Real-time Virtual Object Insertion

Paolo Favaro
Washington University
Electrical Engineering
Campus Box 1127
St. Louis, MO 63130
fava@ee.wustl.edu

Hailin Jin
Washington University
Electrical Engineering
Campus Box 1127
St. Louis, MO 63130
hljin@ee.wustl.edu

Stefano Soatto
University of California, Los Angeles
Computer Science
Los Angeles, CA 90095 and
Washington University, St. Louis
soatto@cs.ucla.edu, soatto@ee.wustl.edu[*]

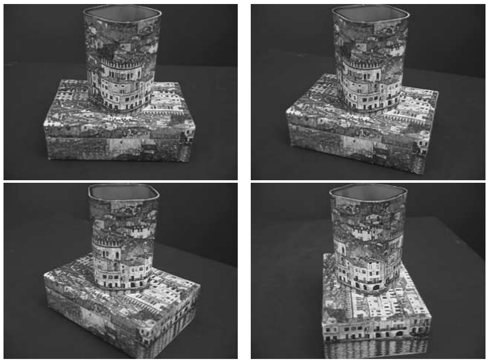

Figure 1. An object is placed on a turntable, which is rotated in front of the camera. Four snapshots are shown.

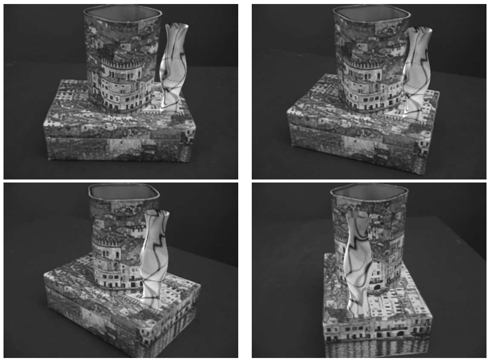

Figure 2. Once the camera motion and the scene structure are estimated, we insert a "virtual" vase into the scene. As it can be observed, its relative position is fixed within the scene. Four rendered views corresponding to the snapshots in Figure 1 are shown.

We present a system to insert virtual objects into real image sequences in real time. The system consists of off-the-shelf hardware (a camera connected to a Pentium PC) and software to (a) automatically select and track region features despite changes in illumination, (b) estimate three-dimensional position and orientation of surface patches relative to an inertial reference frame despite individual point-features appearing and disappearing, (c) insert a texture-mapped virtual object into the scene so as to make it appear to be part of the scene and moving with it (see Figures 1 and 2). This is all done in real time. Additional graphic features, such as cast shadows, inter-reflections, occlusions, can be handled off-line. While virtual object insertion has been demonstrated before, and is even part of commercial video editing products nowadays [1], current systems require several batch steps, including feature tracking, outlier rejection, epipolar bundle adjustment, some of which involve human intervention. What our demo shows is that results of only slightly inferior quality can be obtained in a completely automatic fashion (no human intervention whatsoever), while processing the sequence causally (the pose of the virtual object at time t is decided based on observations of the sequence up to time t). This enables a whole new range of applications for real-time interaction with mixed reality (real as well as virtual objects), all on commercial off-the-shelf hardware while the camera is undergoing arbitrary motion, including hand-held. Our system uses a region tracking algorithm [3] based on a normalized affine warp that accounts for the motion of smooth planar patches and compensates for changes in illumination. Outliers, such as T-junctions, are automatically detected as violations of the region deformation model and rejected. An extended Kalman filter (analogous to [2]) interfaces with the low-level tracker in order to estimate aggregate rigid motion, while allowing individual point-features to appear and disappear as the low-level tracking model is violated. It runs in real time for 40 to 50 patches on a dual-processor 1GHz Pentium III computer. The multi-thread C++ code, which is readily interfaced with a frame grabber as well as Matlab for development, will be made available to the public at the demonstration.

References

[1] `http://www.2d3.com`

[2] A. Chiuso, P. Favaro, H. Jin and S. Soatto. MFm: 3-D Motion From 2-D Motion Causally Integrated Over Time. In *ECCV*, June 2000.

[3] H. Jin, P. Favaro and S. Soatto. Real-time Feature Tracking and Outlier Rejection with Changes in Illumination. In *ICCV*, July 2001.

[*] This research is supported in part by Intel grant 8029.