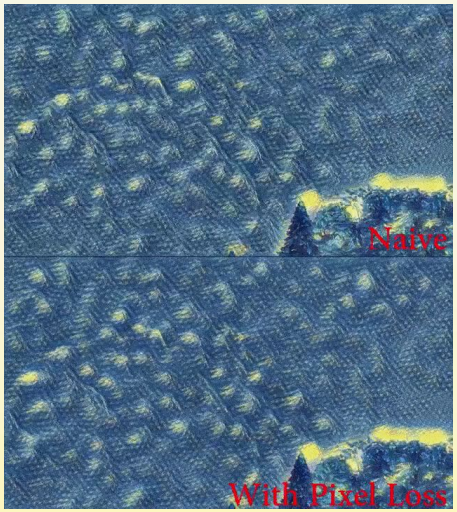
Parallelize 2D Optical Flow Estimation Algorithm on Video

Team: Shiyu Huang, Hongxiang Qiu, Zeyu Zhao, Zongren Zou

About Optical Flow

Source: https://people.csail.mit.edu/celiu/OpticalFlow/

- Optical flow is important in motion analysis
  - Object tracking and activity recognition
  - Motion based segmentation
  - Video processing
    - Fake slow motion video
    - Stabilize synthesized video
    - Video compression



Model and Data

- The state-of-the-art implementation is *DeepFlow: Large displacement optical flow with deep matching (ICCV 2013)*
  - Code made available by authors.
  - Contains two parts
    - Deep Match (match pixels in two images)
      - Has GPU implementation
    - Deep Flow
      - Single-thread CPU based.
      - A pair of 480p images: 5 seconds

- We will focus on the application for video processing. Test data can be any video online.

Where to Parallelize

For example, for slow motion application:

A sequence of images.

Images could be processed independently

Match objects/pixels in two images.

Heavy computation, already has GPU implementation

Compute the optical flow.

Use the flow data to generate slow motion frames.

Heavy computation, could apply parallelism

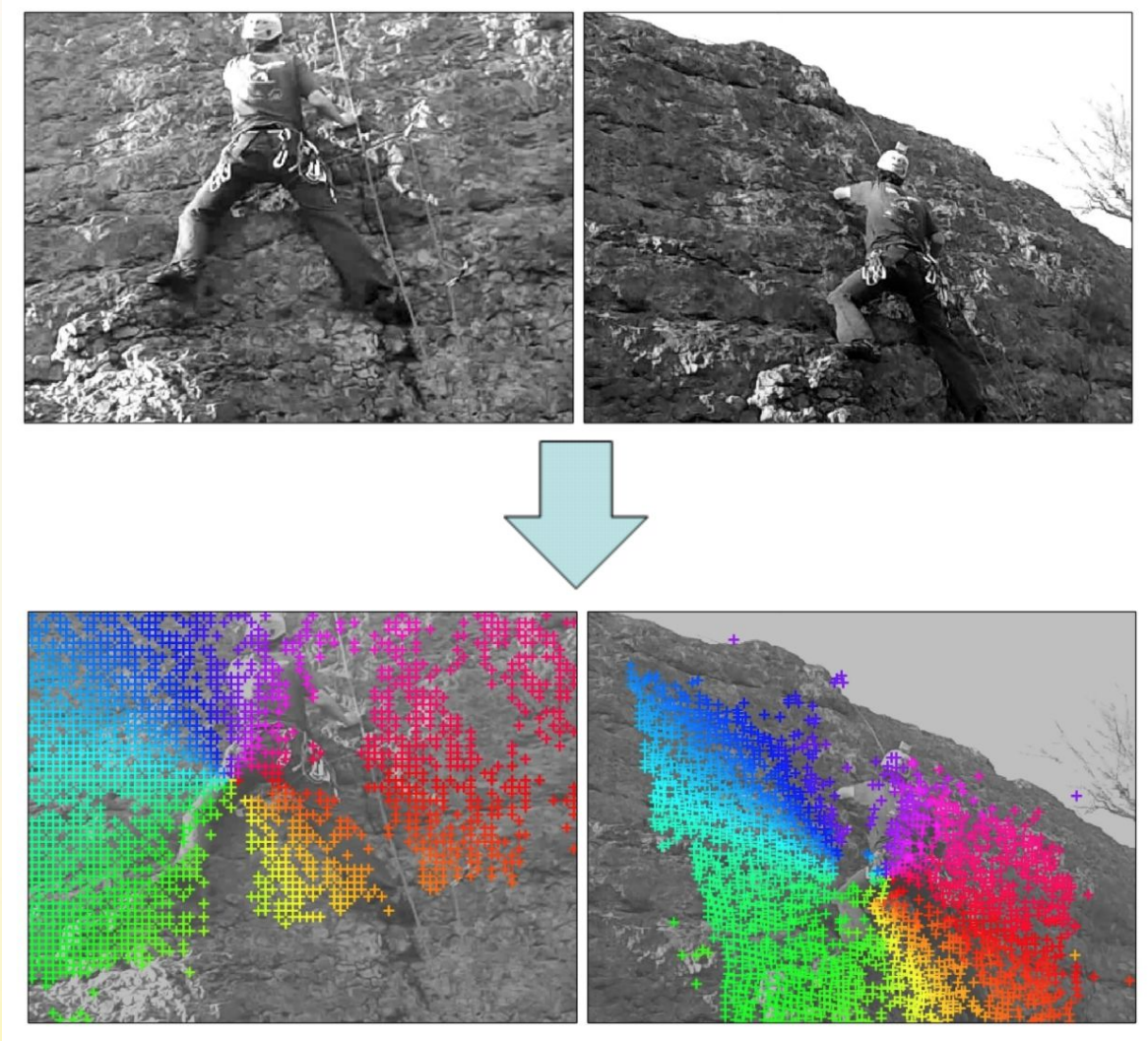
Combine slow motion frames as a video.

- To processing images in parallel, we can do MapReduce
- To accelerate the computation of deep flow and output generation, we can do either OpenMP or OpenACC/CUDA

Appendix

Algorithm

Deep Match:



Algorithm

Deep Flow: