benchmark testing plan

s182354 Xiaohan Lyu

April 2019

Sec.1 Detector and Descriptor

1.1 Feature detection

1-a Harris and Shi-Tomasi.

The main idea to extract corners from an image is to use moving windows to compute the vary value of gray scale. If the value has a strong variety, there has corner in this window.
Harris has such feature that rotation invariance but it doesn't have scale invariance.
Shi-Tomasi Detector can be seen as an improvement of Harris detector. Instead of compute the difference between M matrix's determinant and M matrix's trace in the Harris.
Shi-Tomasi proposed that if the smaller of the two eigenvalues is greater than the minimum threshold, a strong corner point is obtained.

1-b Bolb Detector

Instead of trying to detect corners, one may use local extreme of the responses of certain filters as interest points.

1-c FAST and ORB

FAST corner detector uses a circle of 16 pixels (a Bresenham circle of radius 3) to classify whether a candidate point p is actually a corner.
FAST offers higher performance than other feature detectors, but it is not very robust in the presence of noise. Also, FAST is just a key-points detection algorithm and it does not give feature description of key-points.
ORB solved some disadvantage of FAST. ORB is a feature detector and descriptor based on BRIEF and FAST. An improvement of ORB is that it can resist the noise. ORB offers performance much better both than SIFT and SURF.

1-d SIFT and SURF

SIFT is a feature detector and descriptor.The advantage of SIFT is that can provide high quality features than other methods. But high performance needs high computational effort.
SURF is inspired by SIFT, which the computational effort is lower than SIFT and without loss of accuracy.

1.2 Feature matching

2-a Brute-Force

Brute-Force is a quite effective way to match feature. The advantage is that it 's more quickly than LK tracking. But BF matching still have disadvantages, Which the complexity will grows proportionally depends on number of features and feature detection process will be repeated in every image.

2-b Lucas-Kanade algorithm

Due to Lucas-Kanade algorithm based on a lot of assumption, it has a lot of restriction. "pyramidal LK" algorithm is an improvement for small movements assumption.
Although LK algorithm needs more computation than BF. But it is unnecessary to repeat feature detection process like BF because LK algorithm does not use descriptor.

1.3 Motion estimation

There are three method to estimate between two cameras: frame to frame correspondences, map to frame correspondences and map to map correspondences. For different methods there are different solutions.

Sec.2 Test Plan

This test will evaluate algorithms on images taken with a moving camera.

2.1 Ground Truth

First thing is establishing ground truth. To specifying which point x in frame 1 corresponds to point x in frame 2. For planar scenes, the point x in two frame are related by a homography $H_{ij}(q) \in \mathbb{R}^{3x3}$:

$$x_j = H_{ij}(q) \cdot x_i$$

We can use semi-automatic algorithm to compute H matrix:
Using small balls as makers and manually indicate their position in the first frame. Then track them using adaptive color model and template matching. H matrix is computed from the new positions of the balls. Finally the warp is refined using image alignment.

2.2 Data Set

The testbed can be set up as video streams, that is the KITTI dataset. This dataset can be used for the evalution of stereo optical,flow visual odometry,object detection and tracking.

2.3 Evaluation Criterion

For the evaluation of performance of different combinations of detectors and descriptors, we can set a three-step test.
when test feature and closest 2 matches are got, we can judge whether it is match through a criterion:

$$\frac{d(f^C, f^A)}{d(f^C, f^{A1})} < T_{app}$$

Here f^C is the feature in image C, $f^A$and f^{A1} are two features which are closest matching to f^C. The threshold $T_{app}$is bounded by [0,1] cannot take values larger than 1. If test feature can satisfy above criterion, the rejective match is called no match. the next stage checks the identity of the image which the proposed match is coming.
If it comes from the image of related object, this feature can be tested for third stage. Otherwise, the match is rejected as a false alarm.
Stage three validates the proposed match based on geometry. It uses the epipolar constraints and builds a triplet(initial feature-auxiliary feature-proposed match) to verifies all epipolar conditions.