CATEGORY: COMPUTER VISION - CV10

POSTER P4254 | CONTACT NAME Byungmoon Kim: bmkim@adobe.com

GPU TECHNOLOGY CONFERENCE

A Comparison of TV-L1 Optical Flow Solvers on GPU

Linchao Bao[1,2], Hailin Jin[1], Byungmoon Kim[1], Qingxiong Yang[2]

[1] Adobe Systems Inc.

[2] City University of Hong Kong

1. Abstract

Optical flow has always been an important topic in computer vision and graphics, as it serves as basic building block for many applications. Optical flow algorithms are typically known to be very slow. In this study, we implemented three TV-L1 optical flow solvers on GPU using CUDA, and compared their convergence and performance. Our study provides a guideline for choosing right algorithms in practice.

2. TV-L1 Optical Flow

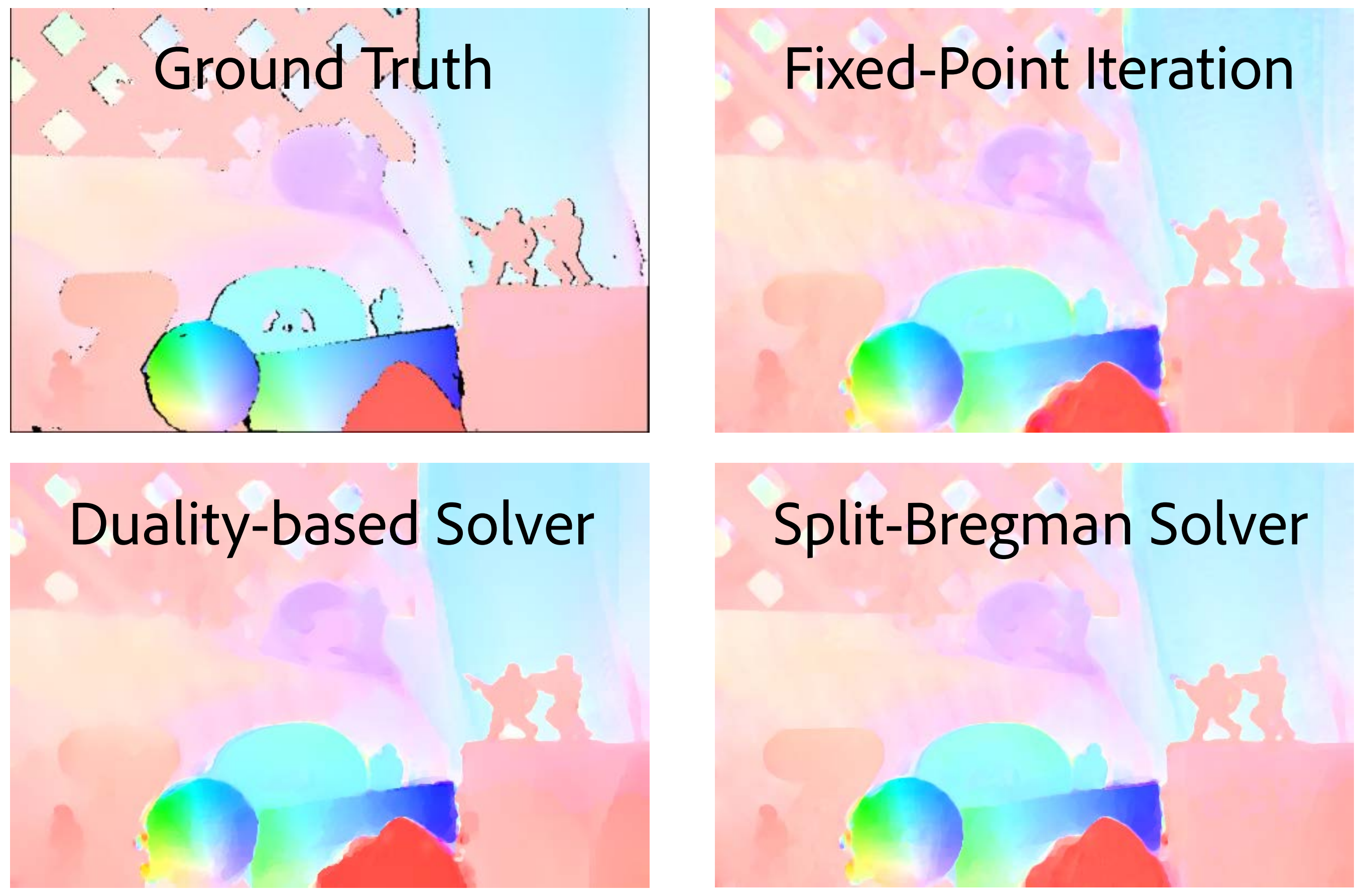
The TV-L1 optical flow is to minimize the following functional (with *isotropic* total variation)

$$E[u(x,y),v(x,y)] = \iint |I_x u + I_y v + I_t| + \lambda\left(\sqrt{u_x^2+u_y^2}+\sqrt{v_x^2+v_y^2}\right) dxdy$$

where *u* and *v* are unknown displacements along *x* and *y* direction (both are a function of *x* and *y*), respectively. The subscripted notation stands for partial derivatives.

3. Classic Coarse-To-Fine Framework

Since the derivation of the optical flow formulation is based on the assumption of small motions (the *linearization*), the classic way to solve it is to employ a coarse-to-fine framework based on warping. And within each warping, the core solver is to solve the incremental flow. In this study, we implemented and compared 3 different solvers on GPU.



4. TV-L1 Solver 1 – Fixed-Point Iteration (FP)

The idea of the fixed-point iteration solver [1] is to substitute nonlinear terms with *u* and *v* values from the previous iteration. Then the problem becomes a quadratic minimization problem (i.e., a linear system), which we solve using Jacobi iterations.

5. TV-L1 Solver 2 – Duality-based Solver (DL)

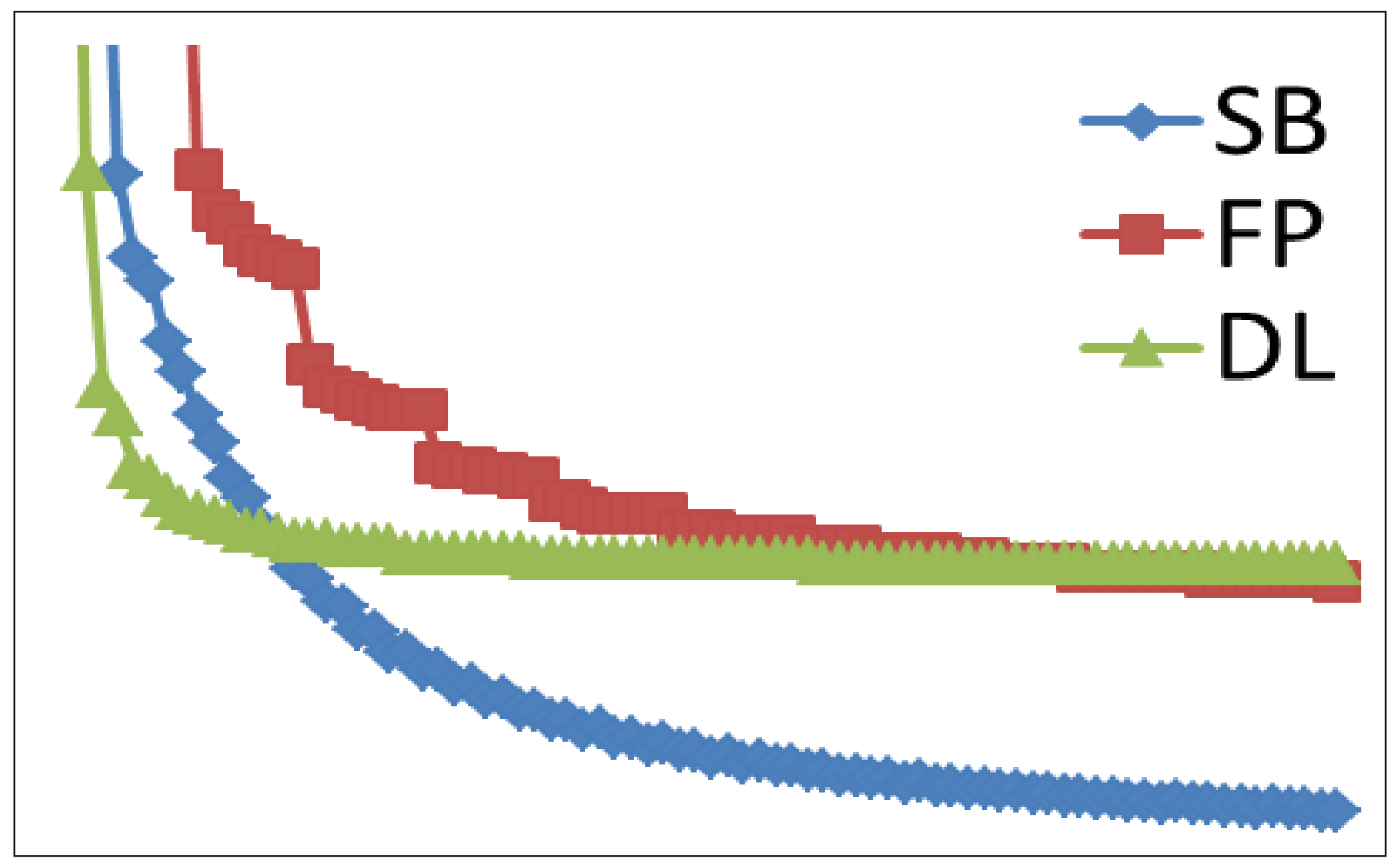
The idea of the duality-based solver [2] is to decouple the data term and smoothness term using auxiliary variables. Then one of the sub-problem becomes a ROF model and can be solved using Chambolle's algorithm [3]. The other sub-problem is a per-pixel minimization, which can be solved by thresholding.

6. TV-L1 Solver 3 – Split-Bregman Solver(SB)

The idea of the split-Bregman solver [4] is to convert L1 terms into quadratic terms using auxiliary variables. Then it becomes solving a quadratic minimization problem and updating the auxiliary variables. Note that it has a nice mathematical theory behind it [5]. We also use Jacobi iteration to solve the quadratic minimization.

7. Solver Comparison

We compared the three solvers (within one warping iteration) with the same input, as well as their own best parameter settings. The cost plot for many real scenes is usually like this



8. Solver Performance on GPU

Most components of the algorithms are good fit to GPU. We use texture memory to perform neighboring propagation in Jacobi iterations and use shared memory for Gaussian filtering and median filtering. For a typical 640 x 480 image, if we run the three solvers to a similar accuracy, the performance is as follows

Processor	SB (sec)	FP (sec)	DL (sec)
CPU (i7-3.8G)	40	16 (SOR)	9
GTX 780 Ti	0.31	0.54 (Jacobi)	0.14
GPU Speed-up	**130x**	**30x**	**60x**

9. Deep Matching: Handling Large Displacement

We also investigated a very recent state-of-the-art LDOF algorithm named DeepFlow [6]. The idea is to incorporate a new correspondence matching algorithm deep-matching. The implementation required critical sections (realized by atomic operations) to find minimum from responses computed from different threads and blocks .

- CPU 2 sec
- GTX 780 Ti 0.04 sec (Speed-up = **50x**)

10. Conclusion

Our study shows that TV-L1 optical flow solvers can be largely speed-up on modern GPU. For high quality flow result, SB solver is the best choice, while for high performance computation, DL solver is a better choice. The new deep-matching algorithm can be employed to handle large-displacement motions in a very efficient way.

Reference

[1] Brox et al, "High Accuracy Optical Flow…", ECCV'04
[2] Wedel et al, "An Improved Algorithm for TV-L1…", DAGM'09
[3] Chambolle, "An Algorithm for Total Variation…", JMIV'04
[4] Ayvaci et al, "Sparse Occlusion Detection for Optical…", IJCV'11
[5] Goldstein and Osher, "The Split-Bregman Method…", SIAM IS'09
[6] Weinzaepfel et al, "DeepFlow: Large displacement…", ICCV'13