

Spacetime Feature for Matching On-Vehicle Video Sequences

Ryota Matsuhisa, Shintaro Ono, Hiroshi Kawasaki*, and Katsushi Ikeuchi

In recent years, on-vehicle cameras are widely spreading and videos in various areas are daily recorded by general vehicles. If it is able to find the same area from them, it will become able to exploit plenty of images effectively for integration, 3D reconstruction, etc. We propose the spacetime feature for continuous DP matching, composed of EPI (Epipolar Plane Image) and THI (Timeline Height Image). The most interesting character of this process is that all static scene properties in urban scenes, except photometry, are considered: Color and depth in EPI, height in THI, and width in DP matching process. In the experiment in Shinjuku area, the answer for a query video segment gave as few as 5 frame error under similar lighting condition. Also it was proved that the feature tends to be robust under low resolution and low framerate.

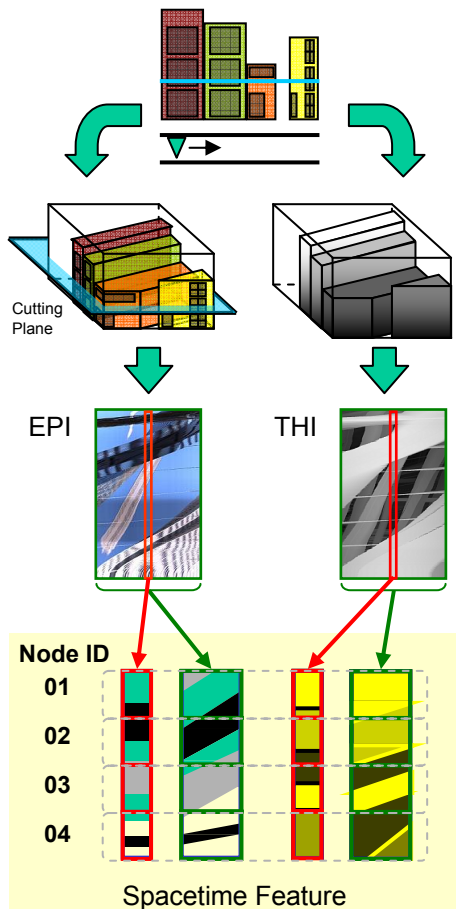
Publications

- [1] R. Matsuhisa, S. Ono, H. Kawasaki, K. Ikeuchi, "3D City Modeling System using Spatiotemporal Matching of Multiple Vehicle Videos" (in Japanese), *16th Symposium on Sensing via Image Information (SSII)*, Jun. 2010.
- [2] R. Matsuhisa, H. Kawasaki, S. Ono, K. Ikeuchi, "Vehicle camera image sequences matching with spatio temporal features" (in Japanese), *Meeting on Image Recognition and Understanding (MIRU)*, Jul. 2010.

Spacetime Feature

A spacetime volume (from on-vehicle video) provides:

- EPI = Epipolar Plane Image
- THI = Timeline Height Image



Using Spacetime Feature for DP

All static scene properties (except photometry) are included.

- Color: in EPI
- Structure (height): THI
- Structure (depth): EPI
- Structure (width): DP matching process

Experiment in Shinjuku Area



Searching query (A)

Omnidirectional videos A, B are captured with different running speed, from different lane and direction.

B_O : Using original video B

B_L : Low-resolution video

B_F : Frame-skipped video

B_{LF} : Frame skip & Low res

