Coding Nexus · Follow publication

Member-only story

Top 4 Open-Source OCR Models

4 min read · 23 hours ago

Algo Insights Following

I've spent a lot of time lately testing different OCR models — trying to find one that truly reads PDFs, scans, and screenshots the way humans do.
Four open-source projects really stand out this year: **DeepSeek-OCR**, **Olmo-OCR 2**, **Qwen3-VL**, and **Dots.OCR**.

Each of them does the same thing (read documents) but in completely different ways. Here's what I've learned from working with them.

. . .

1. DeepSeek-OCR-3B — The Speed Demon

If you want a fast, reliable OCR model that can chew through images and PDFs without drama, **DeepSeek-OCR** is it.

It runs on GPUs using either **vLLM** or **Transformers**, so you can pick whichever fits your setup.

Setting it up takes a few commands:

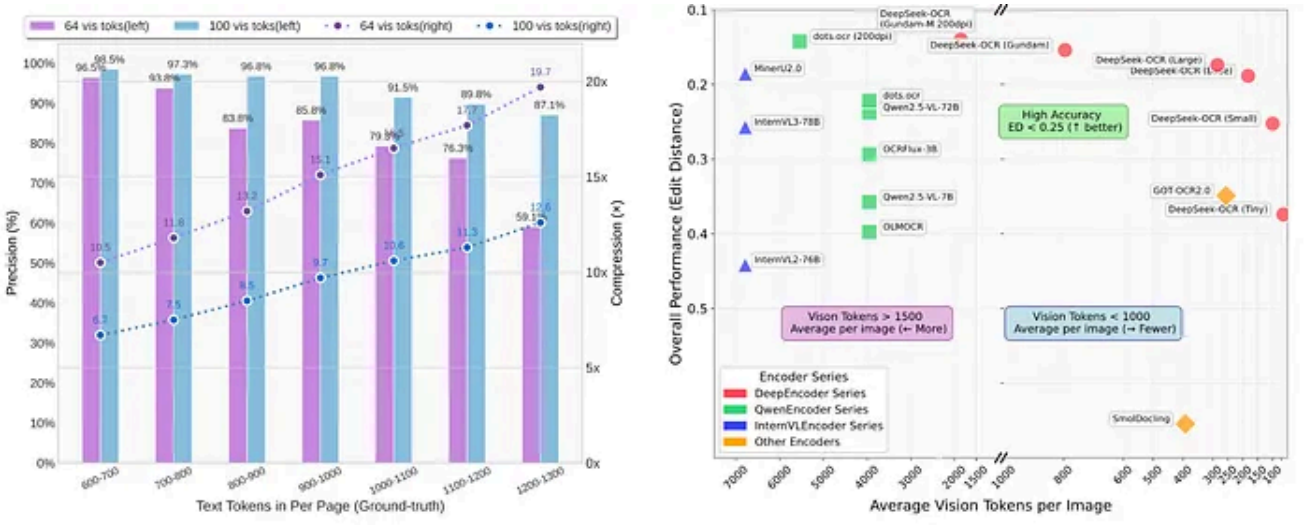
```
git clone https://github.com/deepseek-ai/DeepSeek-OCR.git
cd DeepSeek-OCR
conda create -n deepseek-ocr python=3.12.9 -y
conda activate deepseek-ocr
pip install torch==2.6.0 torchvision==0.21.0 torchaudio==2.6.0 --index-url http
pip install -r requirements.txt
pip install flash-attn==2.7.3 --no-build-isolation
```

Once you're ready, here's a quick Python snippet to try:

```
from vllm import LLM, SamplingParams
from PIL import Image

llm = LLM(model="deepseek-ai/DeepSeek-OCR")
img = Image.open("your_image.png").convert("RGB")
out = llm.generate([{"prompt": "<image>\nRead this.", "multi_modal_data": {"ima
                   SamplingParams(max_tokens=8192))
print(out[0].outputs[0].text)
```

It'll generate text directly from your image. On an A100 GPU, it's incredibly fast—about 2500 tokens per second for PDFs.



(a) Compression on Fox benchmark

(b) Performance on Omnidocbench

• • •

2. Olmo-OCR 2 — The PDF Whisperer

Now, if you're handling **academic papers**, **manuals**, or **technical PDFs**, **Olmo-OCR 2** is in a league of its own.

It's explicitly trained on significant, messy documents — the kind with tables, equations, and headers that always confuse older OCRs.

It doesn't just give you text; it provides you with *Markdown* with a clean reading order.
So instead of a jumble of lines, you get organized sections, lists, and properly formatted math.

Set up's simple on Ubuntu:

```
sudo apt-get update
sudo apt-get install poppler-utils ttf-mscorefonts-installer fonts-crosextra-ca
```

Once that's set up, you can process a folder of PDFs and generate structured Markdown.
It's not lightweight (you'll need a GPU with 15 GB+ memory), but it's efficient — about $200 per million pages, based on their benchmarks.

. . .

3. Qwen3-VL — The All-Round Genius

This one's more than OCR — **Qwen3-VL** is essentially a *vision-language brain*. It reads, reasons, and even understands interfaces.

You can throw books, screenshots, or even videos at it. It'll not only extract text but also *understand what's happening*.
It's like having a research assistant who can also see your screen.

A few highlights:

- Reads in **32 languages**
- Handles multi-column layouts, equations, and GUIs
- Understands spatial reasoning (like "what's to the left of this icon")
- Works with **Transformers ≥ 4.57**

Install it with:

```
pip install "transformers>=4.57.0"
```

If you want to combine OCR with visual reasoning — such as asking questions about a document or creating an interactive reader — this is the model to consider.

. . .

4. Dots.OCR-3B — Small, Simple, and Surprisingly Good

Dots.OCR is my favorite "underdog" model.
It's small compared to the others (just 1.7 B parameters), but it performs like a heavyweight.

It manages text, tables, and layout detection independently — no additional modules or multi-model configurations.
And it works for multiple languages too.

Here's how to set it up:

```
conda create -n dots_ocr python=3.12
conda activate dots_ocr
git clone https://github.com/rednote-hilab/dots.ocr.git
cd dots.ocr
pip install torch==2.7.0 torchvision==0.22.0 torchaudio==2.7.0 --index-url http
pip install -e .
```

It runs quickly and provides structured output that's easy to parse. If you're short on GPU memory or want something fast and reliable, **Dots.OCR** hits the sweet spot.

. . .

Final Thoughts

What I appreciate about these models is how they each advance OCR in unique ways.

Model What It's Best At: **DeepSeek-OCR** combines speed and flexibility. **Olmo-OCR creates** clean PDFs and converts them into Markdown. **Qwen3-VL** offers visual reasoning and multilingual support. **Dots.OCR** is lightweight yet produces high-quality OCR.

If you're developing your own OCR pipeline, you might even combine them — for example, using DeepSeek for images and Olmo for PDFs.

OCR has evolved far beyond simply "reading text." These models can *grasp* structure, formatting, and even the reasoning behind what they read — and that's what makes them exciting to use today.

Ocr Deepseek ChatGPT AI Ai Tools

Follow

Published in Coding Nexus

7.9K followers · Last published 2 hours ago

Coding Nexus is a community of developers, tech enthusiasts, and aspiring coders. Whether you're exploring the depths of Python, diving into data science, mastering web development, or staying updated on the latest trends in AI, Coding Nexus has something for you.

Following

Written by Algo Insights

3.4K followers · 6 following

Algo Insights: Unlocking the Future of Trading with Code, Data, and Intelligence.
https://linktr.ee/Algo_Insights

Responses (1)

Bgerby

What are your thoughts?

Amber Boone
23 hours ago

Great pos. Thanks for including the compute power in each run

Reply

More from Algo Insights and Coding Nexus

In Coding Nexus by Algo Insights

4 Open-Source Tools That Made Me Rethink My Dev Setup

I've been coding for a while now. Most of the tools we use every day... they've been the same for years. Editors, browsers, frameworks. Just...

Sep 3 451 9

In Coding Nexus by Civil Learning

The Guy Who Let ChatGPT Trade for Him—and Somehow It Worked

You know how everyone says, “Don’t let AI touch your Money”? Well, someone on Reddit decided to ignore that.

Oct 8 166 7

In Coding Nexus by Code Coup

10 n8n Tricks Beginners Wish They Knew Earlier

Everyone loves saying n8n is beginner-friendly. Honestly? Not really.

Sep 30 278 6

In Coding Nexus by Algo Insights

How to Build Your Own AI Rig for Running Local LLMs (Gemma, Mistral, Qwen, GPT-OSS and Llama)

About three months ago, I realised I was utterly dependent on companies that didn't care about anything except power, Money, and control.

Oct 8 354 12

See all from Algo Insights

See all from Coding Nexus

Recommended from Medium

In Coding Nexus by Code Coup

Claude Desktop Might Be the Most Useful Free Tool You'll Install This Year

I didn't expect much when I first saw the announcement for Claude Desktop. Another AI wrapper, I thought. Maybe with a shiny UI.

3d ago 160 3

In AI Software Engineer by Joe Njenga

This Viral DeepSeek OCR Model Is Changing How LLMs Work

This DeepSeek OCR model hit an overnight success not seen in any other release—4k+ GitHub stars in less than 24 hours and more than 100k...

3d ago 184 2

Pawel

DeepSeek-OCR: The New OCR Breakthrough

DeepSeek released DeepSeek-OCR yesterday. And it’s not just another OCR model. It’s a proof-of-concept that could fundamentally change how...

5d ago 29 1

Bytefer

The Self-Hosted Alternative to Notebook LM/Perplexity Is Here.

NotebookLM and Perplexity Are Impressive, But They're Missing Half the Picture.

5d ago 100 2

Michael Orozco-Fletcher

OCR Scenario 3: How Well Can Tesseract Read Documents?

Sep 10

In Towards AI by Eivind Kjosbakken

How to Select the 5 Most Relevant Documents for AI Search

Improve the document retrieval step of your RAG pipeline

3d ago 145 2