// FREE CLAUDE SKILL · 5 SECTIONS · BY JOSH POCOCK

Video Jury

7 honest judges on your video before you post it

A free Claude skill. Give it a video and seven separate Claude judges each check one thing, from the first 3 seconds to the cover, then hand you one verdict and up to five fixes.

TERMINAL · BOOT

```
# v1.0 · install once, then run it in any session
$ git clone https://github.com/joshpocock/video-jury ~/.claude/skills/video-jury
✓ installed to ~/.claude/skills/video-jury
$ claude
> /video-jury
```

Get the skill on GitHub →

Read it on the web: strideaiacademy.com/r/video-jury

It never posts, edits or deletes anything. The jury only reads. You decide.



SECTION 01

The one prompt

One line. Paste this into Claude Code and it does the rest.

PROMPT

```
Install the Video Jury skill: clone https://github.com/joshpocock/video-jury into
~/.claude/skills/video-jury, check that ffmpeg is installed, then confirm /video-jury works.
```

01
Copy the prompt above
The repo link plus one line

→

02
Paste it into Claude Code
Claude installs the skill

→

03
`/video-jury my-video.mp4`
Add the cover and caption if you have them

→

04
Read the verdict
One decision, up to five fixes

By hand

>_ TERMINAL

```
git clone https://github.com/joshpocock/video-jury ~/.claude/skills/video-jury
```

Start a new Claude Code session and type `/video-jury` followed by the path to your video.

You also need ffmpeg (free)

ffmpeg is the free tool that pulls the frames and measures the sound. Pick the line for your computer, then open a new terminal.

>_ TERMINAL

```
winget install ffmpeg          # Windows
brew install ffmpeg            # Mac
sudo apt install ffmpeg        # Linux
```

It is built and tested in Claude Code, where the seven judges run at the same time as separate agents. It has not been tested in the Claude chat app.

SECTION 02

Who is on the jury

Claude cannot play a video. So the skill first turns yours into evidence: frames, contact sheets with the time printed on every tile, the moments the picture changes, loudness numbers and the transcript. Then seven judges each read that evidence alone. None of them sees your notes or each other.

JUDGE	THE ONE JOB
Hook	The first 3 seconds. Does the first frame say what this is?
Pacing	Where it sits still, where it rushes, where attention drops
Captions	Every word on screen: readable, spelled right, matching the voice
Sound	Loudness, dead air and peaks, read from meters
Proof	Is every claim backed by what is on screen when you say it?
Thumbnail	The cover at the size it shows in the grid
Scroller	A stranger with a thumb, who says the second they would swipe

Each judge votes **POST**, **FIX** or **HOLD**, gives a score out of 10, and names at most three problems. Every problem comes with the second it happens and the file the judge saw it in. A script counts the votes. The video passes on 5 or more POST votes with no HOLD from the hook judge or the scroller.

SECTION 03

One real run

This is a real run on one of my own Reels, 54 seconds long. The evidence took 11 seconds to build. The last judge answered 96 seconds after the seven started.

PROMPT

```
/video-jury renders/my-reel.mp4
```

>_ OUTPUT

```
# Verdict: FIX FIRST

only 1 of 7 voted POST. POST 1, FIX 6, HOLD 0.

hook       FIX   7
pacing     FIX   7
captions   FIX   6
sound      POST  7
proof      FIX   6
thumbnail  FIX   7
scroller   FIX   6
```

Three of the fixes it handed back, in the judges' own words:

- **2.5 s, captions.** "The chunks break mid-phrase, so some chips read as typos. At 2.5 s 'ONE A PERSONAL' lands inside the hook."
- **24.5 s, proof.** "While he says 'only sends to addresses that are deliverable' … at 32.0 s the Outbox shows an email queued to an address marked 'risky'."
- **17 s, scroller.** "Around 17 s … I have the gist and nothing new is on screen, so I swipe."

The first two are plain to see once you open the frames at those seconds. Nobody had caught them before the jury did.

SECTION 04

Read the notes, not the count

I ran the full jury three times on that same video with nothing changed. The votes moved every time.

RUN	POST	FIX
1	2	5
2	4	3
3	1	6

The proof judge caught the "risky" address in 4 of its 6 passes. So one run is a strong second opinion, and it will miss things. The problems that come back in a second run are the ones to fix first. Ask for a second run and the skill lists them:

PROMPT

```
Run the jury again and show me which notes came back both times.
```

SECTION 05

The fine print

It never posts anything. It does not upload, schedule, edit, trim or delete. It reads your video and writes a verdict file.

No judge can hear your video. Sound is judged from loudness numbers and pictures of the audio. Tone of voice, music taste and mispronounced words are out of reach, and the judges say so.

It does not predict views. It finds the problems a careful stranger would find.

It is heavy on usage. Seven judges reading frames used between 800,000 and 900,000 tokens per run on a one-minute video in my tests. A second run doubles it.

The pass mark is my judgment call. Five POST votes out of seven is a line I picked. It is one number in the script; change it if you disagree.

Your files stay on your computer. The evidence and the verdicts are written to a `video-jury` folder where you run it.

Free and MIT licensed. No signup, no API key, nothing to connect.

Want to build skills like this?

Stride AI Academy is where I teach people to build Claude skills and automations like this one, then sell and deliver them. Four courses and weekly live calls.

See the Academy →

THE ACADEMY	strideaiacademy.com Courses, weekly live calls, and the build-it-with-me community.
THE SKILL	github.com/joshpocock/video-jury MIT licensed. Star it if it saved you.
THIS GUIDE, LIVE	strideaiacademy.com/r/video-jury Copy buttons on every prompt, always the latest version.
EVERY FREE SKILL	Email me every free skill Optional. The library of free skills and guides, by email.

STRIDE