

Picker по алгоритмам Optical Flow

Сопоставьте sparsity, latency и accuracy с правильным примитивом — и с per-frame бюджетом compute.

Decision tree — выберите алгоритм

1. Нужен поток на каждом пикселе?

- ☐ Нет (tracking, AR-якоря, odometry) → Lucas-Kanade.
- ☐ Да (stabilization, interpolation, VSR) → дальше.

2. Пайплайн real-time?

- ☐ Да, CPU only → DIS (Kroeger 2016, 300-600 Hz).
- ☐ Да, GPU доступен → SEA-RAFT (20+ FPS @ 1080p).
- ☐ Да, NVIDIA only → NVIDIA Optical Flow SDK.
- ☐ Нет, offline → дальше.

3. Требуемый потолок ассурасу?

- ☐ Стандартный → RAFT (Sintel EPE 2,855; KITTI F1 5,10%).
- ☐ State-of-the-art → FlowFormer / MegaFlow / DPFlow (2026).

4. Требования по лицензии?

- ☐ Commercial OK → все перечисленные BSD/Apache/MIT.
- ☐ NVIDIA hardware lock → Optical Flow SDK привязан к GPU.

Per-frame compute бюджет

Алгоритм	Тип	Железо	Скорость
Lucas-Kanade	Sparse	CPU	~200 FPS @ 720p
Farnebäck	Dense	CPU	~30 FPS @ 720p
DIS	Dense	CPU	300-600 Hz @ SD
RAFT	Dense	GPU	9 FPS @ 1088×436
SEA-RAFT	Dense	GPU	20+ FPS @ 1080p
FlowFormer	Dense	GPU	~4 FPS @ 1080p

Аудит трёх failure-mode

- ☐ Sparsity соответствует фиче: dense для stabilization / interpolation / VSR.
- ☐ Brightness assumption держится: photometric normalization под flicker / low light.
- ☐ Latency бюджет соблюден: 33 мс/кадр на 30 FPS — RAFT не уместится.

Библиотеки и лицензии — быстрая справка

- ☐ Lucas-Kanade / Farnebäck: cv2 (Apache 2.0).
- ☐ DIS: cv2-contrib DISOpticalFlow (Apache 2.0).
- ☐ RAFT: torchvision raft_large (BSD 3-Clause).
- ☐ SEA-RAFT: GitHub (BSD 3-Clause).
- ☐ FlowFormer: GitHub (MIT).