Coding Nexus · <u>Follow publication</u>

VERT.sh — The Open-Source, Local, and Free File Converter That Runs Entirely on Your Device

5 min read · Oct 14, 2025

👤 Algo Insights Following ⌄

▶ Listen   ⬆ Share   ⋯ More

You know that uncomfortable feeling when you upload a personal file to one of those random "free converter" websites? Yeah — me too.

You wait for it to process, and for a moment, you wonder: *where exactly did that file go?*

That's the problem **VERT.sh** quietly addresses. It's an open-source file converter that operates entirely on your device — no uploads, no data sharing, no hidden cloud servers—just pure, local conversion magic powered by **WebAssembly**.

And the best part? It's totally free and open source.

· · ·
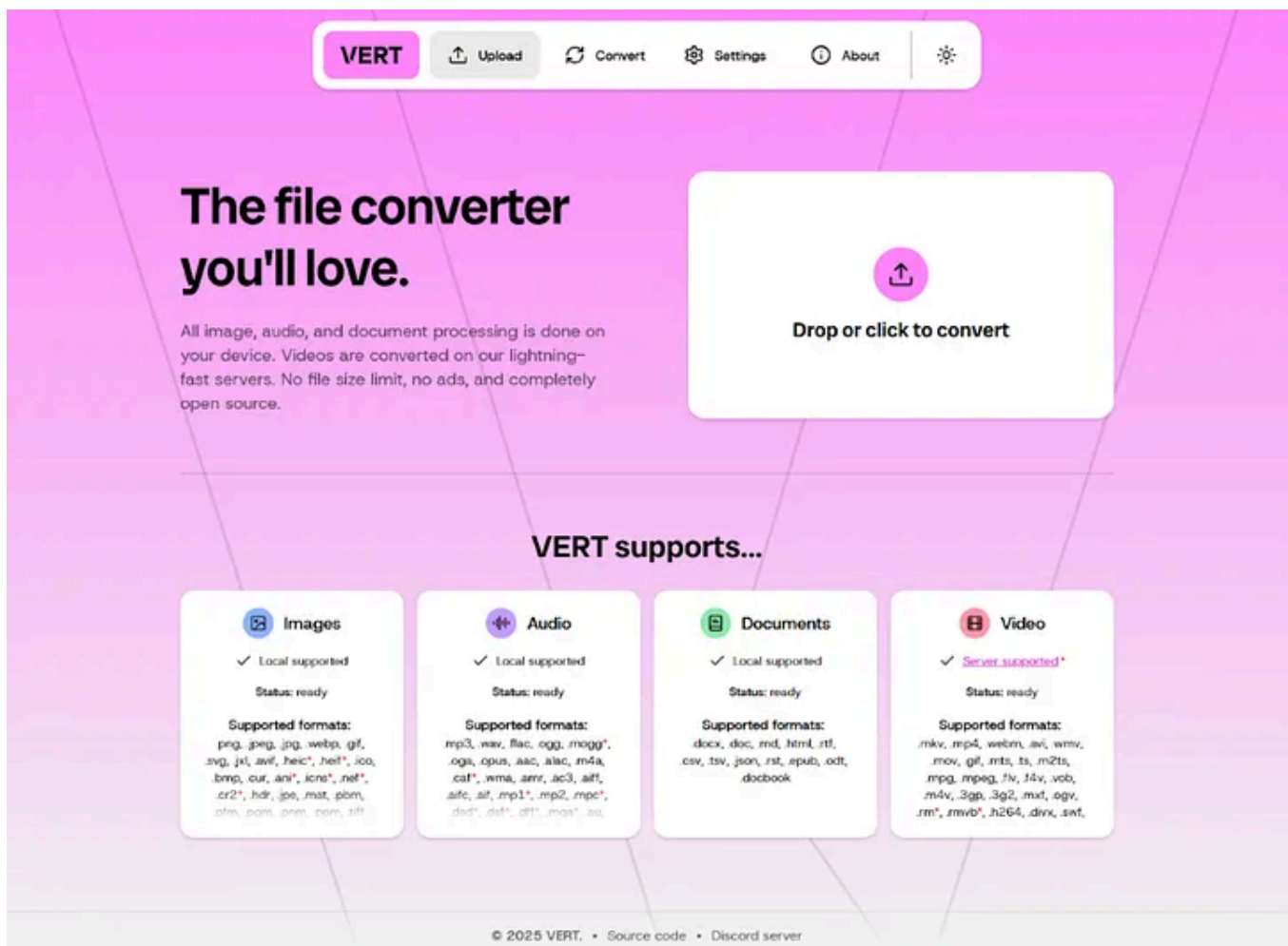
What Exactly Is VERT.sh?

Think of **VERT** as your personal Swiss Army knife for files. You can throw any file at it — image, video, audio, document — and it figures out how to convert it.

The crazy part is that it does all of that *right in your browser*.

It's built with **Svelte** and **TypeScript**, but the main trick is its use of **WebAssembly (WASM)** to perform file conversions locally. That's how it avoids sending files to a server.

Here's a brief list of its capabilities:

- Convert files directly on your device using WebAssembly.

- No size or upload limits.

- Works with over **250 formats** (yep, 250+).

- Has a clean, simple Svelte interface.

- Completely free and open source.

- You can self-host it if you like.

It's like possessing your own file conversion server — without actually operating one.

. . .

Why the "Local" Part Matters

Most of those file conversion sites depend on servers running tools like FFmpeg or ImageMagick. When you upload a file, it actually goes somewhere — perhaps a data

centre or someone's rented AWS instance.

With VERT, none of these things occur.

WebAssembly is what makes this magic work. It essentially allows your browser to run compiled code — like C, C++, or Rust — nearly as fast as a native app. So, the conversion happens inside your browser tab, not on someone else's machine.

No waiting for uploads, no data leaks, no "oops, we keep your files for 24 hours to improve our service."

.   .   .

Getting Started

VERT uses <u>Bun</u> — the new all-in-one JavaScript runtime that's incredibly fast and gaining popularity among developers.

You'll need to install **Bun** first.

For macOS & Linux:

```
curl -fsSL https://bun.sh/install | bash
```

For Windows:

```
powershell -c "irm bun.sh/install.ps1 | iex"
```

Once that's done, you can quickly set up VERT in a few simple steps.

```
# Clone the repo
git clone https://github.com/VERT-sh/VERT
cd VERT/
```

```
# Install dependencies
bun i
```

Then start it up with:

```
bun dev
```

Open your browser and head over to **http://localhost:5173** — and boom, you're looking at VERT running locally.

. . .

Building for Production

If you're planning to host your own instance, perhaps on your domain, you'll need a file with a few values:

```
PUB_HOSTNAME=example.com
PUB_PLAUSIBLE_URL=https://plausible.example.com
PUB_ENV=production
PUB_VERTD_URL=https://vertd.vert.sh
```

Then build it:

```
bun run build
```

That creates a production-ready version in the `build/` folder.
You can serve it with **nginx**, **Caddy**, or any other web server.

. . .

Prefer Docker? That Works Too.

VERT provides a Docker setup for those who prefer to containerise everything (which is most of us).

Here's how to build it:

```
docker build -t vert-sh/vert \
  --build-arg PUB_ENV=production \
  --build-arg PUB_HOSTNAME=vert.sh \
  --build-arg PUB_PLAUSIBLE_URL=https://plausible.example.com \
  --build-arg PUB_VERTD_URL=https://vertd.vert.sh \
  --build-arg PUB_DONATION_URL=https://donations.vert.sh \
  --build-arg PUB_STRIPE_KEY="" .
```

And to run it:

```
docker run -d \
  --restart unless-stopped \
  -p 3000:80 \
  --name "vert" \
  vert-sh/vert
```

That maps port `3000` on your machine to port `80` inside the container.
You can also use `docker compose up -d` If you want a cleaner setup.

. . .

About That Video Conversion Part...

Okay, so there's one minor detail: Video conversion is too demanding to perform entirely in the browser.

That's where `vertd` comes in.

It's a tiny Rust-based daemon that wraps around **FFmpeg** — the same tool most video editors secretly rely on.
Basically, `vertd` handles the heavy lifting for video conversions, while the VERT web app provides a nice interface.

Using the official hosted instance means VERT's servers process your videos. But if you want *everything* to stay local, you can efficiently run your own `vertd` instance.

. . .

How to Run `vertd` Locally

1. **Download** the latest `vertd` release for your OS.

2. **Run** the server — by default, it listens on port. `24153` .

3. In your VERT settings, set the instance URL to:

```
http://localhost:24153
```

Once you do that, your videos will be converted locally using your computer's CPU. No external servers, no uploads — just pure speed.

. . .

Why I Think VERT Deserves Attention

VERT is one of those quiet open-source gems that address a real-world problem using modern technology.

It's built on some of the most interesting tools in today's web ecosystem — **Svelte, TypeScript, Bun, Rust, and WebAssembly** — and they all work together seamlessly.

There's no giant startup behind it. No paywall. No "Pro" tier.
Just a genuinely helpful, privacy-friendly app that works right out of the box.

If you've been curious about how far browsers have come with local computing power, this is one of those projects that makes you stop and go, *wait, that's possible now?*

. . .

Final Thoughts

I've tried many online converters over the years, and most of them are either filled with ads or seem unreliable.

VERT changes that.

It operates on your machine, protects your privacy, and provides you with complete control.
It's the type of open-source project that shows how much the web has advanced — from simple static pages to fully functional, native-like apps that operate inside your browser tab.

If you haven't tried it yet, head over to **vert.sh** and take it for a spin.
No signup. No spam. Just a clean, innovative tool that does what it says.

AI   Ai Agent   Generative Ai Tools   Ai Tools   Data
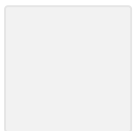
Responses (3)

Bgerby

What are your thoughts?

Ernest Tan
Oct 15

it needs to be made into mobile

👏 3   Reply

Aron T
Oct 18

No pdf conversions is a big gap. Considering all the effort and resource demands of installing and running, brew installable command line tools seem easier and more versatile, at least for images and text files.

👏   💬 1 reply   Reply

Publique2007
Oct 16

Nice, thanks for the tip.

👏   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.

In Coding Nexus by Code Coup

10 n8n Tricks Beginners Wish They Knew Earlier

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

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.

See all from Algo Insights

See all from Coding Nexus

Recommended from Medium

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.

In Towards Deep Learning by Sumit Pandey

Why Everyone Will Want DGX Spark on Their Desk — Yes, Everyone

I just saw this picture today and was amazed, I've been waiting for this moment for a long time. (No, it's not Elon.) It's that tiny...

Oct 14 · 👏 56 · 💬 11

Micheal Lanham

Microsoft Just Solved the AI Agent Problem (And Open-Sourced the Solution)

How the new Agent Framework unifies Semantic Kernel and AutoGen to bring enterprise-grade AI agents from prototype to production

Oct 14   👏 46   💬 1

Bytefer

Voice AI for Everyone: Hyper-Realistic, Instant Cloning TTS That Fits in Your Phone

By running a super-realistic 0.5B-parameter TTS locally, NeuTTS Air enables instant voice cloning, real-time performance, and...

Oct 9   👏 86

Claude Skills: The $3 Automation Secret That's Making Enterprise Teams Look Like Wizards

How a simple folder is replacing $50K consultants and saving companies literal days of work

RAG is Not Dead! No Chunking, No Vectors, Just Vectorless to Get the Higher Accuracy

Over the past two years, I have written numerous articles on how Retrieval-Augmented Generation has become a standard feature in nearly all...

Suleman Safdar

The 7-Line Python Script That Built Me a Mini AI Business

How I turned simple automation into an income stream using OpenAI, FastAPI, and a $5 cloud server

⭐ Oct 14 👋 136 💬 3

See more recommendations