Algorithm Overview

Extract Feature Points

Using Harris Cornerness Measure.
Subpixel Precision – uses biquadric polynomial fitting

Window size = 6 (gaussian filter window size)
Gaussian std = 1.5
k=0.04, (harris parameter.)
cornerness th = 1e15 (calculated cornerness values below this value are set to zero.)

what more can be done:
preset number of feature extraction i.e, “extract 100 features always” (a variant is implemented)

Match Features

Extracted features are matched using neighbourhood and Normalized cross corelation measures [**zhang deriche et al, a robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry**]
NCC th = 0.7 (cor_th in the code)
Search radius = 100 (KA in the code)
windowSize = 3 (window for correlation calculation)

RANSAC

Uses “Normalized 8-Point” Algorithm.
Bucketing is also implemented (Bucketing: image is divided into 64 squares and random correspondences are selected so that they are not very close to each other –to avoid localization)
Adaptive Iteration Number Calculation Implemented.
Outliers in the correspondence set are reduced.

Max Iteration # = 5000
Error Threshold = 4 (errors lower than this is counted as inlier)

What more can be done ?
- Randomized Ransac: Do not test the model on all data set but on a randomly selected trial set.
- Some winners take all: Do not count all the correspondences giving low errors but a percent of the best matches.

Levenberg Marquardt on F

Nonlinear Minimization of the Sampson Error Cost [Multiple View Geometry] function on the parameters of F matrix is implemented.

Parameters:
initial F : output of the Normalized 8 point algorithm on all of the correspondences after RANSAC.
Max # iteration : 100
Initial mu multiplier : 1
UpdateNormThreshold = 1e-16
ParamNormThreshold = 1e-16

What more can be done ?
- Rank 2 constraint on F can be imposed. It will slow the convergence though. The performance increase may not be considerable. Needs experimentation.
- Guided matching can be included to the system

Depth Estimation

1. Essential Matrix Decomposition (The method described in 3DTV Survey report as “Robust Decomposition”)
2. Fundamental Matrix Computation from the calculated P matrices [Multiple View Geometry]
3. Polynomial Triangulation [Hartley and Sturm “Triangulation”]
    a. First Correspondences are corrected assuming perfect P matrices (noise is assume on only correspondences, not on P matrices)
    b. Since the correspondences are corrected, the back-projected rays now meet in space. So any triangulation method can be used to compute 3D points. We have used “Eigen Triangulation”

4. Denoising of 3D points: Some 3D points are very badly located due to outliers or the noise present on them. So they appear very far away from most of the 3D points (“cloud”) or behind the camera. They are cleaned applying:
    a. mean and std of the distance of the points to the camera center is calculated,
    b. mean – 1.5 x std < points < mean + 1.5 x std are preserved, others discarded.

Visualization

VRML file generator is called. Camera centers and 3d point cloud is marked and viewpoints at the camera centers are generated.

What more can be done ?
Mesh representation of the 3D grid

Reconstruction from Multiple Views

[Metric 3D reconstruction from multiple views, Marc Pollefeys]

- Initial Frame Reconstruction :

Frames 0 and 1 are matched.

Reconstruct a 3D coordinate frame from two views using the above procedure. This step produces an initial framework where all other reconstructions will be build upon. If the initail reconstruction is not general enough (dominant planes or rotation only configurations) multiple reconstruction may fail.

Steps of the initial frame reconstruction:

1. Feature Extraction
2. Feature Matching
3. Robust F estimation using Normalized-8 point algorithm (with RANSAC)
4. Nonlinear minimization of sampson error on F (LM)
5. F decomposition into R and T (K is known) and calculation of P matrices
    a. Some people minimized reprojection error over the parameters of the calculated P matrices. This could be included into our system too. Tests must be performed in order to check if the extra computation is necessary ?
6. Estimation of 3D points using Polynomial Triangulation

- Inserting a new frame to the initial reconstruction

1. Matching between frames 1 and 2 is performed.
2. from these matches, the matches whose 3D points are known are selected. (these points are known due to the initial reconstruction. The points selected are in fact the points which are seen in all of the 3 views: 0, 1 and 2)
3. from the 3D – 2D correspondences, estimate the P matrix for the 2nd frame *robustly*. RANSAC is used for robustness and reprojection error is taken for inlier detection. P matrix is found by using Direct linear transformation (projection equations are stacked and equation matrix is solved by SVD)
4. Nonlinear minimization of the Reprojection error on P matrix parameters (LM)
5. From the remaining 2D – 2D correspondences between frames 1 and 2 (which do not have a 3D estimate), new 3D points are initialized using the found P matrix of the 2nd frame and the previously found P matrix of the 1st frame using Polynomial triangulation.
6. The locations of the 3D points are refined by minimizing reprojection error of all frames for that point.(Not implemented. Might be beneficial)

- Noise cancelation

Points which are not visible for at least a number of frames are extracted from the reconstruction (we used 3)

- Additional frames are inserted to the reconstruction as they come using the formed 3D point cloud as explained before.

What more can be done ?

- For the initialization step, a distance calculation between the frames can be included. This way system will initialize the most general framework.
- Multiple retro frame detection. 3D – 2D matchings will be done using all the previious frames that are close to the current view. This will decrease the error accumulation when a camera goes back and forth in a scene.
- Guided Matching can be included in all relevant steps (after 3D-2D P estimation more matches can be found)

Future Work :

- Bundle Adjustment.
- Self Calibration (? maybe)
- Rectification
- Dense Disparity Estimation
- Dense Depth Reconstruction