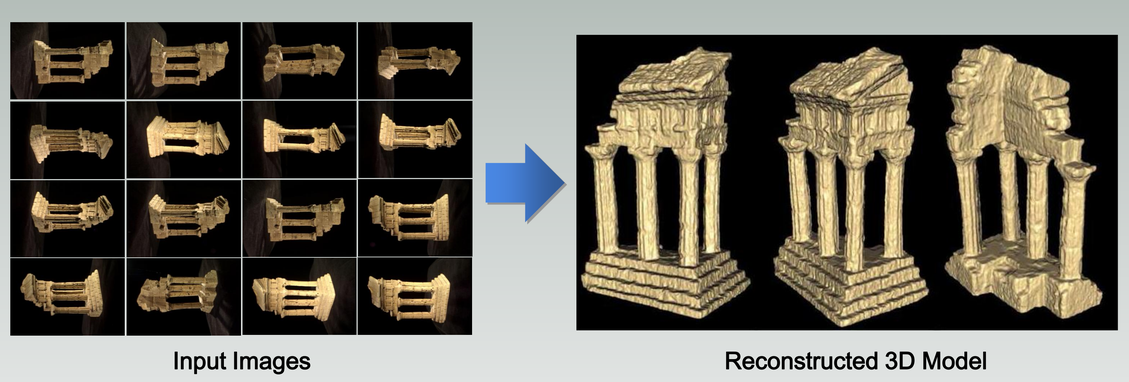
GPU-Friendly Multi-View Stereo Reconstruction Using Surfel Representation and Graph Cuts

Haesol Park, Ju Yong Chang, Man Hee Lee, In Kyu Park, Kyoung Mu Lee, Sang Uk Lee

Seoul National University, Inha University

Multi-View Stereo Reconstruction

- Reconstruct 3D shape of object from multiple calibrated images
- A conventional but still active problem in computer vision and graphics



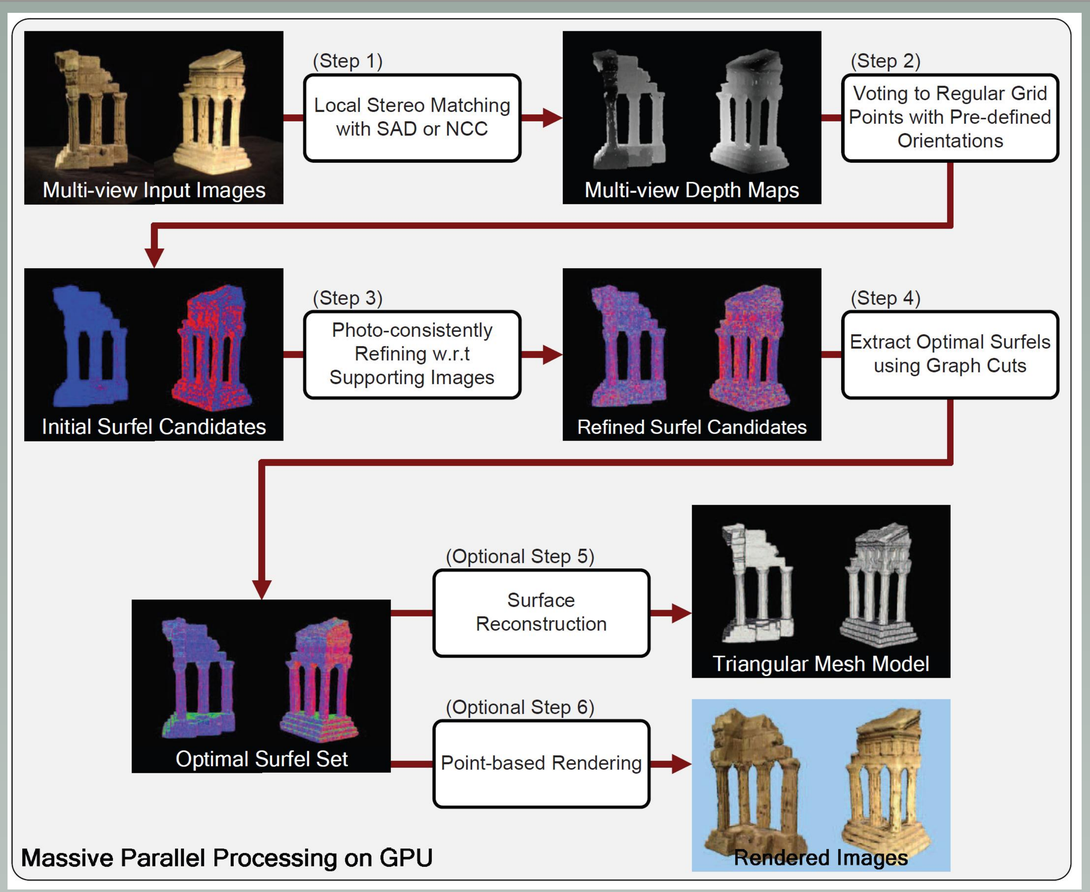
Input Images

Reconstructed 3D Model

Main Contribution

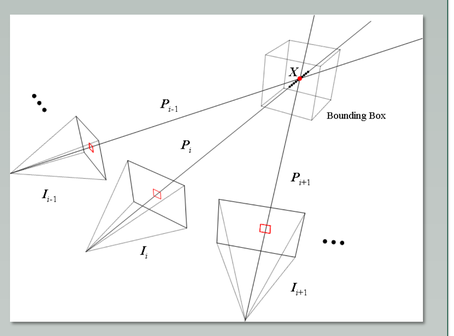
- We effectively combine a surfel -based surface representation with the global optimization framework using graph cuts, allowing the proposed method to produce the optimal surfels in the principled way
- By using photo-consistently refined surfels rather than regularly shaped voxels for the graph cuts optimization, surface details are more accurately reconstructed
- Algorithms are GPU-friendly and implemented ENTIRELY ON GPU using NVIDIA CUDA technology.
- The runtime is reduced to less than 1% compared to the CPU runtime

Proposed Method



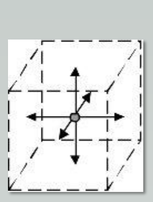
Step1 : Multi-view stereo matching

- Local window matching for photo-consistency test with NCC metric
- Each GPU thread computes locally optimal depth for each pixel in reference image



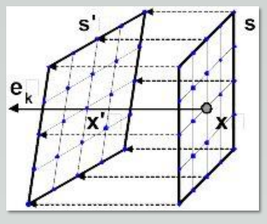
Step2 : Initialization of surfel candidates

- Surfel = a 3D patch with position and orientation
- Each pixel votes for a corresponding voxel in 3D regular grid
- For each voted voxel, generate surfels with 6 standard directions considering the visibility from supporting views
- Each GPU thread performs the voting for a single voxel



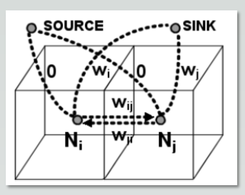
Step 3 : Refinement of surfel orientation and position

- Determine optimal position and orientation for each surfel by evaluating candidates' NCC scores in massive parallel way on GPU
- Each GPU thread optimizes a single surfel



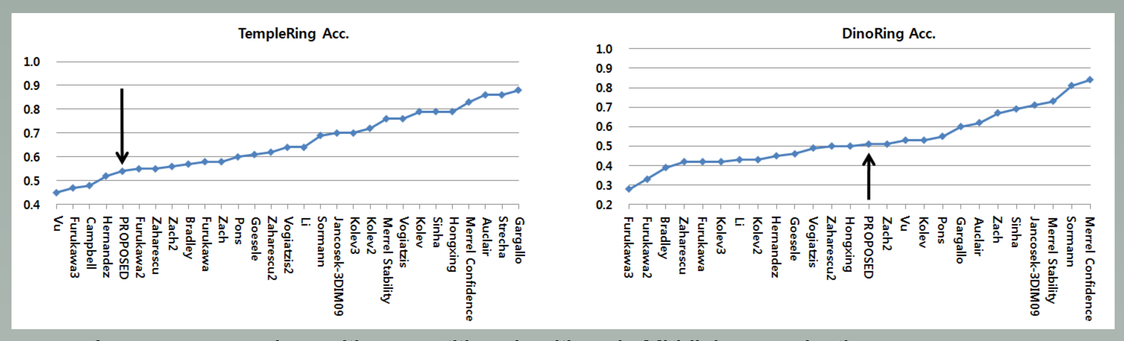
Step 4 : Extraction of optimal surfels using graph cuts

- 6–neighborhood graph
- NCC score of surfels form edge weights
- Perform graph cuts using push-relabel algorithm
- Each GPU thread is per a single node, which is essentially a cell in the 3D regular grid
- GPU graph cuts are implemented using CUDA Cuts but improved

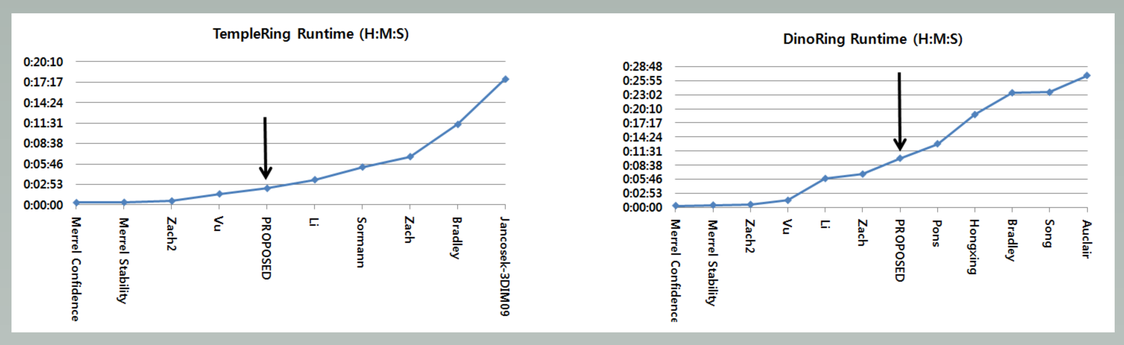


Experiments & Results

- Comparative evaluation for *TempleRing* (47 images) and *DinoRing* (48 images) dataset from Middlebury website



Accuracy comparison with competitive algorithms in Middlebury evaluation



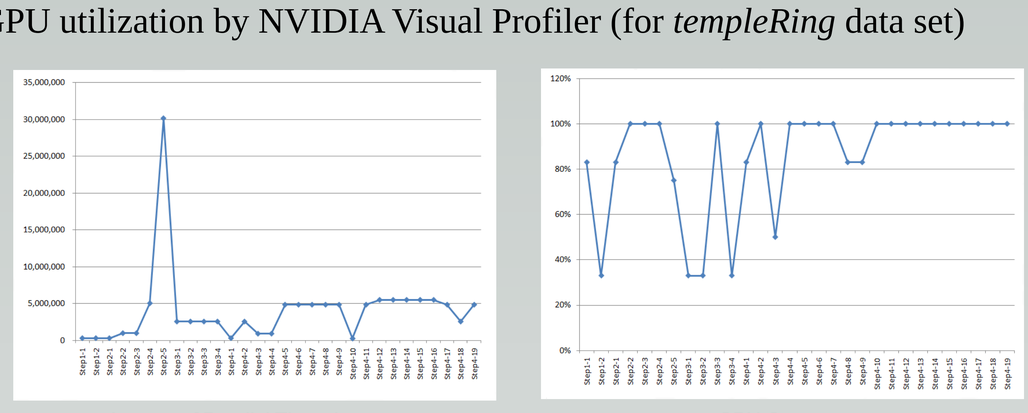
Runtime comparison with competitive algorithms in Middlebury evaluation

- Runtime comparison for *templeRing* data set

	Stereo Matching	Initialization	Refinement	Extraction	Total
CPU	3467	1189	8911	1995	15562
GPU	31	7	58	44	140
Ratio	111.84	169.86	153.64	45.34	111.16

CPU : Intel Core i7 920 2.67GHz, 4GB RAM, GPU : NVIDIA Tesla C2050 3GB RAM (in seconds)

- GPU utilization by NVIDIA Visual Profiler (for *templeRing* data set)



Amount of Parallel Threads

GPU Occupancy

- 3D reconstruction results on various data set