# ORB-SLAM with a Generalized Camera

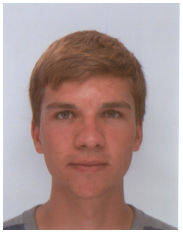
3D Vision Project Proposal
Supervised by: Peidong Liu and Dominik Honegger
March 14, 2016


GROUP MEMBERS

Lukas Bischofberger

Marius Grimm

Leonard Schai



Tobias Ulrich


## I. DESCRIPTION OF THE PROJECT

ORB-SLAM [**?**] is one of the state-of-the-art feature based simultaneous localization and mapping (SLAM) algorithms. It currently supports monocular and stereo configurations. However, due to the limited field of view (FoV) of the cameras, it would be beneficial for environment perceptions if we can have 360-degrees FoV.
The idea for this project is to extend the current ORB-SLAM algorithm to support a generalized camera [**?**]. A sensor board with up to eight synchronized global-shutter cameras is currently under construction and is hoped to be ready by the end of the semester to be used as a generalized camera to provide 360-degress FoV. For now data from the V-charge car project with 4 fisheye cameras is available.

## II. WORK PACKAGES AND TIMELINE

As a preparation and get familiar task we will need to change the implemented camera model to fit our data. As we're given image sequences taken with a fish-eye lense, we must adapt the undistorion methods in ORB-SLAM to use the corresponding camera model and its specific intrinsics.

a) Change current camera distortion model to fisheye cameras and convert fisheye data
1) 2D-2D motion estimation with generalized camera model (4/8 cameras) ([**?**], [**?**], [**?**])
2) 3D-2D NPnP (non-perspective n points) pose estimation ([**?**], [**?**], [**?**])
3) ORB-SLAM: ([**?**], [**?**], [**?**])
   a) Tracking part
      i) Replace current monocular 2D-2D motion estimation for initialization with 1)
      ii) Replace 3D-2D pose estimation for re-localization with 2)
   b) Mapping part
      i) Modify current monocular triangulation method to support generalized camera
   c) Other common technical issues
      i) Modify Frame, KeyFrame, Map classes etc. to support generalized camera
      ii) Modify Bundle-Adjustment cost function to support new feature position parametrization with Plucker line
      iii) etc.

At the very beginning we first have to adjust the camera distortion model of the SLAM-algorithm since currently pinhole-cameras are assumed. However, we only have fisheye data at our disposal.
The minimum requirements for the system to work are modifications of the current monocular SLAM under part 3) but also the implementation of the 2D-2D motion estimation with a generalized camera under 1).
After those parts are working the replacement of the 3D-2D pose estimation for re-localization under 3a) (with 2.) is going to be implemented next.

Since the different work packages are all interlinked it is difficult to divide the responsibilities. Yet, Tobias and Marius are going to start working on the Tracking part while Lukas and Leonard work on the Mapping part. After those work packages are completed the group will collectively modify other common technical issues. Those minimum requirements are planned to be working by the end of April.

The monocular SLAM algorithm is provided and implemented in C++ and ROS and the operating system of our choice is Linux. For the modifications towards a generalized SLAM algorithm the library OpenGV is going to be used.

## III. Outcomes and Demonstration

The outcome of the project is planned to be a real-time running algorithm which is able to localize, track and map more stable and with higher precision than the monocular algorithm.

As offline presentation we have rosbags available with recorded data of a monocular camera. For the fisheye cameras we first have to record the data in rosbags ourselves.
We will present our algorithm offline with the image data from multiple fisheye cameras. If already available at this point in time, also the 8-camera-system could be demonstrated offline or even online.