

Montare video con Claude Code: guida al setup

File da mandare in DM · chi commenta "video" · BIG BRAIN

Questa è la guida per replicare il sistema con cui ho montato il mio primo video parlando con Claude Code, da terminale. Droppi il girato in una cartella, dici a Claude cosa vuoi (a voce o scritto), e ottieni il video montato: filler tolti, audio ricucito, sottotitoli, slide in sovrapposizione.

Il motore è **video-use**, un progetto open source di browser-use. Tempo di setup: circa 15 minuti.

1. COSA TI SERVE

COSA	A COSA SERVE	COME
Claude Code	l'agente che pilota il montaggio	già installato, lavori da terminale
Homebrew	per installare il resto (su Mac)	brew.sh
FFmpeg	il render vero e proprio	<code>brew install ffmpeg</code>
uv	gestisce Python per video-use	<code>brew install uv</code>
API key ElevenLabs	trascrive l'audio del tuo video	elevenlabs.io, vedi punto 4

Niente altro per partire. (Per le animazioni/motion graphics serve un pezzo in più, ma non per pulire e pubblicare un talking head: parti da qui.)

2. INSTALLA VIDEO-USE (LA VIA FACILE)

Apri Claude Code in una cartella qualsiasi e incolla questo prompt:

```
Set up https://github.com/browser-use/video-use for me.

Read install.md first to install this repo, wire up ffmpeg, register the skill
with whichever agent you're running under, and set up the ElevenLabs API key
(ask me to paste it when you need it). Then read SKILL.md for daily usage, and
always read helpers/ because that's where the editing scripts live. After
install, don't transcribe anything on your own. Just tell me it's ready and
wait for me to drop footage into a folder.
```

Claude fa tutto da solo: clona il repo, lo registra come skill, collega ffmpeg, e a un certo punto ti chiede la API key di ElevenLabs.

Se preferisci farlo a mano

```
git clone https://github.com/browser-use/video-use ~/Developer/video-use
ln -sf ~/Developer/video-use ~/.claude/skills/video-use
cd ~/Developer/video-use
uv sync
cp .env.example .env          # poi apri .env e metti la tua key
```

3. DUE TRUCCHI CHE MI HANNO FATTO RISPARMIARE ORE

Questi due punti sono dove ci si blocca. Risolvili in anticipo:

1. **La versione di Python lasciala gestire a uv.** Non installarlo a mano da Homebrew: quello di sistema è spesso troppo vecchio, l'ultimo di Homebrew troppo nuovo per alcune librerie audio. `uv sync` scarica da solo la versione giusta (3.12) e crea l'ambiente. Funziona e basta.
2. **La key di ElevenLabs deve avere il permesso giusto.** Se ti dà errore `401 missing_permissions` non è la key sbagliata: è solo che manca un permesso. Vedi punto 4.

4. LA KEY DI ELEVENLABS

video-use non "guarda" il video, lo **legge**: manda l'audio a ElevenLabs e ottiene la trascrizione con i tempi parola per parola. È così che capisce dove tagliare.

1. Vai su **elevenlabs.io**, crea un account.
2. Profilo → **API Keys** → crea una nuova key.
3. **Importante:** alla key dai il permesso **Speech to Text = Access**. Senza quel permesso la trascrizione fallisce con `401`.
4. Incolla la key quando Claude te la chiede (oppure mettila nel file `.env` del repo video-use, riga `ELEVENLABS_API_KEY=...`).

La key è l'unico segreto. Non metterla mai dentro un repo pubblico o su GitHub: vive solo nel `.env` locale.

5. PRIMO MONTAGGIO

1. Crea una cartella per il progetto e metti dentro il girato grezzo.
2. Da lì apri Claude Code (`claude`) e di' cosa vuoi, ad esempio: *"monta questo in un video pulito, toglie le esitazioni e mettimi i sottotitoli"*.
3. Claude inventa il girato, ti **propone una strategia** e aspetta il tuo OK prima di toccare qualcosa.

4. Approvi, lui monta, e si auto-controlla l'output a ogni taglio prima di mostrartelo.

5. Trovi il risultato in `final.mp4` accanto al girato.

Da qui itera a voce: "il taglio al minuto 3 è brusco", "i sottotitoli falli più piccoli", "fai entrare la slide qui". È una conversazione, non una timeline.

*Consiglio: io ho dato le indicazioni **a voce** (con Wispr Flow) mentre facevo altro. Parlare al posto di cliccare cambia completamente il ritmo.*

6. CREA LA TUA SKILL DI STILE (IL PASSO CHE FA LA DIFFERENZA)

Il primo montaggio è il più lungo, perché stai decidendo il tuo stile: dimensione dei sottotitoli, come entrano le slide, il ritmo dei tagli.

Quando sei soddisfatto, chiedi a Claude di **trasformare quelle scelte in una skill**. Qualcosa tipo:

Crea una skill che cattura lo stile di montaggio che abbiamo definito qui: sottotitoli, slide in sovrapposizione, ritmo dei tagli, i parametri che hanno funzionato. La prossima volta voglio ripartire da questo, non da zero.

Dalla seconda volta in poi gran parte del lavoro è già fatto, e ti concentri solo sul migliorare qualità ed effetti. È esattamente il salto che ho fatto io.

IN BREVE

1. `brew install ffmpeg uv`
2. Incolla il prompt di setup in Claude Code
3. Crea la key ElevenLabs col permesso Speech to Text
4. Metti il girato in una cartella, apri Claude, di' cosa vuoi
5. Quando trovi il tuo stile, fattelo salvare in una skill

Buon montaggio. Se ti blocchi, scrivimi.