Camera Localization with Depth from Image Sequences

3D Vision Project Proposal
Supervised by: Torsten Sattler
March 11, 2016

GROUP MEMBERS

Marko Panjek Tian Jin Lionel Trebuchon Sonali Andani




I. Description of the project

The goal of this project is to make feature matching for 3D reconstruction and camera localization more robust by creating feature descriptors invariant to viewpoint changes.[4]
Using Structure from Motion techniques, 3D point clouds can be obtained from a dataset of 2D images. The feature descriptors used in matching images for the 3D model can then be modified to be more robust to viewpoint changes. These viewpoint invariant patches are extracted using the information of surface normals. The feature patches can then be warped to the same plane and orientation, where new SIFT descriptors can be obtained from the warped images. These new descriptors are now invariant to orientation and viewpoint.
This same process is applied to a set of new images to generate view-point invariant descriptors. Only one feature match is enough to localize the camera location of new images. Once we have the match, we use the surface normal information, main gradient orientation of original SIFT descriptors and scales of SIFT features to correctly rotate, position and scale the new 3D point cloud to the reference one. By this we also get the pose of cameras used in the second point cloud. All this can be done using only one feature match, but to make it more robust to noise and small errors in calculations, we can use a voting algorithm to determine the pose, that has the highest probability of being right.

II. Work packages and timeline

There are two parts to this project: Generating the viewpoint-invariant features for the 3D point cloud, and matching new image sets to the 3D point cloud. To do this, we will be using VisualSFM[3], libpointmatcher[1] and C++.

The first step will be to detect SIFT features in reference images and build the 3D model. Then feature patches need to be made viewpoint invariant. SIFT descriptors are formed again, but this time from viewpoint invariant patches. This will be done with VisualSFM and C++ in the following steps:

- Use VisualSFM to generate a 3D model from a set of images. This will give us the 3D points and feature matches for points in the images. Done by all of us already.
- Using the 3D points and SIFT descriptors of matched features, find a normal plane and orientation for each feature using C++ and libpointmatcher[2]. We will need to explore the best way to define a global plane and direction for each feature. Direction may be obtained by SIFT descriptors, and the plane can be estimated by a combination of normal vectors generated by VisualSFM and the location of nearby points in 3D space. This will be done in C++. This will be done by Tian and Marko by the end of March.
- Take each feature patch and warp it to the plane and direction obtained in the previous step. This will be done in C++. Sonali and Tian will do this by mid-April.

- Obtain new SIFT descriptors from the warped feature patches. These new SIFT descriptors will be our viewpoint-invariant descriptors. Sonali and Lionel will do this by end of April.

Once we have these new feature descriptors in our model, we can move on to matching new images to the model. Given at least two new images:

- Generate a separate 3D model using VisualSFM.
- Use the same method as above to generate viewpoint-invariant descriptors. We will all do the above two steps by first week of May (As the method for these parts are identical to the steps above).
- Attempt to match a descriptor from the new 3D model with the original 3D model using the viewpoint-invariant descriptors. A challenge here will be determining how the new 3D model fits the reference model from only one feature point match. We will be using the information of surface normal, main gradient orientation and SIFT feature scale to try and get to the solution. This will require experimentation to determine the best method. Lionel and Marko will do this by the third week of May.

The added information provided by these new feature descriptors can now be used for new camera localization in the reference frame. We will all work on this part, using information obtained from the previous steps, to see if we can successfully localize the camera.

III. Outcomes and Demonstration

Our goal is to be able to match new images to the model, localise the camera poses from which these pictures were taken. This will be done by using the viewpoint-invariant patches we will create. We can demonstrate this process by implementing the entire pipleline and successfully localizing new photos against our 3D model.

References

[1] libpointmatcher. http://libpointmatcher.readthedocs.org/en/latest/. Accessed:2016-03-11.

[2] libpointmatcher: Datapoint filters. http://libpointmatcher.readthedocs.org/en/latest/Datafilters/#samplingnormhead. Accessed:2016-03-11.

[3] ChangChang Wu. Visualsfm : A visual structure from motion system. http://ccwu.me/vsfm/, 2016. Accessed:2016-03-11.

[4] Changchang Wu, Brian Clipp, Xiaowei Li, Jan-Michael Frahm, and Marc Pollefeys. 3d model matching with viewpoint-invariant patches (vip). In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.