

Local AI Upscaling: Make Blurry Images Sharp Without the Cloud

March 6, 2026 · by Mark Bartlett

[Download this guide as PDF](#)

Quick Answer: Install Upscayl (free, open source), drag in your image, click upscale. That's it. It uses Real-ESRGAN under the hood and runs on any GPU with Vulkan support, including NVIDIA, AMD, and Intel. For batch processing hundreds of images, use the Real-ESRGAN command line. For video upscaling, use Video2X (free, also open source, but slow). Most upscaling models are under 70MB and run on GPUs as old as the GTX 1060. You don't need a beefy rig for this.

More on this topic: [ComfyUI vs A1111 vs Fooocus](#) · [Best Used GPUs for Local AI](#) · [VRAM Requirements](#)

You've got a shoebox of old family photos scanned at 640x480. Or game screenshots you want as wallpaper. Or 200 product images that need to be twice as big for a website redesign. Cloud upscaling services charge \$5-10/month and send every image to someone else's server.

Local upscaling runs on your machine, costs nothing after setup, and finishes faster than uploading. The models are tiny compared to LLMs. Real-ESRGAN, the most popular upscaling model, is 67MB. A GTX 1060 from 2016 handles it fine.

Here's how to pick the right tool, do your first upscale in two minutes, and batch-process a whole folder overnight.

Why local instead of cloud

Family photos, medical images, client work, anything you wouldn't email to a stranger. Cloud services process your images on their servers. Some keep copies. Local upscaling never sends data anywhere.

Topaz Gigapixel went subscription-only in September 2025, \$29/month. Cloud APIs charge per image. Local tools are free and open source.

No upload, no waiting for a server queue, no download. Real-ESRGAN processes a photo in 2-6 seconds on a mid-range GPU.

Need to upscale 500 photos? Cloud services throttle you or charge extra. Locally, you point the tool at a folder and walk away.

The models: what actually does the upscaling

The software is just a wrapper. The model does the work. Here are the ones worth knowing.

Model	Scale	Best for	File size
RealESRGAN_x4plus	4x	General photos, the default choice	67 MB
RealESRGAN_x2plus	2x	When 4x is overkill (less artifact risk)	67 MB
4x-UltraSharp	4x	Sharp edges, text, UI screenshots, digital art	67 MB
RealESRGAN_x4plus_anime_6B	4x	Anime, illustration, cel-shaded art	17 MB
4x-Foolhardy-Remacri	4x	Texture reconstruction, game assets	67 MB
realesr-animevideov3	4x	Anime video, frame-by-frame	8 MB

RealESRGAN_x4plus handles 80% of what people throw at it. Start there. Switch to 4x-UltraSharp if you need cleaner edges on text or screenshots. Use the anime model for anything with flat colors and hard lines.

All of these are free. You can grab them from [OpenModelDB](#) or they come bundled with most upscaling apps.

2x vs 4x: when bigger isn't better

A 4x upscale takes a 500x500 image to 2000x2000. That's 16x more pixels the model has to invent. On a clean, high-res source image, 4x can introduce artifacts: fake skin pores, hallucinated fabric textures, grass patterns that weren't there.

If your source image is already decent (phone photos from the last 5 years, for example), try 2x first. The output is cleaner, the file size is smaller, and the processing is faster. Save 4x for genuinely low-res sources, old scans, webcam captures, small thumbnails you need to blow up.

The software: pick one and go

Upscayl (easiest, recommended for most people)

[Upscayl](#) is a free, open-source desktop app. Download it, install it, drag in a photo, pick a model, click upscale. The current version (v2.15.0, December 2024) added a High Fidelity model, clipboard paste support, and a lens viewer for comparing before/after.

It uses Real-ESRGAN's NCNN backend with Vulkan, so it works on NVIDIA, AMD, and Intel GPUs. No CUDA required. 43,000+ stars on GitHub.

The two-minute walkthrough:

1. Download Upscayl from upscayl.org (Windows, macOS, Linux)
2. Install and open it
3. Drag an image into the window (or click "Select Image")
4. Pick "General Photo (Real-ESRGAN)" from the model dropdown
5. Choose 4x scale
6. Click "Upscale"
7. Done. Output saves next to your original by default

For batch processing, switch to the "Batch" tab, point it at a folder, and let it run. On a mid-range GPU, expect 10-20 seconds per image.

Limitation: Requires a Vulkan-compatible GPU. No CPU fallback. If you have a very old laptop with integrated graphics, Upscayl won't work. Use the Real-ESRGAN CLI instead.

Real-ESRGAN CLI (batch processing, scriptable)

If you need to upscale hundreds of images unattended, or you want to script it into a workflow, the command line is the way.

For NVIDIA GPUs (PyTorch/CUDA):

```
pip install realesrgan

# Single image
python -m realesrgan -i photo.jpg -o photo_upscaled.png -n RealESRGAN_x4plus -s 4
```

```
# Entire folder
python -m realesrgan -i /path/to/photos/ -o /path/to/output/ -n RealESRGAN_x4plus
```

For AMD/Intel GPUs (Vulkan, no Python needed):

Download the pre-built binary from [Real-ESRGAN-ncnn-vulkan](#). Unzip and run:

```
./realesrgan-ncnn-vulkan -i photo.jpg -o photo_upscaled.png -n realesrgan-x4plus
```

Key flags:

- `--tile 256` – use tiling to reduce VRAM usage (lets you run on 2GB GPUs)
- `--face_enhance` – apply GFPGAN face enhancement
- `-s 2` – output at 2x instead of 4x

Batch 500 photos overnight: Point the CLI at your folder, pipe the output somewhere, and let it run. At 3-6 seconds per image on an RTX 3060, 500 images take about 30-50 minutes.

chaiNer (most flexible, power users)

[chaiNer](#) (v0.25.1, October 2024) is a node-based image processing app. Think of it like a visual pipeline builder: load image, denoise, upscale with model X, adjust colors, sharpen, save. You connect nodes with wires and the data flows through.

It supports PyTorch, NCNN, ONNX, and TensorRT backends, and can load any model from OpenModelDB. That's over 160 specialized upscaling models for photos, anime, game textures, manga, old film.

chaiNer is overkill if you just want to drag and drop a photo. It's the right tool when you want a repeatable processing pipeline: denoise with one model, upscale with another, color-correct, then batch the whole thing.

ComfyUI (if you're already generating images)

If you're using [ComfyUI for Stable Diffusion or Flux](#), you can upscale inside your generation workflow. Four methods, each with different quality/speed/VRAM trade-offs:

1. **ESRGAN model node** – `Load Upscale Model` → `Upscale Image (Using Model)` → `Save Image`. Fast (5-6 seconds), 2-6 GB VRAM. Functionally identical to standalone Real-ESRGAN.

2. **Latent upscale + KSampler** – Generate at base resolution, pass the latent through a Latent Upscale node (2x), then a second KSampler at low denoise (0.3-0.5). Everything stays in latent space. 4-10 GB VRAM depending on model.
3. **Ultimate SD Upscale** – Processes your image tile-by-tile through the diffusion model. First upscales with ESRGAN, then re-renders each tile through img2img. Higher quality than pure ESRGAN. 8-12 GB VRAM.
4. **ControlNet Tile + Ultimate SD Upscale** – The highest quality ComfyUI method short of SUPIR. ControlNet Tile feeds color and structure as conditioning, keeping the diffusion model close to the original. 10-14 GB VRAM.

For Flux users, the **Flux.1-dev-Controlnet-Upscaler** from Jasper AI is a dedicated ControlNet trained for upscaling with Flux. Set strength to 0.6, GGUF Q4_K_M variant works on 8-12 GB VRAM.

SUPIR (maximum quality, damaged photos)

SUPIR (Scaling Up to Excellence) uses SDXL, a 2.6-billion-parameter diffusion model, as a generative prior. It also integrates LLaVA (a vision-language model) to auto-caption your image and guide restoration. It understands what's in your image and generates appropriate detail – skin texture looks different from metal, fabric weave differs from concrete.

On a badly degraded photo, the difference between SUPIR and Real-ESRGAN is immediately obvious. SUPIR reconstructs plausible facial features from 20x20 pixel faces, works around heavy JPEG compression damage, and generates individually distinguishable foliage where Real-ESRGAN produces uniform textures.

The trade-offs are real: SUPIR needs 12GB+ VRAM (8GB minimum in fp8 mode), takes 30-60 seconds per image, is NVIDIA-only, and has a non-commercial license. It hallucinates without a good prompt – always provide a descriptive prompt and negative prompt. It's terrible at text (34.6% OCR accuracy, worse than bicubic interpolation) and adds unwanted grain to anime/illustration. Use it for the 10 photos that matter most, not for batch processing 500 vacation shots.

Run it through [kijai/ComfyUI-SUPIR](#) (2,200 stars, actively maintained) or the MonsterMMORPG standalone enhanced fork with one-click installers.

SUPIR Config	VRAM Needed
fp8 UNet, no LLaVA	~8 GB
fp16, no LLaVA	~12 GB

SUPIR Config	VRAM Needed
fp16 + LLaVA (auto-captioning)	~30 GB

VRAM requirements at a glance

Method	Min VRAM	Speed (per image)	GPU Support
Real-ESRGAN (tiled)	2 GB	2-10 sec	NVIDIA, AMD, Intel
Upscayl	2 GB	10-20 sec	Any Vulkan GPU
ComfyUI ESRGAN node	2-4 GB	5-6 sec	NVIDIA, AMD (ROCm)
ComfyUI Ultimate SD Upscale (SDXL)	8 GB	3-8 min	NVIDIA, AMD (ROCm)
ComfyUI ControlNet Tile (SDXL)	10 GB	5-10 min	NVIDIA, AMD (ROCm)
SUPIR (fp8, no LLaVA)	8 GB	30-60 sec	NVIDIA only
SUPIR (fp16, no LLaVA)	12 GB	30-60 sec	NVIDIA only

Quality comparison by content type

Content Type	Best Method	Avoid
Clean photos	ControlNet Tile or Real-ESRGAN x4plus	—
Damaged/old photos	SUPIR (v0Q)	Real-ESRGAN (amplifies noise)
Portraits and faces	SUPIR + CodeFormer	—
Anime and illustration	Real-ESRGAN anime_6B	SUPIR (adds grain to flat colors)
Text and UI screenshots	4x-UltraSharp (ESRGAN)	SUPIR (generates wrong characters)
Batch processing (500+ images)	Real-ESRGAN / Upscayl	SUPIR or diffusion methods

The free Topaz Gigapixel alternative question

Topaz went subscription-only in September 2025 – \$29/month. The perpetual licenses are gone.

What Topaz does that free tools don't: face recovery from pixelated blobs, combined denoise + sharpen + upscale in one pass, Lightroom/Photoshop plugin integration. What free tools do just as well: clean 2-4x upscaling (Real-ESRGAN), anime/illustration upscaling (Real-ESRGAN anime model), maximum quality photo restoration (SUPIR matches or exceeds Topaz), and full pipeline automation (ComfyUI, chaiNNer).

For clean photos where you just need more pixels, Upscayl gets you 90% of Topaz quality. SUPIR can match or beat Topaz on heavily degraded photos but demands more GPU and setup time. Topaz earns its subscription for professional photographers doing volume work with the Lightroom plugin.

Video upscaling: it works, but set expectations

Upscaling video is frame-by-frame image upscaling with frame interpolation on top. A 10-minute 1080p video at 30fps has 18,000 frames. At 3 seconds per frame, that's 15 hours of processing.

It works. It's just slow.

Video2X

[Video2X](#) (v6.0.0) is the most popular free option. Version 6 is a complete rewrite in C/C++, which means it's faster and less painful to install than the old Python version. It supports Real-ESRGAN, Anime4K, Real-CUGAN, and RIFE (for frame interpolation), and requires zero extra disk space during processing.

Good for: anime upscaling (480p to 1080p), old home videos, game footage.

Limitation: Windows and Linux only. No macOS support.

Flowframes

[Flowframes](#) handles frame interpolation (turning 30fps into 60fps) more than upscaling, but it can do both. Windows only. Good for making old video footage smoother.

Realistic expectations for video

Source	Target	10 min video	RTX 3060 time
480p	1080p (2x)	18,000 frames	~6-8 hours
720p	1440p (2x)	18,000 frames	~8-12 hours
1080p	4K (2x)	18,000 frames	~15-20 hours

These are rough estimates. Faster GPUs help, but video upscaling is a patience game. Queue it up before bed.

Anime upscales better than live-action because the flat colors and hard edges are easier for the models to reconstruct. Live-action footage with lots of motion, grain, and fine detail takes longer and the results are less consistent.

Hardware: you probably already have enough

Upscaling models are tiny. Real-ESRGAN is 67MB with 16.7 million parameters. For comparison, a small LLM like Qwen 3.5 9B is 5-6GB. Your GPU barely notices an upscaling model.

Your GPU	What works
GTX 1060 6GB	Upscayl, Real-ESRGAN (with tiling), chaiNNer. Handles everything in this guide
RTX 3060 12GB	Comfortable for all image upscaling, usable for video
RTX 3090 24GB	Fast at everything, including SUPIR and diffusion-based upscaling
AMD RX 6700 XT	Upscayl and NCNN tools work via Vulkan
Intel Arc A770	Vulkan support, works with Upscayl
Apple M1/M2/M3	Upscayl works on macOS. Real-ESRGAN NCNN works via MoltenVK
No GPU (CPU only)	Real-ESRGAN NCNN works on CPU. Slow (30-60 sec per image) but functional

Any gaming GPU from the last 8 years handles image upscaling. The only exception is integrated graphics without Vulkan support, and even then the CLI tools fall back to CPU.

When upscaling can't help

AI upscaling invents detail based on patterns it learned during training. It doesn't recover information that was never captured. A few cases where it falls apart:

Text below ~12px in the source. The model rebuilds letterforms as shapes and gets them wrong. Spacing changes, serifs mutate, characters drift. If you need to upscale a document, use a scanner at higher DPI instead.

Heavily compressed images. JPEG artifacts at quality 10 get sharpened right along with the real image. Real-ESRGAN amplifies the damage. For severely degraded photos, SUPIR is the better option (see the SUPIR section above – needs 12GB+ VRAM, much slower, but it can work around compression damage).

Already-clean photos. If your source is a recent phone photo at full resolution, 4x upscaling may add textures that weren't there: fake skin pores, invented fabric patterns. Use 2x or skip upscaling entirely.

The 5-minute version

If you read nothing else:

1. Download [Upscayl](#)
2. Install it
3. Drag in your image
4. Pick "General Photo" model
5. Click "Upscale"

Free. Private. Works on any GPU made in the last decade. If you need more control, more models, or batch automation, the tools above have you covered.

```
# Or do it from the terminal in one line
pip install realesrgan && python -m realesrgan -i photo.jpg -o upscaled.png -n RealESRGAN_x4plus
```

Get notified when we publish new guides.

[Subscribe – free, no spam](#)

Source: <https://insiderllm.com/guides/local-ai-upscaling-guide/>

Free guides for running AI locally