УДК 004.896 + 004.93'11 + 519.612

ОПТИМИЗАЦИЯ В РЕАЛЬНОМ ВРЕМЕНИ ОЦЕНКИ ПОЗЫ С ИСПОЛЬЗОВАНИЕМ ОДНОВРЕМЕННОЙ ЛОКАЛИЗАЦИИ И КАРТИРОВАНИЯ ДЛЯ МОБИЛЬНЫХ РОБОТОВ

REAL-TIME OPTIMIZATION OF POSE ESTIMATION USING SLAM FOR MOBILE ROBOTS

Махмуд Ж (Национальный исследовательский университет ИТМО)
Научный руководитель – доцент, Колюбин С.А.
(Национальный исследовательский университет ИТМО)

Abstract. This article states an algorithm for performing optimization of the estimated pose of a mobile robot (Localization). The optimization runs in real-time, and it includes also the optimization of the Mapping process in SLAM, by availing the sparsity of the linearized system and applying the Schur complement trick with Cholesky Decomposition.

Introduction. Mobile Robots designed for performing specific tasks that include moving inside an environment. This environment might be unknown to the robot. In this case, the robot must discover the surroundings and carry decisions based on this gathered information. Simultaneous Localization and Mapping (SLAM) is an important discipline in Robotics. It started in the mid-90th of the twentieth century. The massive development in the field of sensor manufacturing, Vision processing, High-speed features detection, and Programming capabilities helped the evolution of SLAM applications. SLAM can provide vision and perception for mobile robots. This article states mathematical and algorithmic techniques to optimize the pose estimation process in SLAM for mobile robots in Real-time.

Main Part. The pose of a robot can be interpreted in a multidimensional space. An arbitrary pose point in this space should describe the position and the orientation of some distinguished spots in the mobile robot (Joints or Center of Mass). For example, a six-dimensional Euclidean space (Coordinates and Euler angles) can be used to represent the space of all possible poses of the CoM of the Robot. This Euclidian space contains singularities in terms of Euler Angles. Hence, the pose space better to be a combination of an Euclidean space (for Coordinates) and a Manifold (for Rotation). In this space (Euclidean combined with Manifold), one needs to reformulate the jacobians' relations and the model of the robot's pose. Specifically, no need to transfer the kinematics and dynamics of the robot into this space. The task here is to optimize the estimation of the pose, and the results of this procedure can be interpreted as an input for the dynamic model of the robot or the navigation system.

Pose Estimation is a problem that has many solutions. In mobile Robots, the pose is estimated by solving the dynamic model, this depends on how realistic is our model. One can also use sensing equipment: Inertial Measurement Unit (IMU) through integrating the acceleration and the gyroscopic velocity along a specific period or by using a camera to track visual features (edge detection, classification of objects, Line detection) for some known 3D absolute coordinates by solving PnP problem. These methods differ by accuracy, speed, and financial cost. This article assumes that the estimation method is done using sensing equipment (camera with/without IMU) combined with solving the dynamic model of the mobile robot.

Simultaneous Localization and Mapping (SLAM) is the solution to a challenging issue in Robotics. How to know the precise position of the robot in an unknown map and draw a map based on the relative motion of this robot? This solution can be described mathematically as a probabilistic problem, by representing the pose of the robot and the 3D map points as probabilistic variables. there are two types of solutions for SLAM. Filter-based and feature-based solution. The feature-based solution is proven to provide better accuracy for the same computational complexity. As stated earlier, the estimation process is done using sensing equipment (camera with/without IMU) and the dynamic model of the robot. Camera and IMU sensors are cheap and provide a rich fusion of inertial and visual information. The task to consider is to optimize the estimated pose using a feature-based solution of SLAM.

In a feature-based approach, visual features represent the 2D coordinates in the image frame of some interesting points in the scene. Knowing the camera projection matrix by performing an offline calibration, one can estimate the 3D coordinates in the world frame. Noting that the depth can only be estimated using a depth sensor or a stereo vision system. The IMU provides the pose by integrating Inertial data along time, a recent approach is to use the Preintegration technique, which only depends on the latest pose and Inertial data between the last pose and the current state. Hence, there is no obligation to preserve all the inertial information from the starting point of time to calculate the integration. The estimation process can be summarized as follow: Visual features extraction and solving PnP, in case of blurry data, we use the inertial data of IMU to compensate for the visual error. And in the case of inaccurate biases estimation of IMU, which affects the Inertial measurements, we estimate the pose using the dynamic model of the robot. The result of the estimation will be optimized.

The optimization based on how accurate our estimation is. This can be measured by describing the probabilistic estimation in terms of an error function. More specifically, the least-squares error function can be derived from probabilistic measurements considering a Gaussian noise on the data. This error function is non-linear and non-convex. This error function needs to be minimized. The pose of the current camera scene is described using six parameters (on a Euclidean space and on a Manifold). The IMU data can use the camera pose by a simple transformation between the camera system and the IMU system. Also, we need to optimize the velocity calculated by IMU. The dynamic model includes a number of parameters that also needs to be optimized. This leads to a set of variables (6 - 25) included in the error function. The SLAM also builds the map of points, which are also represented as probabilistic parameters. Every single point on the map has its own probability, which can be included also in the error function using 3 variables (coordinates) in the world frame. Usually, the map consists of a huge amount of points (500 - 5000) at one scene (at a time). In a nutshell, we need to minimize an error function that includes up to (5000) parameters in real-time. The error function can be interpreted by calculating the projection error of the tracked features, also the IMU error in terms of estimated velocity by the IMU and the current estimate, combined with the robot model calculation.

The non-linear error function is linearized using the Taylor series. To perform the linearization, we need to calculate the Jacobians on the Euclidian space for pose coordinates and point coordinates and Jacobians for pose orientation on a manifold. The resulting Jacobian helps in describing a linearization of the system, also helps in finding the Hessian of the system. Here is the key of Optimization. Even that the system is huge to solve, it is known that the Hessian Matrix of the considered system is sparse. This is because of the independence of the variables in the error function. The sparse systems have interesting features. Each time we get a fusion from the sensor and the robot model, we need to solve a huge sparse linear system, concerning high-frequency data fusion. The system is solved using Gauss-Newton or Levenberg-Marquardt method to converge to an optimal solution. The Schur complement trick and Cholesky decomposition can provide an efficient paradigm to solve the resulting linear system.

Results. The system is more robust, because of the estimation based on robot model, in case of vision failure (blurring – lightening), or IMU bias error. The optimization of Pose estimation is already running in real-time (the system to solve is not big). The main advantage is in Optimizing the Map points by solving a linear system (size of 5000 Map points and 250 parameters of 10 subsequent poses) in only 4 seconds using library G2O and “CHOLMOD” solver. This solver uses the sparsity and apply the Schur complement trick. The trivial inversion method of this system matrix is slower by at least 5 times.

Махмуд Ж (автор) Подпись

Колюбин С.А. (научный руководитель) Подпись