

PRODUCTIVITY · 10 min read · August 31, 2026

Turn YouTube Videos Into Claude Skills

The free watch skill that lets Claude actually watch a video, the exact setup, and the prompt that turns any tutorial into a one-word command you can reuse forever.

Out of the box, Claude can't watch a video. Paste a YouTube link and it guesses from the title, or pulls a caption file that misses everything shown on screen. The fix from the video is a free skill called **watch**. Once it's installed, Claude downloads the video, reads the frames, transcribes the audio, and answers questions about what actually happens in it.

That's half the trick. The other half is what you do next: tell Claude to turn what the video teaches into a **skill of its own**, so a 30-minute tutorial becomes a one-word command you run forever. This guide is both halves, in order.

WHAT YOU NEED

A Claude account with **Code execution and file creation** turned on. Here's exactly where that lives: open claude.ai, click **your initials in the bottom-left corner**, click **Settings**, then click **Capabilities** in the menu. Find **Code execution and file creation** and flip the toggle on. That's it, and you only ever do it once.

The watch skill itself is free and open source. For most YouTube videos with captions, that's everything. Only caption-less videos need an extra transcription key, and the skill tells you if it hits one.

PART 1 Install the Watch Skill

The skill lives at github.com/bradautomates/claude-video. It's open source, and the install file has been downloaded about 10,000 times. Three ways in, easiest first.

Way 1: Ask Claude to install it (chat or Cowork)

Open a chat, paste this, and send it:

PROMPT

```
Install the watch skill from this repository so I can use it in future chats:  
https://github.com/bradautomates/claude-video
```

Why it's worded that way: naming the skill tells Claude which part of that project page matters, and "so I can use it in future chats" pushes it to actually save the skill instead of setting something up that dies with the conversation. When it finishes, ask "Do you have the watch skill?" in a new chat. If yes, you're done and you never touched a file. If it didn't stick, use Way 2, which always works.

Way 2: Upload the file (the sure path)

1. Go to [the skill's GitHub page](#), click **Releases** on the right side, and download the file called **watch.skill** from the latest release.
2. In Claude, click your initials, then **Settings**, then **Capabilities**.
3. Make sure **Code execution and file creation** is on.
4. Under **Skills**, click the **+** and drop the watch.skill file in. If the file picker greys it out or hides it, rename the file to **watch.zip** first. It's the same file, and the picker accepts it.
5. Open a new chat and ask: "Do you have the watch skill?" If it says yes, you're done. In Cowork, you can also type a slash to see it in the menu.

Way 3: Claude Code (only if you use the terminal, most people don't)

PROMPT

```
/plugin marketplace add bradautomates/claude-video  
/plugin install watch@claude-video
```

FIRST RUN

The first time you use it, the skill sets up two small free tools it needs (ffmpeg and yt-dlp) on its own. That one-time setup can take a minute or two. After that, it's fast.

PART 2 Watch Your First Video

Installed, whichever way you went? Then here's what actually happens from now on. You don't open an app and you don't go anywhere new. In any chat, you type `/watch`, paste a video link, and ask a question, and Claude does the watching: it pulls the transcript, reads what's on screen, and answers you like someone who just sat through the whole thing so you didn't have to.

The command is one line: `/watch`, the link, then your question. The question is where the value is, so here are the four worth stealing, each with why it's phrased that way.

1. Learn the method (the one from the video)

PROMPT

```
/watch [paste the YouTube link] What method does this video teach? Walk me through it step by step, in the order the video presents it, and tell me which step the creator says people get wrong.
```

“Step by step, in the order the video presents it” stops Claude from reshuffling the method into a generic summary. “Which step people get wrong” surfaces the one warning that’s usually the most valuable 20 seconds of the whole video.

2. Pull what’s shown on screen, not said

PROMPT

```
/watch [link] What are the exact example subject lines he shows on screen? Read them off the slide and tell me the timestamp where each appears.
```

This is the skill’s unfair advantage: it reads slides, tables, whiteboards, and settings menus that never make it into the captions. Asking for the timestamp lets you jump to the moment and check it yourself.

3. Watch only the part you need

PROMPT

```
/watch [link] --start 12:00 --end 18:30 Summarize just this section and list every tool or setting shown on screen.
```

The `--start` and `--end` flags matter for two reasons: long videos get sampled thinly past about 10 minutes, and watching costs real usage. Fencing the section gets you full detail on the six minutes you care about instead of a skim of forty.

4. Decide if a video is worth your time at all

PROMPT

```
/watch [link] In three sentences: what does this video actually teach, who is it for, and is there anything in it that isn't in every other video on this topic?
```

Run this before committing twenty minutes to anything. The last clause forces an honest answer about whether the video has something new or is the same advice re-recorded.

One honest note on cost: watching burns more of your Claude usage than chatting, because every frame it reads counts. A 5-minute video is nothing. A 40-minute video on a free plan will eat a real chunk of your session, so on long videos, use `--start` and `--end` to watch only the part you need.

PART 3 Turn the Video Into a Skill You Reuse

This is the payoff from the video, so here's the whole craft of it, not just the prompt.

The prompt, and what each line is doing

PROMPT

```
Watch this. Pull out the exact method it teaches, step by step, and turn it into a skill I can reuse. Include the steps in order, the rules it follows, the mistakes it warns about, and the checks it uses. Give the skill a clear one-word name and package it as a single .skill file I can save.
```

- "The exact method, step by step" forces Claude to extract the system instead of summarizing the vibe. This is the line that separates a real skill from a book report.
- "The rules it follows, the mistakes it warns about, the checks it uses" pulls the three things a tutorial buries in asides, and they become the skill's guardrails.
- "A clear one-word name" is what makes it a command. `copy`, `ads`, `mealplan`.
- "Package it as a single .skill file" gets you one saveable file instead of loose text.

What a good result looks like

When I ran this on the copywriting video, the skill that came back had the video's four checks built in as rules it runs before answering, a scoring mode where I can paste my own draft and it grades each rule pass, weak, or fail with a specific fix, a practice mode with the video's drills, and guardrails it added on its own, like refusing to invent statistics or testimonials. That's the bar. A skill should come back knowing the method, not just remembering the video.

Then give it a real job

This is the run from my video. One prompt, a made-up business, everything the skill needs to know in five sentences:

PROMPT

Write my welcome email sequence for Muddy Paws Mobile Grooming. I'm a one-woman mobile dog groomer in Charlotte. I drive a converted van to people's driveways and groom one dog at a time. No cages, no waiting room, no other dogs barking. My people are busy dog owners whose dogs hate the car and hate the shop. Five emails. I want them to book their first appointment.

What came back was a five-email sequence with a send schedule (day 0, 1, 3, 5, 7), and every email built with good copywriting, because that's what this is all about.

- **The subject line sells the benefit.** "your dog never has to get in the car," plus four variants, each labeled by the angle it plays (curiosity gap, question hook, lowercase personal).
- **The body talks like a person.** It opens with the dog's problem ("Most dogs figure it out by the third turn. The car means the shop."), sells the outcome ("Your dog walks out your front door, gets groomed, and walks back in. That's the whole trip."), and ends with an open loop into the next email.
- **It polices itself.** Two links max per email, and a compliance note under every one: "'no cages, no other dogs' is a factual claim, keep it true on every appointment," and "don't say 'only 3 spots left' unless you actually count and it's actually 3."

- **It even handled the setup.** Send from your own domain, use a real reply-to inbox, and pull anyone who books out of the sequence immediately.

Notice what made that work: the prompt gave it a who, a what, a customer, and a goal. The skill brought the method. Swap in your business and your goal, and that's the whole trick.

The three-question test before you save it

1. **Is the video's actual method in there?** Open the skill and look for the named steps and rules from the video. If you see generic advice that could have come from anywhere, say: "Use the specific frameworks from the video, not general advice," and it rebuilds.
2. **Does it know when to run?** Ask Claude: "When would you use this skill?" The answer should match the jobs you want it for. If it's too broad or too narrow, say: "Narrow the description so it triggers on [the jobs you want]."
3. **Does it fit your situation?** Run it once on something real. If the method assumes products and you sell services, or assumes a big email list and yours is small, tell it exactly that: "This assumes [X], I have [Y], update the skill." Skills are editable text, not sealed apps.

Save it. Claude hands you the skill with a **Save skill** button right on it. Click it, and the skill installs to your account on the spot. (If you don't see the button, add "package it as a single .skill file I can upload" to your request and install it the way you installed watch: Settings, Capabilities, Skills, +.) From then on it runs on request.

Make it smarter later. The skill isn't frozen. Watch a second video on the same topic next month and say: "Watch this and merge what it adds into my copy skill. Keep everything that's already there." One skill, growing, instead of twelve overlapping ones.

IN PRACTICE

Feed it a 15-minute copywriting video, one that teaches a real system, and the file that comes back installs as /copy. Then /copy writes your emails, your ads, and your newsletter with good copywriting baked in: benefit-first lines, structure that pulls people through, the fluff already cut. One skill, three jobs you do every week. The tutorial ran once. The skill runs forever.

Five videos worth feeding it first

1. A copywriting system video, because /copy then covers emails, ads, proposals, and your newsletter in one skill. Pick a method with steps, not a tips list.
2. A tutorial for a tool you already pay for (Canva, QuickBooks, your email platform), so the how-to lives in Claude instead of in your bookmarks.
3. A talk by someone whose writing or sales style you want to borrow, turned into a checklist skill.
4. A process video from your own industry, the thing you'd otherwise take notes on.
5. Any video you keep meaning to "watch later." Now later takes one line.

PART 4 When It Doesn't Work

- **The video has no captions and the transcript comes back empty.** The skill can transcribe the audio itself, but that one feature needs an API key from Groq, which has a free tier (the skill tells you exactly what to do when it happens). Most YouTube videos have captions, so most people never see this.
- **A long video comes back thin.** Past about 10 minutes, the skill samples frames to keep costs down. Ask again with `--start` and `--end` around the section you care about, and it goes deep on just that part.

- **The new skill doesn't show up.** Attaching the file to a chat message is not installing it. Use the **Save skill** button when Claude offers one, or install the file under Settings, Capabilities, Skills, and make sure Code execution and file creation is on.
- **It works in the video but not for your case.** Tell the skill what went wrong in plain words ("this welcome sequence assumes an email list, I'm starting from zero, update the skill") and Claude rewrites it. Skills are editable text, not sealed apps.

DO THIS TODAY

Install the skill (Way 1 or Way 2, ten minutes at most). Then pick one video you've been meaning to watch, run prompt 1 on it, and turn it into your first skill before you close the laptop. One video, one skill, one command you'll still be using next month. That's the whole assignment.

Want *more* like this?

Free AI guides added regularly. No jargon, no overwhelm.

kellystrattonai.com

Instagram / TikTok: @kellystratton.ai | Facebook / YouTube: @kellystrattonai