Eigenflow

GUIDE EIGENFLOW

SupoClip, l'alternative à OpusClip

Notre test sur de vrais rushs

PAR

Abyane Nabeebaccus

INSTAGRAM

@abyane_nb

ÉDITION

novembre 2026

Tu as commenté CLIPS. Le dépôt est ici : https://github.com/FujiwaraChoki/supoclip. On l'a installé sur un Mac et on lui a donné nos propres rushs. Voici comment l'installer, ce qui en est sorti et ce qui nous a déçus.

À savoir avant de lire

- Test du 4 novembre 2026, commit 97def33 du dépôt, sur deux rushs face caméra en français de 4 min 50 et 5 min 48. Aucun clip n'a été publié, donc on ne sait pas s'ils feraient des vues.

- On n'a pas testé de vidéo d'une heure, ni la transcription payante AssemblyAI, qui est pourtant le réglage par défaut du projet, ni un modèle local avec Ollama.

- SupoClip est sous licence AGPL-3.0, avec 2 543 étoiles et 576 forks le 4 novembre. Il existe aussi une version hébergée et payante. Eigenflow n'a aucun lien avec SupoClip ni avec OpusClip.

1. Ce que ça fait

Tu lui donnes une vidéo, en fichier ou par un lien YouTube, Twitch ou Kick. Il la transcrit, une IA choisit des passages et les note sur 100, en quatre parts de 25 points : accroche, attention, utilité et envie de partager. Ensuite il recadre en 9:16, cherche le visage, ajoute un titre et des sous-titres mot à mot.

Sur le prix, OpusClip affiche 29 $ par mois pour son offre Pro et 15 $ pour Starter, avec une offre gratuite limitée. SupoClip en local n'a pas d'abonnement, mais il consomme ta machine et, selon tes réglages, des clés payantes.

2. L'installer

Il te faut Docker et Git. Nous, on a utilisé Docker Desktop sur un Mac, avec 7,75 Go de mémoire pour Docker. L'installation a fait passer le disque de Docker à 25 Go, prévois de la place.

> git clone https://github.com/FujiwaraChoki/supoclip.git

> cd supoclip

> cp .env.example .env

Dans `.env`, voici les réglages qu'on a changés pour un Mac sans carte graphique et une transcription gratuite. La clé Google sert à l'IA qui choisit les clips, et elle est facturée à l'usage. Remplace chaque secret d'exemple par une valeur générée avec `openssl rand -hex 24`.

> BACKEND_PLATFORM=linux/arm64

> BACKEND_CPU_ONLY=true

> TRANSCRIPTION_PROVIDER=whisper

> WHISPER_MODEL_SIZE=small

> TRANSCRIPTION_LANGUAGE=fr

> LLM=google-gla:gemini-3-flash-preview

> GOOGLE_API_KEY=ta_cle

> FRONTEND_PORT=3107

Puis on a lancé seulement les services utiles :

> docker compose up -d --build frontend backend worker ai-worker redis postgres

Ouvre http://localhost:3107, crée un compte local et dépose ta vidéo. Les ports s'ouvrent uniquement sur ta machine. C'est du code tiers : lis le `docker-compose.yml` avant, et ne garde pas ta clé dans le dossier après tes tests.

3. Ce qui en est sorti

Nos essais, avec les temps relevés dans le journal du service :

- Rush paysage de 4 min 50, modèle Whisper `base` : 3 clips notés 82, 74 et 70, en 3 min 32.

- Rush vertical de 5 min 48, modèle `base` : 3 clips notés 91, 82 et 80, en 5 min 36.

- Rush paysage, modèle `small` en français : 2 clips notés 78 et 73, en 2 min 54.

- Rush paysage, modèle `medium` : bloqué. Le service a redémarré environ 40 secondes après le départ et la tâche est restée à 30 %, sans message d'erreur. Le fichier de 1,4 Go du modèle s'était bien téléchargé. On n'a pas établi la cause exacte, le manque de mémoire est notre hypothèse.

Le travail qu'il fait seul est réel : choix des passages, recadrage, titre, sous-titres. En moins de six minutes, on avait des clips exportables sans ouvrir un logiciel de montage.

4. Ce qui nous a déçus

Les sous-titres en français, avec la transcription gratuite. Le modèle `base` a écrit « on les supe pas » pour « on les suit pas », « Clote » pour « Claude » et « renoué » pour « Runway ». Le modèle `small` fait nettement mieux, mais il a encore écrit « Alexandre Mozy » pour « Alex Hormozi ». Relis tout avant de publier : l'éditeur intégré permet de corriger.

Les prises ratées. Un rush brut contient des phrases répétées, et elles restent dans les clips. Donne-lui plutôt une vidéo déjà propre, un podcast ou une interview.

Le cadrage. Le visage est bien suivi sur la plupart des plans, mais un de nos clips garde tout le plan paysage en petit, sur un fond flou.

La note. Sur le même rush, le même sujet est passé de 82 à 78 d'un essai à l'autre, avec d'autres coupes. C'est l'avis de son IA, utile pour trier. Rien ne prouve qu'un clip noté 91 fera plus de vues qu'un autre.

La relance. Quand on a renvoyé exactement le même fichier, il a réutilisé l'analyse précédente et toutes les notes sont tombées à 0. Pour changer de réglage, on a dû modifier les métadonnées du fichier.

5. Ce que ça coûte vraiment

- Logiciel : gratuit en local, sous AGPL-3.0.

- Transcription : gratuite avec Whisper, au prix de la qualité vue plus haut. AssemblyAI est le réglage par défaut du projet et il est payant. On ne l'a pas testé, donc on ne peut pas dire s'il corrige le français.

- IA qui choisit les clips : on a utilisé Gemini sur notre clé prépayée, facturé à l'usage. On n'a pas relevé le montant exact de ces essais. Le dépôt propose aussi un modèle local avec Ollama, qu'on n'a pas essayé.

- Machine : environ 25 Go de disque pour Docker, et un Mac bien occupé pendant le traitement.

6. Ce qu'on ne sait pas

- Comment il tient sur une vidéo d'une heure, et combien de temps ça prend.

- Si la qualité des sous-titres en français est bonne avec AssemblyAI.

- Si ses clips les mieux notés font vraiment plus de vues.

- Ce que donne sa publication directe vers YouTube, TikTok et Instagram : on ne l'a pas branchée.

Si tu veux essayer, commence par une vidéo courte et déjà propre, garde le modèle `small` sur un Mac, et relis chaque sous-titre.