Supporting material: Plane labeling trinocular stereo matching with baseline recovery

Luis Horna
The University of Edinburgh
s1145121@sms.ed.ac.uk

Robert B. Fisher
The University of Edinburgh
rbf@inf.ed.ac.uk

February 21, 2017

Abstract

In this document we present additional material for the paper submission "Plane labeling trinocular stereo matching with baseline recovery" which recovers the transformation that describes the displacement of a binocular stereo rig in a scene, and uses this to include a third image to reduce some of the ambiguities inherent to binocular stereo. The core idea of the proposed algorithm is the assumption that the binocular baseline is projected to the third view, and thus can be used to constrain the transformation estimation of the stereo rig.

1 Baseline recovery considerations

In section 3.1 (of submitted paper) $T_{lu} = \beta K^{-1} e'_{lu}$, and $T_{ru} = \beta K^{-1} e'_{ru}$ are used as initial estimates. Notice that the line connecting e'_{lu} and e'_{ru} should result in a parallel line to the x axis when T_{lu} and T_{ru} are exact, but under realistic conditions and noise T'_r is an initial estimate of T_r. Because of this situation our algorithm uses Levenberg-Marquardt to find updated versions $\hat{T}_{lu}$ and $\hat{T}_{ru}$. It's important to note that a rotation/noise in the initial camera pose may cause swapped positions of T_{lu} and T_{ru} due to epipoles e'_{lu} and e'_{ru} locations, resulting in an inverted T'_r estimate. This may be consistent with the three views but it is incorrect. To handle this situation a second optimization is done using the previously computed $R_{\Delta\theta_x \Delta\theta_y \Delta\theta_z}$ (see section 3.1) as initialization to estimate updated $(\hat{T}'_{lu}, \hat{T}'_{ru})$. The new solution is compared to first estimate and the best solution kept. We found this approach to be able to handle the inverted T'_r problem.

2 Baseline recovery diagram

To further expand the explanation of section 3.1, fig.1 shows how matching keypoints $(x_{li}, x_{ri}, x_{li}^u)$ are used to compute initial fundamental matrices (F_{lu}, F_{ru}) later used to extract (R, T_{lu}, T_{ru}), and finally compute $(\hat{R}, \hat{T}_{lu}, \hat{T}_{ru})$.

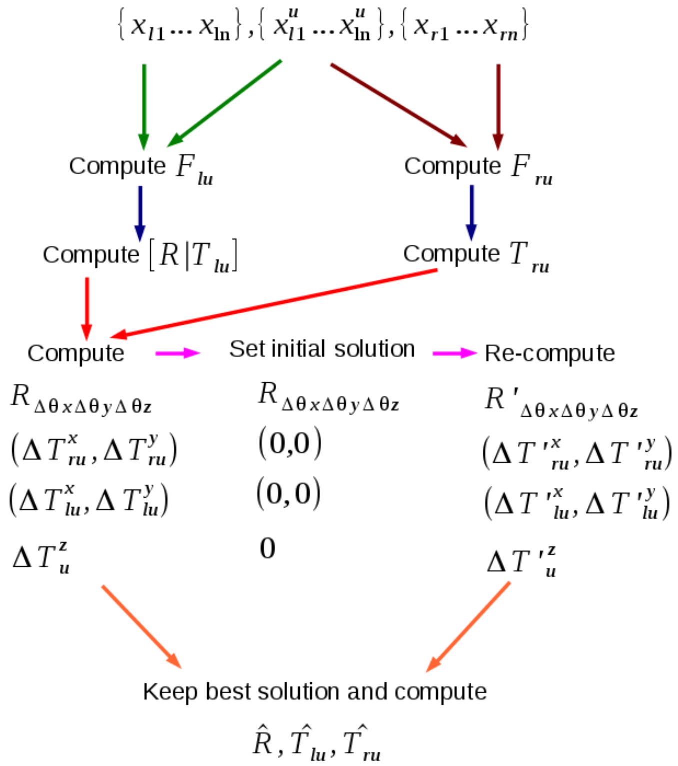


Figure 1: baseline recovery algorithm

3 Parameter settings and training considerations

Our algorithm only computes left and right disparity maps. The third image is only used to improve the pixel similarity cost. To estimate the disparity plane assignment we use the multi-scale DPI algorithm from [1] and replace their raw pixel cost with eq.9 from the submitted paper with $\alpha_t = 0.8, \tau_{grad}^b = 3/255, \tau_{grad}^t = 1.33\tau_{grad}^b, \tau_{cen}^b = 9/25, \tau_{cen}^t = 1.2\tau_{cen}^b$. The DPI algorithm uses following parameters: aggregation window sizes of $\omega_1 = 41 \times 41$, $\omega_2 = 25 \times 25$, $\sigma_r = 10/255$, $\sigma_d = 0.5$, $\tau_w = 2.5$, $\tau_{diff} = 0.07$, $\tau_{unique} = 0.01$, $\alpha = 30$, $K_1 = 1$, $K_2 = 6$, $r = 2$, $P = 6$, $\tau_h = 2$, $\tau'_w = 0.5$, $K_w = 8$. For quarter and half size $\lambda = 0.09$, and $\lambda = 0.18$ for full size. The disparity estimation is iterated 4 times at quarter size, 1 time at half-size and 4 times at full size. The refinement iterations are set to 5 times each scale. These parameters were obtained by using the Middlebury training data, and every fifth image from KITTI 2015 and 2012 training data.

4 Converting a disparity map to optical flow

In section 5 we convert our resulting disparity maps to optical flow. The conversion process is done using the following steps:

(1) Use the baseline recovery algorithm to compute $[\hat{R}|\hat{T}_{lu}]$.

(2) Use the estimated disparity at point x_{li} to obtain 3D point X_i using existing calibration.

(3) Project 3D point X_i to x^u_{li}.

(4) Compute $\vec{f_{lu}} = x^u_{li} - x_{li}$. The vector $\vec{f_{lu}}$ is the optical flow.

This conversion only works for static scenes such as those in KITTI 2012. Notice that no actual matching (x^l_{ui}, x_{li}) is computed. If a dynamic scene is presented this optical flow may not be correct.

5 KITTI 2012 results

The results presented in the paper can be found online at the KITTI 2012 web site for stereo and optical flow under the name *TBR*. Use the following links to find our result:

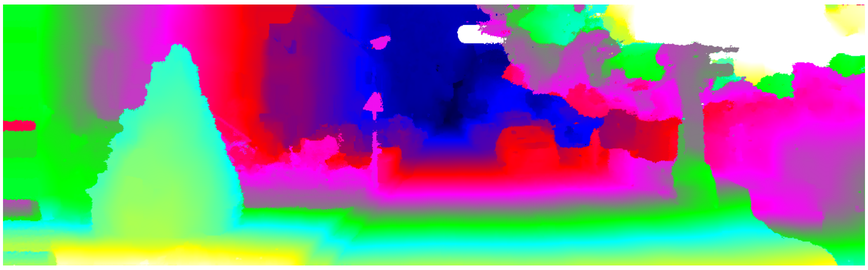
- Stereo results `http://www.cvlibs.net/datasets/kitti/eval_stereo_flow.php?benchmark=stereo`.
- Optical flow results `http://www.cvlibs.net/datasets/kitti/eval_stereo_flow.php?benchmark=flow`.

Fig.2 shows an example of our resulting disparity map and its mapping to optical flow using the recovered motion. The color map used for display is the same as in the KITTI 2012 benchmark, more detailed information can be found in the KITTI 2012 benchmark websites under the table entry *TBR*.

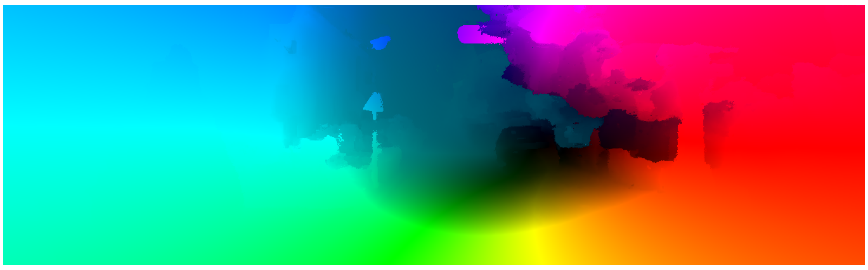
References

[1] L. Horna and R.B. Fisher. 3d plane labeling stereo matching with content aware adaptive windows. *VISAPP*, 2017.

(a) KITTI image 1



(b) Disparity map



(c) Optical flow

Figure 2: Result for KITTI 2012 test image 1.