

Edição de vídeo com IA — o guia completo

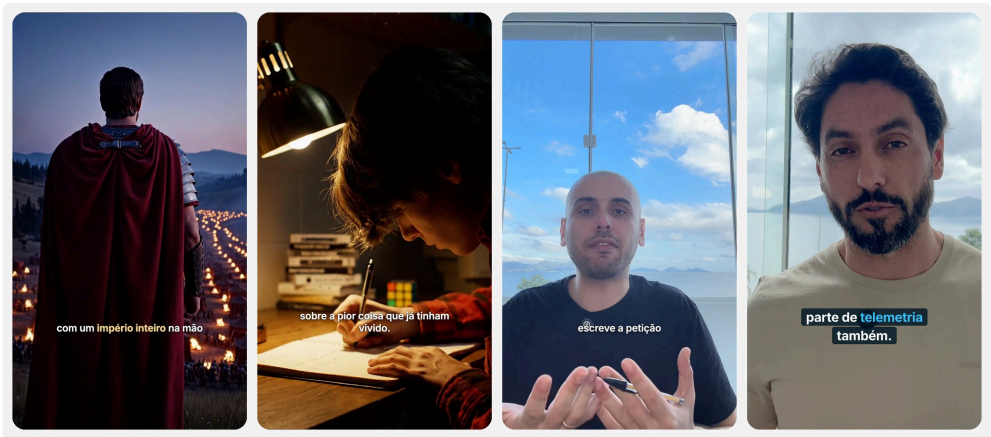
Este documento foi escrito pra ser lido por você e **pela sua IA**. Você não precisa entender cada comando: arraste este PDF pro Claude Code (ou pro Codex, do ChatGPT) e diga:

Leia este guia inteiro. Quero fazer o Tipo [1, 2 ou 3] com o vídeo que vou te mandar. Instale o que faltar, siga os passos e me mostre o resultado de cada etapa antes de seguir.

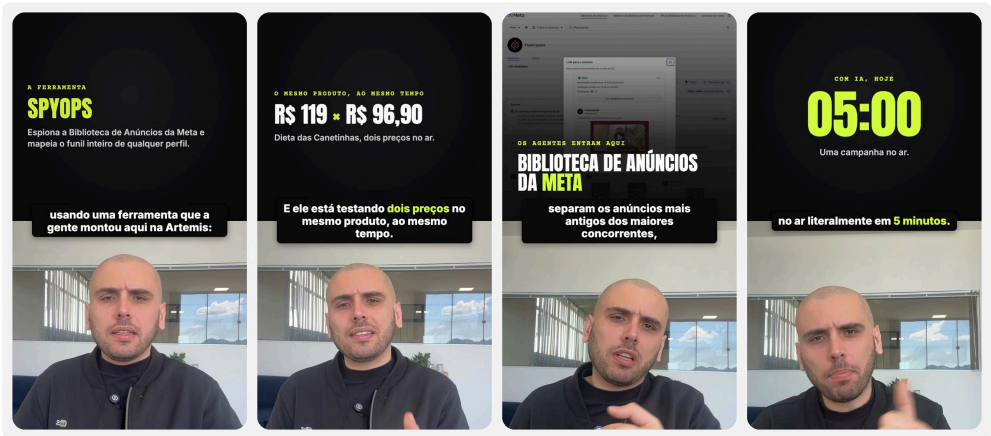
A IA instala as ferramentas, roda os comandos, confere o resultado e te mostra. Você grava, aprova e posta. Ninguém abre editor de vídeo. As ferramentas citadas aqui são as que usamos. Se a sua IA sugerir outra que faz o mesmo, pode trocar: o que importa é o fluxo.

Os três tipos de vídeo

Tipo 1 — Fala com cobertura. Você fala pra câmera. Entram legenda discreta, cenas geradas por IA ilustrando o que você diz e zoom seco no rosto.



Tipo 2 — Tela dividida. Prova em cima (anúncio, página, sistema funcionando, número), você embaixo, legenda na emenda.



Tipo 3 — Motion graphics (Remotion). Ninguém grava nada. O vídeo inteiro é desenhado em código: texto animado, fotos, números, gráficos, com ou sem locução.

Instalação (uma vez só)

Você precisa de:

1. **Claude Code** (app do Claude → aba Code, plano Pro ou Max) **ou Codex** (ChatGPT Plus ou superior). É a IA que roda tudo no seu computador.
2. **ffmpeg** — corta, junta, sobrepõe e exporta vídeo.
3. **Whisper** (whisper.cpp) — transcreve sua fala no seu computador, com o tempo de cada palavra. Grátis.
4. **Node.js** (versão 20 ou mais nova) e **Python 3**.
5. **Puppeteer** — desenha legenda e cartões em HTML e transforma em imagem.
6. **Remotion** — pra o Tipo 3. Instala por projeto, a IA faz isso na hora.
7. **Gerador de vídeo por IA** — pra as cenas de cobertura do Tipo 1. Usamos o **Veo 3.1** (Google AI Studio, aistudio.google.com, pago por clipe). Alternativas: Kling, Runway, Sora. Sem orçamento? Vídeo grátis do Pexels (pexels.com/api).

Cole na sua IA:

Prepare meu computador para editar vídeo com IA:

1. Confira se tenho ffmpeg, Node.js 20+, Python 3 e git. Instale o que faltar (Mac: Homebrew; Windows: winget).
2. Instale o whisper.cpp e baixe o modelo ggml-small.bin (serve pra português). Confirme que o comando whisper-cli funciona.
3. Crie a pasta ~/videos-ia com as subpastas bruto, broll, cards, saida.
4. Dentro de ~/videos-ia rode npm init -y e npm install puppeteer.
5. Me diga o que foi instalado e o que falhou.

Chave do Veo (opcional, só pro Tipo 1 com cena gerada): crie a chave em aistudio.google.com → Get API key e ative o faturamento. Depois cole na IA:

Crie o arquivo ~/videos-ia/.env com a linha GEMINI_API_KEY=COLE_SUA_CHAVE_AQUI e nunca mostre essa chave no chat.

Antes de editar: escolha um molde

Toda edição nossa parte de um reel que já funcionou. Nós temos um **banco de reels de exemplo**, capturado pelo **SpyOps**, nossa ferramenta de espionagem e modelagem. **Você não precisa dele pra seguir este guia** (ele é vendido à parte, no fim do documento). Sem ele, salve no computador um reel que você admira e mande pra IA:

Este é o meu molde. Extraia um quadro a cada 1 segundo e monte uma folha de quadros (ffmpeg -vf "fps=1,scale=270:-1,tile=6x3"). Olhe a folha e me diga: onde fica a legenda e com quantas palavras, de quantos em quantos segundos entra uma cena por cima, se tem zoom no rosto, que tipo de cartão aparece. Use isso como regra da minha edição.

Tipo 1 — Fala com cobertura

Gravação: celular na vertical, 1080×1920, você falando o roteiro. Luz de frente, um vídeo por arquivo.

Passo 1 — Transcrever com o tempo de cada palavra

```
ffmpeg -i bruto/v1.mp4 -ar 16000 -ac 1 v1.wav  
whisper-cli -m ggml-small.bin -l pt -f v1.wav -ml 1 -ojf -of palavras
```

Sai o `palavras.json`, com o início e o fim de cada palavra. A IA agrupa em blocos de 2 a 4 palavras, no formato `[[início, fim, "texto"], ...]`, e salva em `legendas.json`. **Revise o texto:** o Whisper erra nome próprio e termo técnico. A legenda sai do texto corrigido.

Passo 2 — Marcar onde entra cobertura

A IA lê a transcrição e escolhe de 6 a 10 trechos que pedem imagem: quando você cita um lugar, uma pessoa, um objeto, uma cena. Regra prática: uma cena a cada 5 a 8 segundos, de 1,5 a 2,5 segundos cada. O primeiro trecho entra nos 2 primeiros segundos, pra segurar a atenção.

Passo 3 — Gerar as cenas

Um prompt em inglês por trecho, com linguagem de cinema e sem nada escrito na tela. Exemplo:

```
Cinematic slow push-in on an ancient marble bust in a dark museum, single soft side light, dust particles in the light beam, film grain, 35mm, shallow depth of field.
```

Prompt negativo de sempre: `text, letters, subtitles, captions, logos, watermark, music`.

Chamada do Veo pela API (a IA escreve o script):

- POST https://generativelanguage.googleapis.com/v1beta/models/veo-3.1-generate-preview:predictLongRunning?key=SUA_CHAVE
- corpo: `{"instances":[{"prompt":"..."}],"parameters":{"aspectRatio":"9:16","durationSeconds":8,"negativePrompt":"..."}}`
- a resposta é uma operação; consultar `GET /v1beta/<nome-da-operação>` a cada 10 segundos até `done: true`, pegar a primeira `uri` de vídeo e baixar com `?key=SUA_CHAVE`.
- cerca de 1 minuto por clipe. Gere todos em sequência, em segundo plano, e tente de novo até 3 vezes se um falhar.

Passo 4 — Preparar cada clipe

O Veo às vezes devolve tarja preta em cima e embaixo, e sai em 24 fps. Pra cada clipe:

```
ffmpeg -i broll/c1.mp4 -vf cropdetect=limit=24:round=2 -f null -  
ffmpeg -i broll/c1.mp4 -an -vf  
"crop=W:H:X:Y,fps=30,scale=1080:1920:force_original_aspect_ratio=increase,crop=1080:1920,noise=all=4:allf=t,format=yuv420p"  
-c:v libx264 -crf 14 broll/c1-ok.mp4
```

O primeiro comando acha o recorte (`crop=W:H:X:Y`); o segundo aplica, passa pra 30 fps, preenche o quadro vertical e põe um grão leve pra a cena casar com o vídeo do celular.

Passo 5 — Legenda e cartões

Cada bloco de legenda vira um PNG transparente de 1080×1920, desenhado em HTML e fotografado pelo Puppeteer (`page.screenshot({omitBackground: true})`). Estilo que funciona:

- fonte sem serifa, 46 a 52 px, branca, com sombra forte (`text-shadow: 0 2px 14px rgba(0,0,0,.75)`);
- posição a uns 62% da altura, fora do rosto;
- a palavra-chave do bloco em cor de destaque.

Os PNGs viram uma faixa de vídeo com transparência: um arquivo `legendas.txt` lista cada PNG com a duração dele, e

```
ffmpeg -f concat -safe 0 -i legendas.txt -c:v prores_ks -pix_fmt yuva444p10le legendas.mov
```

Cartão (número grande, capa de livro, frase de destaque) segue o mesmo caminho: HTML → PNG 1080×1920.

Passo 6 — Montar

Um único comando de ffmpeg com `filter_complex`, em camadas:

1. o seu vídeo de base;

2. **zoom seco no rosto:** uma cópia ampliada 12% (`crop=iw/1.12:ih/1.12,scale=1080:1920`) ligada só em alguns trechos, com `overlay=enable='between(t,17.5,20.5)+between(t,24.2,28.8)'` . O corte seco faz o papel do jump cut;
3. **as cenas e os cartões** por cima, cada um no seu tempo: `trim=start=0.5:duration=2.2,setpts=PTS-STARTPTS+INICIO/TB` e `overlay=enable='between(t,INICIO,FIM)'` ;
4. **a legenda** (`legendas.mov`) por cima de tudo;
5. **o áudio original**, copiado sem mexer (`-map 0:a -c:a aac`).

Peça à IA pra guardar a lista de cenas num script (`montar.py`), com uma linha por cena: início, fim, arquivo. Mudar a edição vira editar uma lista.

Tipo 2 — Tela dividida

Mesma transcrição e mesma legenda do Tipo 1. Muda o layout:

- **Embaixo (1080×960):** o seu vídeo, recortado pra mostrar do peito pra cima: `crop=1080:960:0:Y`, com o Y escolhido olhando um quadro.
- **Em cima (1080×960):** a prova, em sequência contínua. Cada trecho é um arquivo de vídeo cortado no tempo certo e depois emendado com `ffmpeg -f concat`.
- **Legenda:** em pílula escura bem na emenda das duas metades.
- **Juntar:** `[topo][baixo]vstack`, legenda por cima, áudio original.

De onde vem a prova (sempre real):

- **Print de página:** um script Puppeteer abre a página no tamanho de celular (390×844), tira o print da página inteira e a IA faz ela "rolar" no vídeo. O script só tira print: não clica, não preenche formulário.
 - **Anúncio de concorrente:** Biblioteca de Anúncios da Meta. Com o SpyOps, vêm baixados com vídeo, copy e data de início. Sem ele, print da página pública do anúncio.
 - **Vídeo vertical na metade de cima:** o próprio vídeo ampliado e borrado no fundo (`boxblur=24:2`) e o vídeo inteiro centralizado na frente.
 - **Cartão parado:** zoom lento pra não ficar estático: `zoompan=z='1+0.035*on/N'`. Amplie a imagem 2x antes do zoompan pra não tremer.
-

Tipo 3 — Motion graphics em Remotion

Remotion é uma biblioteca que transforma código React em vídeo. Cada cena é um componente, cada animação é uma conta em cima do número do quadro. Como é código, a IA escreve, ajusta e renderiza sozinha. É o jeito de fazer vídeo de marca sem gravar nada.

Criar o projeto:

```
npx create-video@latest meu-reel
```

Escolha o modelo em branco (Blank) com TypeScript. Depois `npm install` e `npx remotion studio` abre uma prévia no navegador.

Como a IA deve montar (cole junto com o roteiro):

```
Crie em Remotion um reel vertical 1080x1920, 30 fps, com o roteiro abaixo.
- Em src/Root.tsx, registre uma Composition com a duração total em quadros.
- Uma cena por bloco do roteiro, cada uma em um <Sequence from={...} durationInFrames={...}>.
- Animações com useCurrentFrame(), interpolate() e spring(): texto entrando palavra por palavra, números contando, fotos com zoom lento.
- Fotos e locução em public/, carregadas com staticFile(). Locução com <Audio>.
- Fontes com @remotion/google-fonts.
- Paleta da minha marca: [cores]. Fonte: [fonte].
- Rode npx tsc --noEmit antes de renderizar.
- Renderize com: npx remotion render src/index.ts MeuReel saida/reel.mp4 --codec=h264 --crf=18
- Tire um quadro a cada 2 segundos e me mostre numa folha antes de dizer que terminou.
Roteiro: [cole aqui]
```

Regras que aprendemos:

- **Texto curto na tela.** No máximo 12 palavras por cena, de 2 a 4 segundos. Quem lê no celular desiste de bloco grande.
 - **Foto real vence ilustração.** Nos reels do cliente náutico, foto do produto e da equipe com texto por cima rendeu mais que animação abstrata.
 - **Locução:** grave a sua (celular perto da boca, ambiente quieto) ou gere com TTS. Use os tempos do Whisper pra a animação entrar junto com a fala.
 - **Formato reportagem:** caixa branca de texto com serifa, local e data no canto, foto inteira atrás. Funciona pra case de cliente.
-

Conferir antes de entregar (vale pros três tipos)

A IA não diz que terminou sem olhar o vídeo:

```
ffmpeg -i saida/reel.mp4 -vf "fps=1/2,scale=216:-1,tile=8x4" -frames:v 1 folha.jpg
```

Ela abre a folha e confere: legenda em cima do rosto, texto cortado na borda, cena com tarja, cartão com erro de digitação. Corrige e monta de novo.

Exportar duas versões:

- **Pra postar:** 1080×1920, H.264, `-crf 18`, áudio AAC 192k, `-movflags +faststart`.
- **Pra ver no celular ou mandar no WhatsApp:** `-vf scale=720:-2 -b:v 2800k`.

O que dá errado

- **O Veo para de gerar (erro 429):** estourou o limite de gasto do projeto no Google AI Studio. Pagar a fatura não basta: suba o limite na página de gastos do projeto.
 - **A legenda some em fundo claro:** sombra mais forte ou pílula escura atrás.
 - **Tarja preta na cena:** rode o Passo 4 de novo. É o `cropdetect` que resolve.
 - **O Whisper errou um nome:** corrija o texto antes de gerar a legenda.
 - **Áudio fora de sincronia:** nunca reencode o áudio junto com os cortes. Copie o áudio do bruto inteiro, numa trilha só.
 - **Vídeo do iPhone com cor lavada:** é HDR. Peça à IA pra converter pra SDR antes de editar (`zscale` + `tonemap`).
 - **Duas montagens ao mesmo tempo corrompem o arquivo:** um vídeo por vez.
-

SpyOps: espione o que já está dando dinheiro



Todo reel e todo anúncio que fazemos começam olhando o que o concorrente já paga pra rodar. Quem faz esse trabalho é o SpyOps.

Você digita o nome do concorrente. Ele entra na Biblioteca de Anúncios da Meta, baixa todos os anúncios ativos (copy, artes e vídeos), transcreve o que é falado, segue os links até o checkout e escreve um relatório com o que vale modelar. Leva de 5 a 15 minutos.

Os reels de tela dividida deste guia saíram dele:



O que ele já nos mostrou:

- **31 anúncios ativos de um único infoprodutor**, com todos os ângulos que ele está testando lado a lado.
- **O mesmo produto vendido a R\$ 119 e a R\$ 96,90 ao mesmo tempo**. Teste de preço que só aparece seguindo o link até o checkout.
- **Um quiz que grava cada resposta**. Quem desiste no meio vira público de remarketing.
- **Anúncios no ar desde 2023**. Ninguém paga três anos de anúncio que não vende. É esse que você modela primeiro.

O que você leva:

- A copy de todos os anúncios ativos do concorrente, em texto.
- As artes e os vídeos baixados, com a transcrição do que é falado.
- O caminho do anúncio até o checkout: página, preço, oferta.
- Um relatório com os anúncios que rodam há mais tempo e o que modelar.
- O seu próprio banco de reels e anúncios de exemplo: o molde que este guia pede.

Roda dentro da IA que você já usa: Claude (Claude Code) ou ChatGPT (Codex). Você pede em português, do mesmo jeito que pediu a edição.

Lê só informação pública. Não faz login em nada e não preenche formulário. Roda no Mac e no Windows.

R\$ 129, pagamento único, com 7 dias de garantia.

Comprar o SpyOps: <https://payfast.greenn.com.br/174701/offer/iaOxUQ>