

Gemini Watermark Remover

Strip the visible Gemini watermark from local images and video

IMAGES + VIDEO · SHARP FRAME PIPELINE · AUDIO PRESERVED

This Claude Code Skill removes the visible Gemini watermark from files on your machine. Images run through the upstream removal pipeline; video gets a custom path that demuxes to frames, cleans each one, fits a single static correction across the whole clip, and remuxes with the original audio — so you get a sharp result instead of a blurred patch where the mark used to be.

01 Check your requirements

Node.js 18+	Required for everything
pnpm	One-time dependency install
ffmpeg + ffprobe	Video only — install with <code>brew install ffmpeg</code>

If you're only cleaning images, you can skip ffmpeg entirely.

02 Install the skill

The simplest route: paste this into a Claude Code session and let it handle the rest.

```
Install the Claude Code skill from
https://github.com/AryamanSi17/claude-skill-watermark-remover
— run its install.sh.
```

Under the hood that's just

```
curl -fsSL https://raw.githubusercontent.com/AryamanSi17/
claude-skill-watermark-remover/main/install.sh | bash
```

The installer clones into `~/.claude/skills/`, installs the video pipeline's dependencies, and checks for ffmpeg. Re-running it later just pulls the latest version.

Prefer to do it by hand?

```
git clone https://github.com/AryamanSil7/claude-skill-watermark-remover.git \  
  ~/.claude/skills/claude-watermark-remover-gemini  
cd ~/.claude/skills/claude-watermark-remover-gemini/vendor/video  
pnpm install
```

03 Start a fresh session

If the skill doesn't show up right away, close Claude Code and open it again. Skills are only picked up at session start — this trips up almost everyone the first time.

04 Clean a file

Inside Claude Code, invoke the skill with a path:

```
/claude-watermark-remover-gemini ~/Downloads/image.png
```

Or call the runtime directly

```
# single image  
node scripts/run.mjs remove ~/pic.png --output ~/pic_clean.png  
  
# single video, audio preserved  
node scripts/run.mjs remove ~/clip.mp4 --output ~/clip_clean.mp4  
  
# whole folder  
node scripts/run.mjs remove ~/gemini_pics --out-dir ~/clean
```

Images (.png .jpg .webp) and videos (.mp4 .mov .webm .mkv .avi .m4v) are detected automatically from the file extension.

THE ONE RULE TO REMEMBER

Pass either `--output <file>` or `--out-dir <dir>`. Never both, never neither — that's the most common way the command fails.

05 Useful flags

FLAG	APPLIES TO	PURPOSE
--overwrite	All	Replace an existing output file
--json	All	Machine-readable result
--video-method	Video	frames (default) or denoise
--no-stabilize	Video / frames	Skip temporal stabilization, for comparison
--crf <n>	Video / frames	x264 quality, default 16 (lower = better, larger)
--concurrency <n>	Video / frames	Frames in parallel, defaults to CPU count capped at 8
--passes <n>	Video / denoise	Re-run over its own output

06 Know the limits

Residual is worst on frames where something bright crosses the watermark footprint — those frames carry the least information about what was underneath. Stabilization also needs the background under the mark to move; on a frozen background it simply no-ops.

Video output is re-encoded, so it isn't bit-exact with the source, and only the first audio stream carries through — subtitles and extra audio tracks are dropped. Batch mode on a whole folder routes video through the older denoiser rather than the sharper frames path, so clean videos one at a time when quality matters.

RESPONSIBLE USE

This removes the visible watermark only. It does not strip invisible provenance signals such as SynthID, which Google embeds separately — so a cleaned file should not be assumed free of AI-generation markers. Use it on your own generations, and not to pass AI-generated media off as authentic where that would mislead people.

THAT'S IT — HAPPY BUILDING

Built by [@ai_aryaman](#) on Instagram