Visual Odometry on Smartphone

Jai Prakash

Carnegie Mellon University

Master of Science in Computer Vision

jprakash@andrew.cmu.edu

Utkarsh Sinha

Carnegie Mellon University

Master of Science in Computer Vision

usinha@andrew.cmu.edu

Abstract

In this project, we are trying to localize the camera using visual odometry. The major component of the project is to generate keyframes according to pre-defined heuristics and triangulate the points to create 3D reconstruction of the scene. The intermediate frames can be found using Perspective-n-point algorithm. In addition, we also perform local bundle adjustment over last few frames so that the localization is locally consistent. We also plan to exploit the onboard inertial sensors to get prior for the localization.

1. Introduction

Augmented reality has been around for years, yet not all problems are solved in that domain. One of the challenges being precise localization of the device in the world. Most of the augmented reality applications on smartphones are based on markers. One good example of marker-based AR is Vuforia. On the other hand, there are standalone devices like Hololens, which has number of sensors to understand the scene and localize the head mounted display in the scene.

In many of the Augmented Reality application, you do not have to understand the scene. Sometimes, just localizing the camera in the world would solve the problem. In this project we focus on localizing the phone using the camera and the inertial sensors.

2. Background

Visual SLAM vs. Visual Odometry: The focus in the visual SLAM techniques is both in reconstruting the scene and also localizing the camera in the scene. However, our main focus is just in localization of the camera. For the scope of the project, we focus only to be locally consistent. So, our system might intrduce drift over time.

3. Method

Our system contains of the following blocks.

- **Feature Extraction and matching:** We have experimented with OpenCV KLT features, AKAZE features and ORB features. The KLT features can also be used for tracking the features in the subsequent frames.

4. Results

So far, we have been working with the datasets available online. The results are illustrated on the Middlebury Temple dataset [2]. We are able to recosntruct the temple structure using the two keyframes (handpicked for now). Once the reconstruction is done, we are able to recover the camera poses as illustrated in the figure xxx
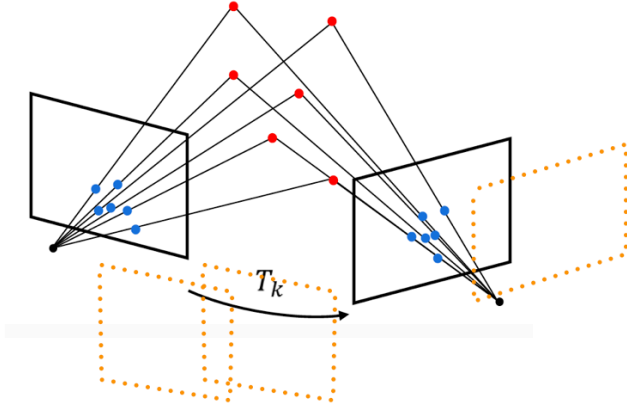
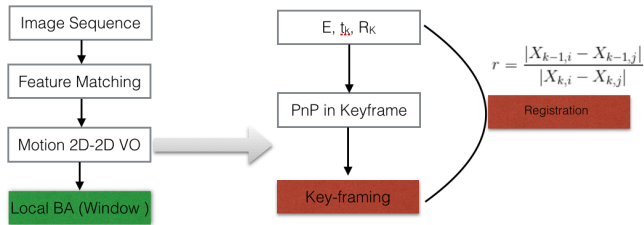Figure 1: A visualization of the Visual Odometry system

Figure 2: Block diagram of the system. The red blocks are implemented as a part of Geometry based vision project. For this project we are focusing on implementation of local bundle adjustment shown in green

5. Future Work

- Evaluation of various features.

- Integration of inertial sensor in the pipeline

- Integration of Google Ceres-solver for bundle adjustment.

References

[1] Szeliski, Richard. *Computer Vision: Algorithms and Applications*. 1st ed. London: Springer-Verlag, 2010. Print.

[2] Temple dataset `http://vision.middlebury.edu/mview/data/`