Full Flow: Optical Flow Estimation By Global Optimization over Regular Grids

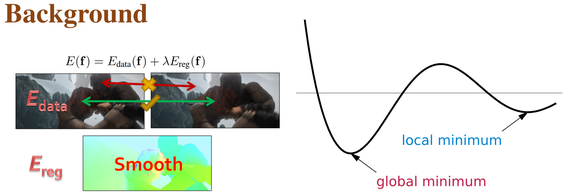
Qifeng Chen Stanford University

Vladlen Koltun Intel Labs

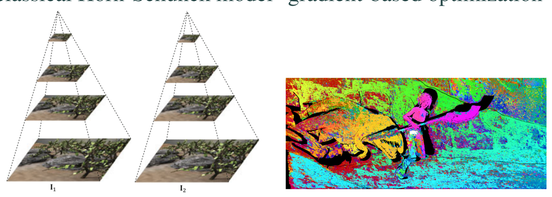
Abstract

We present a global optimization approach to optical flow estimation. The approach optimizes a classical optical flow objective over the full space of mappings between discrete grids. No descriptor matching is used. The highly regular structure of the space of mappings enables optimizations that reduce the computational complexity of the algorithm's inner loop from quadratic to linear and support efficient matching of tens of thousands of nodes to tens of thousands of displacements. We show that one-shot global optimization of a classical Horn-Schunck-type objective over regular grids at a single resolution is sufficient to initialize continuous interpolation and achieve state-of-the-art performance on challenging modern benchmarks.

Background



classical Horn-Schunck model gradient-based optimization



coarse-to-fine optimization

descriptor matching

Model

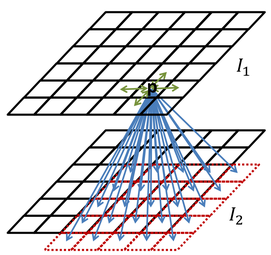


Figure 1: Optical flow over regular grids.

Our objective function is

$$E(\mathbf{f}) = \sum_{p \in I_1} \rho_D(p, f_p, I_1, I_2) + \lambda \sum_{\{p,q\} \in \mathcal{N}} w_{p,q}\, \rho_S(f_p - f_q),$$

where $\mathcal{N} \subset \Omega^2$ is the 4-connected pixel grid and

$$\rho_D(p, f_p, I_1, I_2) = 1 - \max(NCC, 0),$$
$$\rho_S(f) = \min\left(\rho(f^1) + \rho(f^2), \tau\right).$$

f^1, f^2 are the two components of vector f and $\rho(\cdot)$ is a penalty function, such as the L^1 or the Charbonnier penalty.

Optimization

Our objective is a discrete Markov random field with a two-dimensional label space. At first glance, global optimization of this model may appear intractable. The number of nodes and the number of labels are both in the tens of thousands. We show that the full problem is tractable.

Message passing algorithm

To optimize the model, we use TRW-S due to its effectiveness in optimizing models with large label spaces. We can optimize an equivalent MRF objective:

$$\boldsymbol{\theta}(1) = \sum_{p \in \Omega} \theta_p(l_p) + \sum_{\{p,q\} \in \mathcal{N}} \theta_{pq}(l_p, l_q).$$
$$\theta_p(s) = \rho_D(p, s, I_1, I_2),$$
$$\theta_{pq}(s,t) = \rho_S(s - t).$$

The message update rule is

$$m_{p \to q}(t) = \min_s \left(\phi_{pq}(s) + \theta_{pq}(s,t)\right),$$
$$\phi_{pq}(s) = \frac{1}{2}\Big(\theta_p(s) + \sum_r m_{r \to p}(s)\Big) - m_{q \to p}(s).$$

Complexity reduction

A brute-force implementation of a message update requires $O(M^2)$ operations, where M is the number of labels (tens of thousands). We now show that, given ϕ_{pq}, all elements of the message $m_{p \to q}$ can be computed in $O(M)$ operations (ϕ_{pq} can be computed using $O(M)$ operations). Rearranging terms, we obtain

$$m_{p \to q}(t) = \min\Big(\mathcal{D}_{pq}(t), T_{pq}\Big),$$

where

$$\mathcal{D}_{pq}(t) = \min_s \left(\phi_{pq}(s) + \rho(t^1 - s^1) + \rho(t^2 - s^2)\right),$$
$$T_{pq} = \min_s \left(\phi_{pq}(s) + \tau\right).$$

T_{pq} can be computed using $O(M)$ operations given ϕ_{pq}. $\mathcal{D}_{pq}$ can be decomposed into two sets of $O(\sqrt{M})$ 1D min-convolutions:

$$\mathcal{D}_{pq}(t^1, t^2) = \min_{s^2} \mathcal{D}_{pq|s^2}(t^1) + \rho(t^2 - s^2),$$
$$\mathcal{D}_{pq|s^2}(t^1) = \min_{s^1} \phi_{pq}(s^1, s^2) + \rho(t^1 - s^1).$$

Each min-convolution can be evaluated in $O(\sqrt{M})$ operations, for a total complexity of $O(M)$.

Further acceleration

A min-convolution has the following general form:

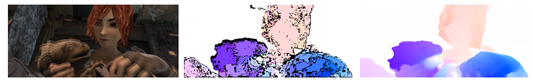
$$h(i) = \min_j g(j) + \rho(i - j).$$

The min-convolution can be computed using $O(n)$ operations, where $n = \sqrt{M}$ However, commonly used algorithms require computing intersections of shifted copies of ρ. We use totally monotone matrix searching [1] to compute the min-convolution without computing intersections.

Implementation

Parallelization. Speedup by a factor of 6.6.

Occlusion handling. We adopt the common tactic of forward-backward consistency checking: compute the forward flow from I_1 to I_2 and the backward flow from I_2 to I_1, and discard inconsistent matches.



(a) Input image (b) Before interpolation (c) After interpolation

Figure 2: Postprocessing.

Postprocessing. After removing inconsistent matches as described in the previous paragraph, we interpolate the results to get subpixel-resolution flow. We use the EpicFlow interpolation scheme [2], which has become a common postprocessing step in recent state-of-the-art pipelines.

Experiments

Our Matlab code is less than 50 lines long, not including the general-purpose solver. The running time is about 30 seconds for 3 iterations of TRW-S. We perform the optimization on $1/3$-resolution images.

Comparison to prior work

In experiments reported in this section, we use the L^1 norm for regularization ($\rho(x) = |x|$) and no truncation ($\tau = \infty$). This decision is motivated by the controlled experiments.

	Final pass			Clean pass		
	all	noc	occ	all	noc	occ
FlowFields	**5.810**	**2.621**	31.799	3.748	**1.056**	25.700
FullFlow	5.895	2.838	**30.793**	**3.601**	1.296	**22.424**
DiscreteFlow	6.077	2.937	31.685	**3.567**	1.108	23.626
EpicFlow	6.285	3.060	32.564	4.115	1.360	26.595
TF+OFM	6.727	3.388	33.929	4.917	1.874	29.735
NNF-Local	7.249	2.973	42.088	5.386	1.397	37.896
PH-Flow	7.423	3.795	36.960	4.388	1.714	26.202
Classic+NL	9.153	4.814	44.509	7.961	3.770	42.079

Table 1: Endpoint errors on the MPI Sintel test set.

MPI Sintel. MPI Sintel is a dataset for large-displacement optical flow. There are two types of sequences in the dataset, clean and final. The clean sequences exhibit a variety of illumination and shading effects including specular reflectance and soft shadows. The final sequences additionally have motion blur, depth of field, and atmospheric effects.

Our approach ranks 2nd on the key EPE-all metric for both final and clean sequences.

KITTI 2015. KITTI Optical Flow 2015 is an optical flow dataset that comprises outdoor images of dynamic scenes captured from a car. Note that SOF was developed concurrently with our work and uses substantially more information at training time.

	all	non-occluded
SOF	**16.81%**	**10.86%**
DiscreteFlow	22.38%	12.18%
FullFlow	24.26%	15.35%
EpicFlow	27.10%	17.61%
DeepFlow	29.18%	19.15%
Horn-Schunck	42.18%	34.13%

Figure 3: Accuracy of different methods on the KITTI 2015 test set.

Controlled experiments

The generality of the presented optimization framework enables a comprehensive evaluation of different data terms and regularization terms.

	Truncated	Non-truncated
NCC+L^1	**2.710**	**2.710**
NCC+Charbonnier	2.883	2.888
NCC+L^2	2.896	2.976
HS+L^1	5.972	6.523
HS+Charbonnier	5.847	6.181
HS+L^2	6.337	6.529

Table 2: Controlled evaluation of the data term, penalty function, and truncation.

The results are shown in Table 2, which provides the average EPE over the images used for the evaluation for each combination of the three factors (data term, penalty function, truncation). The results suggest that the data term is of primary importance: the patch-based truncated NCC term is much more effective than the pixelwise Horn-Schunck data term, irrespective of the regularizer.

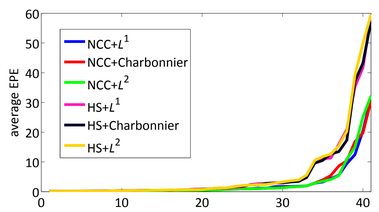


Figure 4: Average endpoint error for each tested image, sorted by magnitude in each condition.

References

[1] Alok Aggarwal, Maria M. Klawe, Shlomo Moran, Peter W. Shor, and Robert E. Wilber. Geometric applications of a matrix-searching algorithm. *Algorithmica*, 2, 1987.

[2] Jérome Revaud, Philippe Weinzaepfel, Zaïd Harchaoui, and Cordelia Schmid. EpicFlow: Edge-preserving interpolation of correspondences for optical flow. In *CVPR*, 2015.