度实用的模型，能够进行大规模的预训练数据生产，是 LLM 不可或缺的助手。当然，仅靠 OCR

不足以完全验证真正的上下文光学压缩，我们未来将进行数字-光学文本交错预训练、大海捞针测试以及其他评估。从另一个角度看，光学上下文压缩仍然提供了巨大的研究和改进空间，代表着一个有前景的新方向。

深度解读 论文的结论部分是对整个研究工作的凝练总结和最终展望，它简洁而有力地重申了核心贡献和未来方向。

总结核心贡献：

1. **验证了一个新理论**：结论开门见山，再次强调了最核心的发现——“上下文光学压缩”是可行的。并且给出了一个关键的量化指标：可以用 1 份视觉信息，高效解码出超过 10 份的文本信息。这个发现为整个 AI 领域提供了一种全新的、思考信息压缩和处理的视角。

2. **打造了一个好工具**：结论强调了 DeepSeek-OCR 的“高度实用性”。它不仅是一个用于验证理论的“实验品”，更是一个可以投入大规模生产的“工具”，能够成为 AI 产业的“基础设施”，为其他模型的训练提供数据支持。

承认局限与展望未来： 科学的态度不仅在于展示成果，也在于承认工作的局限性。作者坦诚地指出，目前只用了 OCR 任务来验证这个想法，但这还不够。为了真正证明“光学压缩”的普适性，未来还需要进行更严苛的测试。

- **数字-光学文本交错预训练**：这意味着在训练模型时，让它同时处理纯文本和被压缩成图像的文本，让模型学会在这两种信息形态之间自由切换和理解。

- **大海捞针测试 (Needle-in-a-Haystack)**：这是一种专门用来测试模型长上下文能力的评估方法。测试者会把一个特定的信息（“针”）藏在一大堆无关的文本（“干草堆”）中，然后提问关于这个“针”的问题，看模型能否准确地把它找出来。用这种方法来测试光学压缩后的长文本，将是对其信息保真度的终极考验。

最后，作者以一个开放和充满信心的姿态结束了全文。他们认为，“光学上下文压缩”不仅仅是一项技术，更是一个充满想象空间和研究潜力的“新方向”。这番话既是对自己工作的总结，也是向整个科研社区发出的邀请，鼓励更多的研究者加入到这个激动人心的探索中来。

参考文献

Marker. URL https://github.com/datalab-to/marker. Mathpix. URL https://mathpix.com/. Ocrflux, 2025. URL https://github.com/chatdoc-com/OCRFlux. G. AI. Gemini 2.5-pro, 2025. URL https://gemini.google.com/. 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. L. Blecher, G. Cucurull, T. Scialom, and R. Stojnic. Nougat: Neural optical understanding for academic documents. arXiv preprint arXiv:2308.13418, 2023. 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. 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. 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. M. Dehghani, J