Fast and efficient dense variational stereo on GPU

J. Mairal - R. Keriven
A. Chariot
CERTIS, ENPC
Marne-La-Vallée, France

Problem Statement: Achieve fast variational stereo reconstruction.

Context:
- Variational stereo algorithms are usually slow but provide smooth and accurate results.
- Graphics cards can be considered nowadays as very efficient parallel SIMD machines.

Achievements:
- Parallelization and discretization of an algorithm on a GPU.
- Speedup of 10-15 times between a recent GPU and a CPU (3 GHz).
- Extension to three cameras.

Future work:
- Work with more than one GPU.
- Achieve fast multi-view reconstruction on GPU.
- Taking discontinuities into account.

GPU Discretization

A simple two-cameras model which runs entirely on the GPU.

- The images are back-projected onto the surface $\mathcal{S}$, which is composed of triangles $\mathcal{T}$.
- $\rho_T$ denotes the normalized cross-correlation function on the triangle $\mathcal{T}$.
- $\tau(s)$ denotes a regularization function which is adapted for a GPU implementation.

Energy minimization by means of a descent gradient.

- Several levels of detail to prevent from converging toward a local minima.
- For each vertex $M$ of the mesh, $\frac{\partial E(S)}{\partial d_M} = \sum_{T \in \mathcal{T}} \frac{\partial E_T}{\partial d_M}$. is computed, where $V(M)$ is the set of the 6 triangles to which $M$ belongs.

Direct regularization by smoothing $d_M$.

- For each vertex $M$, where $M'$ is its neighbors. The obtained results here were better than the use of mean curvature motion.

An extension to three cameras

An algorithm almost as fast as the two-cameras version, which still runs on the GPU.

- One camera is denoted as the « left » or « center » camera.
- We consider the « left » and the « right » camera.
- The triangles are classified in a first step in three categories, using the Stencil Buffer of the graphics cards:
  - Occluded triangles.
  - Triangles associated with the « left » camera: $S_L$.
  - Triangles associated with the « right » camera: $S_R$.

- For each « non-occluded » triangle, a simple normal test to select the best camera.
- Then, the two-camera algorithm is run on $S_L$, then on $S_R$.
- The complexities of the two-camera and the three-cameras version are the same.
- The overhead between them is about 30% on the GPU.

Results on data sets with occluded areas are better.

Results:

- These examples were obtained with two input images.
- The two first datasets are courtesy of Kyros Kutulakos (University of Toronto).

Fast results:

- These examples were computed in less than 250ms with a NVIDIA Geforce 7800 GTX.
- Not as fast as plane-sweep based methods, but more accurate !

GPU/CPU speedup : 10 to 15
- Geforce 7800 vs 3GHz CPU, Iterations per second for one particular level of detail.

References:


