Video VLM Frame-Budget Decision Sheet

Fora Soft Learn · AI for Video Engineering

1 · The cost is two multiplications

A video VLM never sees your video — only sampled frames turned into tokens. Cost = (frames × tokens-per-frame + audio) ÷ 1,000,000 × price-per-million.

```
10-min video @ 1 fps, Gemini default resolution:
  600 frames  x 258 tokens/frame = 154,800 visual tokens
  600 seconds x  32 tokens/second =  19,200 audio tokens   -> ~174,000 input tokens
  174,000 / 1,000,000 x $0.30/M  = ~ $0.052 per video
Low resolution (66 tokens/frame): ~59,000 tokens -> ~ $0.018 per video
```

2 · The context window is a hard wall

A 1-million-token model holds about 1 hour of video at default resolution, or about 3 hours at low resolution (Google Gemini docs, 2026). Beyond that you must lower resolution, lower frame rate, or switch to streaming.

3 · Frame sampling vs token streaming

Frame sampling (offline)

- Whole video sent at once; reasons across all of it.
- Limited by the context window (~1-3 h).
- Pay once per video, up front.
- Best: recorded meetings, lectures, archive review.

Token streaming (online)

- Frames fed continuously; old frames fade.
- Handles endless video; remembers recent past only.
- Pay continuously while the stream runs.
- Best: live cameras, ongoing calls, real-time alerts.

4 · Pre-flight checklist before you build

- [ ] What is the shortest event the feature must catch? If under one second, 1 fps will miss it — raise the frame rate.
- [ ] Does the answer need to arrive live, or later? Live = token streaming; later = frame sampling.
- [ ] Will the whole video fit the context window at your chosen fps and resolution? If not, chunk it or stream it.
- [ ] Default or low media resolution? Low cuts visual tokens ~4x and is fine for slow content (lectures, static cameras).
- [ ] For long, uncurated footage, is adaptive / query-aware sampling worth it? It can add ~8-10 points of accuracy over uniform 1 fps.