Claude Video Editing Prompt Pack

The three prompts that turn a folder on your laptop into a video editor Claude Code can run.

Why I made this

Everybody is editing videos with Claude now, and most of them put the setup behind a course. There is nothing in it worth gating. It is one folder, five free tools and three prompts, and the whole thing takes about a minute to stand up.

The reason the obvious version fails is that people open Claude, paste in a video and ask for an edit. Claude cannot see the file, does not have the tools, and has no idea what you want the result to look like. So it writes you a description of an edit instead of making one.

This pack fixes those three gaps in order. Prompt one gives Claude the tools. Prompt two gives it your taste. Prompt three gives it the footage. After that you talk to it in plain words like you would to an editor, and the same folder handles every video you make after this one.

01

How it actually works

1. You make a folder called AI Video Editor with three subfolders inside it, raw footage, references and exports. You start a new chat in the Claude desktop app, choose local, and point it at that folder. Claude can now read, write and run things inside it, and nowhere else.
2. Prompt one sets up the five tools Claude needs: Remotion builds the video in code, HyperFrames does the motion graphics, FFmpeg cuts and exports, Whisper transcribes and times the captions, and yt-dlp downloads reference videos. You install none of it by hand, you paste the prompt and Claude does it.
3. Prompt two hands over your taste. You give Claude three to five videos in the style you want and it writes a style sheet. Prompt three takes your raw file, cuts the dead air, adds the captions and the music, and lays the result out for review before anything is final.

02

The one thing almost everyone gets wrong

The taste step is the one people skip, and skipping it is why most first edits come back looking generic. Tools are the easy half. Claude will happily cut your video the moment FFmpeg and Whisper are in, and the cut will be technically fine and completely characterless, because nothing in the setup ever told it what good looks like to you.

The fix is three to five reference videos and one line of wording. Pick videos whose pacing and caption style you want your own footage to come out looking like, which is usually not the same as your favourite videos. Then tell Claude to get inspired by them rather than copy them.

WATCH OUT

Leave out the words 'get inspired by these, do not copy them' and Claude imitates whichever reference it looked at last, shot for shot. With those words in, it pulls out the pattern across all of them: how long a shot holds, where captions sit, how loud the music sits under the voice. One sentence is the difference between a style sheet and a knock-off.

03

The three prompts

PROMPT 1: GIVE CLAUDE THE TOOLS

```
You are setting up a video editing workspace in this folder. Get everything in place
that is needed to edit video from here, then tell me what you did.

Set up these five tools:
- Remotion, to build the video itself in code
- HyperFrames, for motion graphics and on-screen animation
- FFmpeg, for cutting, trimming and exporting
- Whisper, to transcribe the audio and time captions to the words
- yt-dlp, to pull reference videos down into the references folder

The folder you are working in:

AI Video Editor/
  raw footage/
  references/
  exports/

Rules:
- Check what is already on this machine before you install anything, and skip whatever
is already there.
- Keep every file inside this folder. Do not write anywhere else on my system.
- When you are done, run one small test per tool and show me the output of each.
- If something fails to install, say so plainly and tell me what you need from me. Do
not quietly work around it.
- Write what you set up, and the version of each tool, into a file called tools.md in
the root of this folder.
```

PROMPT 2: TEACH IT YOUR TASTE

```
Here are the videos whose style I want my own videos to end up with:

[LINK 1]
[LINK 2]
[LINK 3]

Download each one into the references folder with yt-dlp, watch them, and write me a
style sheet called taste.md in the root of this folder.

For each reference, write down:
- how long a shot holds before it cuts
- how pauses and dead air are handled, what gets cut and what gets left in
- the caption style: where on screen, how heavy the font, how many words show at once,
what colour
- how the music sits against the voice, where it comes in, where it drops out
- the pacing of the first five seconds

Then write one combined style sheet that describes the pattern across all of them, as
rules I could hand to someone else.

Get inspired by these videos, do not copy them. I want the rules underneath the edit,
not a recreation of any single video. If two references disagree on something, say so
in taste.md, tell me which way you leaned, and why.
```

PROMPT 3: EDIT THE FOOTAGE

```
Read taste.md first and follow it.

My raw file is in the raw footage folder. Turn it into a finished cut:

1. Transcribe the audio with Whisper.
2. Cut the dead air, the long pauses, the restarts and the false starts. Leave the
breathing room taste.md asks for. Do not cut it to the bone.
3. Lay word-timed captions over the video in the caption style taste.md describes.
4. Put a music bed under the voice at the level taste.md describes.
5. Render a review copy into the exports folder and tell me the filename.

Before you render anything, show me the cut list: every segment you are keeping with
its timestamps and the first few words spoken in it, and every segment you are dropping
with the reason you dropped it. I will approve it or change it first.

Do not export a final file until I tell you the review copy is right.
```

Run them in that order, once each. Prompt one and prompt two you only ever run again if you change machines or change your style. Prompt three is the one you run for every new video.

Set it up in Claude Code or Codex

1. Download the Claude desktop app from Anthropic and open Claude Code inside it. The desktop app is what lets a chat reach the files on your machine.
2. Make the folder. Call it AI Video Editor and put three subfolders in it, named raw footage, references and exports. Put it somewhere you can find again, your home folder or Documents is fine. Avoid a folder that syncs to iCloud or Dropbox, because renders write large files and a syncing folder will fight you.
3. Start a new chat, choose local, and point it at AI Video Editor. Everything from here happens inside that one folder. If you ever get 'I cannot find that file', it is almost always because the chat was pointed somewhere else.
4. Paste prompt one and let it finish. It will check what is already on your machine, set up what is missing, run a small test per tool and write tools.md. Read the test output before you move on. A tool that failed here will fail later in a way that is harder to read.
5. Paste prompt two with your three to five reference links in it. Claude downloads them into references, watches them and writes taste.md. Open taste.md and read it. If it describes something you do not want, say so now and have Claude rewrite it, because every future edit reads this file.
6. Film something normally and drop the raw file into raw footage. One file, nothing renamed, straight off the phone or camera.
7. Paste prompt three. Claude transcribes, builds a cut list and shows it to you before rendering. Approve it or change it, then let it render the review copy into exports.
8. On Codex, the only difference is where the instructions live. Put the same three prompts in a file in the project folder and add a pointer to them in AGENTS.md so they load with the session, or paste them in directly the same way. The folder layout, the tools and the wording do not change.

There is no API key in this setup. Remotion, HyperFrames, FFmpeg, Whisper and yt-dlp are all free, and Claude is doing the driving, so the only account involved is the one you already have.

> 💡 **QUICK TIPS**
>
> - A successful first run looks like this: Claude prints a cut list with timestamps, you approve it, and a file appears in exports a few minutes later with captions burned in and music underneath. If you get a written description of an edit instead of a file, the chat is not pointed at the folder.
> - Keep tools.md and taste.md in the root of the folder, not in a subfolder. Claude reads the root first and these two files are what make every later edit consistent.
> - The first render is the slow one. Whisper has to transcribe the whole file before anything else can happen, so give a long recording a few minutes before you assume it has stalled.

04

The lines I actually say after the first cut

> **FIX THE CAPTION PLACEMENT**
>
> `The captions are sitting too high and they are covering my face. Move them down to [POSITION, e.g. the lower third] and show [NUMBER] words at a time instead. Update taste.md so you do it this way by default from now on.`

SAVE A PAUSE YOU CUT

```
The pause at [TIMESTAMP] is a beat, not dead air. Put it back and leave it full length. Keep every other cut exactly as it is and show me the updated cut list before you re-render.
```

FIX THE MUSIC LEVEL

```
The music is burying my voice from [START] to [END]. Duck it under the dialogue and bring it back up in the gaps between sentences. Tell me what level you set it to.
```

ADD A REFERENCE AND RETRAIN THE TASTE

```
Add these references and rewrite taste.md: [LINK], [LINK]. Get inspired by them, do not copy them. Then tell me what changed in the style sheet and which old rule each change replaced.
```

EXPORT THE APPROVED CUT

```
The review copy is right. Render the final to exports as [FILENAME] at [RESOLUTION] in [ASPECT RATIO] for [PLATFORM]. Do not change a single cut or caption from the version I just approved.
```

When it breaks

ERROR	FIX
Claude says it cannot find the file, or describes an edit instead of making one	The chat is not pointed at the folder. Start a new chat, choose local, pick AI Video Editor again. Claude only sees inside the folder you hand it.
A tool fails during prompt one	Read which one. Ask Claude to show the exact command it ran and the error it got, then fix that one tool before running prompt two. Running the edit prompt on a broken install fails in a much less readable way.
Captions are out of sync with the audio	Whisper transcribed the raw file, not the cut. Ask Claude to transcribe the edited file and re-time the captions against that version.
Every edit comes out looking the same no matter what references you give it	taste.md was never written, or it is not being read. Open it. If it is thin or generic, give Claude better references and tell it to rewrite the sheet, then re-run prompt three.
The cut feels rushed and clipped	Claude cut on silence alone. Tell it which pauses are beats, name the timestamps, and ask it to re-cut from the same cut list rather than starting over.
yt-dlp cannot download a reference link	That video is not downloadable. Swap in a different reference, or describe the style in plain words and have Claude write it into taste.md by hand.
The export has video but no audio	The audio stream was dropped when the segments were joined. Ask Claude to show you the FFmpeg command it ran and re-run it with the audio mapped.
Claude exported a final file before you approved anything	The review step got skipped. Put the line 'do not export a final file until I tell you the review copy is right' back at the end of prompt three and run it again.

Ready to go deeper?

Want the videos made for you? Paste any link or idea into Content Engine and get a published video in under 3 minutes, with DM automation armed to capture leads.

Meet Content Engine →

https://builds.digicuratoragency.com/content-engine/