**An ICP-based Navigation System for Human Support Robot**

Zhang Haibin

Report of Research Student

Kubota Laboratory

As concern for Japan’s aging populating and shrinking workforce continues to mount, many researches are focusing on nursing and healthcare-related robots to help ease the burden on society. In order to successfully integrate robots into the home environment to coexist with family members, three features are considered essential: safe interaction, a compact and lightweight body, and a simple interface. In this paper, we introduce a robot that possesses these qualities: the human support robot (HSR). HSR is designed to support human activities to assist with independent living in the home for handicapped people. In these paper, we proposed a way to estimate the position of robot while moving as the first part of our navigation system. Using RGB image and depth maps captured by Xtion, we calculate the 3D coordinate’s position by a pair of features based on ICP algorithm. Lastly, we use the TUM benchmark to validate our system’s accuracy.

1. **Introduction**

The rapid decline of the Japanese population due to an aging demographic and falling birthrate poses serious social issues. Meanwhile, the shortage of nursing and healthcare providers are also worthy the society concern. Not just in rural, but also in urban areas, Japan is experiencing a “super-aging” society. People aged 65 and older in Japan make up a fifth of its total population, estimated to reach a third by the year 2050.

Toyota Partner Robot is the name given to a series of robots developed by Toyota to embody “kindness” and “intelligence”, which will be able to assist with human activities in applications such as living assistance, healthcare, manufacturing, and mobility.

The HSR, described in this paper, is designed for assisting people in their everyday activities (“Living assistance”) and improving living conditions and overall quality of life of elderly or disables individuals by fetching and manipulating objects found in home or care facility. For the early stages of the development, disabled individuals are targeted since they have the greatest need for receiving assistance in their everyday activities. As development progresses and user trials are completed, the goal is to expand functionality to accommodate the more general needs of the elderly and ultimately society at large.

HSR has three basic modes: Pick-up, Fetch, and Manual Control. Imaging a family service scene, when people hope to controlled HSR to fetch some a milk box on the table through voice commands or a simple graphical user interface via common handheld touchscreen-enabled devices, such as tablet PCs and smartphone, HSR needs autonomous navigation to reach the object's location, and hopefully is an optimal path.

For our whole robot navigation system, we use one figure below as explanation. The figure is a complete robot navigation system flow chart. Depending on robot’s environmental sensors to obtain environmental information, we can extract image feature and set those features as landmark. Ultimately, we get the environmental model. while the left part briefly introduce robot localizing and map reconstruction, the right part is that robot navigation and control for a specific target. With the map model, mobile robots can then navigate autonomously, exploring in the map and deciding the optimizing way.

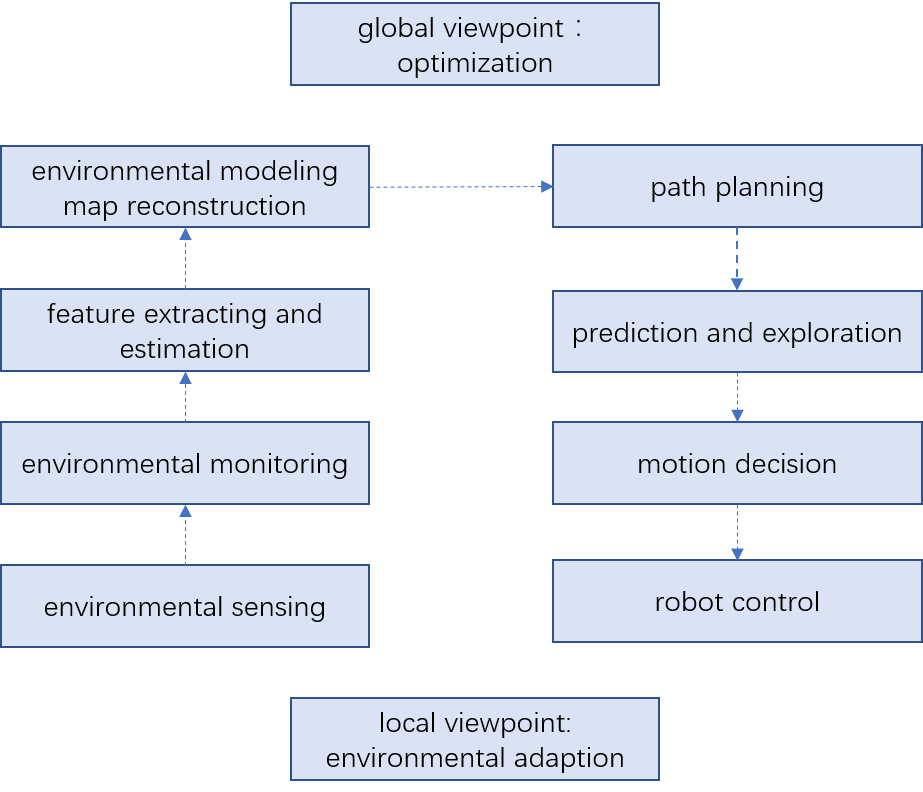


Fig. 1 Flowchart of robot navigation

Position estimation for a mobile robot using vision and odometry. Location is basic to navigation. Various techniques have been described for estimating the position and orientation of a robot. Basic processing approach is to use camera’s motion instead of robot for calculating position.

Therefore, the main problem point is how to estimate camera’s motion depending on sequence of images. In order for a mobile robot to navigate through an unknown environment, it first has to solve the problem of SLAM. For SLAM, the robot has to estimate a map of the environment while at the same time localizing itself with respect to this map. For the main problem point in this research, we propose a way to solve self-location of robot.

1. **Background**

As mentioned above, HSR is being developed to support independent living for people with limited arm or leg mobility. Clearly, the demand for this type of support exists, and the demand far exceeds the supply. Therefore, we propose robots as a practical solution to the demand for independent living support. In 2006, Toyota developed the Toyota Delivery Robot to fetch and deliver a variety of needed objects at the request of hospital patients. This robot was successfully demonstrated at the Japanese Circulation Society. The newly-developed HSR is the successor of the Toyota Delivery Robot and designed to improve functionality and workspace while shrinking the footprint to better suit the home environment (Fig.2).

Still, in the table 1, we show the main components of the various parts of the HSR and the installed sensors.

图片包含 室内, 杯子

已生成高可信度的说明

Fig. 2 Human Support Robot

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | Human Support Robot (HSR) | | | | |
| Size | Footprint |  | | | |
| Height(min/max) | 1005/1350mm | | | |
| Weight | About 37kg | | | |
| Moving base | Sensors | * Laser Ranger Sensor * IMU | | | |
| Arm | Length | 600mm |  |  |  |
| Payload(recommended/max) | 0.5/1.2kg |  |  |  |
| Grip | Sensors | * Gripping force sensor * Wide-angle camera | | | |
| Head | Sensors | * RGB-D sensor x1 * Stereo camera x1 * Wide-angle camera x1 * Microphone array x1 | | | |
| Body | Expandability | * USB x3 * VGA x1 * LAN x1 * Serial x1 * 12V-0.5A ouput x1 | | | |

Table 1: The main components of the HSR

The Asus Xtion Pro Live sensor is a combination of an RGB camera with a depth camera. Its underlying technology will be familiar to many, because it uses the same sensing hardware as the Microsoft Kinect. Unlike the Kinect, the Xtion is USB-powered and is much smaller, making it more suitable to Botball-style robotics.

The Xtion is capable of generating a 640×480 image where each pixel contains a distance from the Xtion camera (precision 1mm). This image can be updated at 30fps. In this paper we will primarily focus on the RGB color data and depth maps.

|  |  |
| --- | --- |
| Field of View | 58° H, 45° V, 70° D |
| Depth Image Size | VGA (640x480) : 30 fps  QVGA (320x240): 60 fps |
| Resolution | SXGA (1280\*1024) |
| Interface | USB2.0 |

Table 3: Asus Xtion PRO LIVE technical specification

1. **System Overview**

In Fig. 3, we illustrate an overview of our system for estimating the robot’s position by ICP algorithm. The input data to the system are an RGB color image and a depth image captured using a Xtion PRO LIVE sensor mounted on the head of HSR. Then the processing is like below:

1. For the new arrival of the current frame, extract the key points and descriptors.
2. If the system is not initialized, take this frame as a reference frame, calculate the 3D position of the key points according to the depth map, and return to the first step.
3. Estimate the motion of the reference frame and the current frame
4. Determine whether the above estimation is successful or not
5. If successful, the current frame as a new reference frame, return to the first step. If failed, record the number of consecutive lost frames. When the continuous loss of more than a certain number of frames, set VO state is lost, then set algorithm ends. If not exceeded, return to the first step.

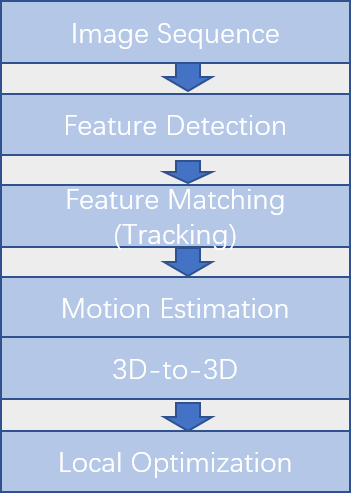


Fig. 3 Pipeline of visual odometry

We consider the use of feature point method to achieve vision odometer. Its job is to calculate camera motion and feature point locations based on the input image. First of all, for simplicity, we only consider the pose estimation between two adjacent frames. After that, we save the feature points into a local map and calculate the current frame-to-map positional relationship.

We first define the concepts of reference frame and current frame. With the reference frame as the coordinate system, we match the current frame with it and estimate the motion relationship. It is assumed that the transformation matrix of the reference frame relative to the world coordinate system is , and the current frame relative to the world coordinate system are , so the estimated motion can be calculated due to .

From time t-1 to time t, we take t-1 as reference to find the motion at time t.

3.1 Feature extraction and matching

ORB feature consist of two parts, which are key-point and descriptor. So extract orb features are divided into the following two steps.

1. Fast corner extraction: find the corner of the image. Comparing with the original FAST, the main direction of the feature point is calculated in ORB, which adds rotational invariance to the subsequent BRIEF descriptor.
2. BRIEF descriptor: Describe the surrounding image area of the feature point extracted in the previous step.

Next, we make a feature match between two frames of the image. Feature matching is a crucial part of visual slam, which can determine the corresponding relationship between the currently seen signpost and the previous signpost. One of the simplest methods of feature matching is Brute-Force Matcher. That is, for each feature point and all feature points measure the distance between descriptors, and then sort, only take the nearest one as the matching point. We use the Hamming distance as a descriptor distance, and we can express the similarities between the two features.

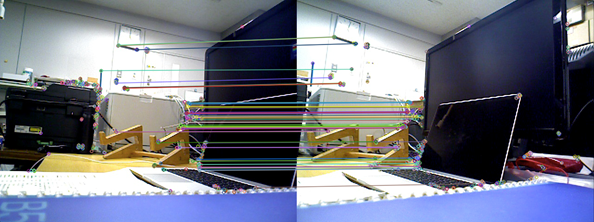


Fig. 4 Feature matching of two frames

3.2 Iterative Closest Point

We get the corresponding relationship between the feature points of the image, so we can use 3d-3d matching points’ pair to estimate the motion of camera.

If we have two group 3d matching points,

to make . We can use ICP to solve it and get pose estimation of to two group matching points.

For the errors of points,

.

Building the least squares problem,

.

Steps to slove this problem are in below. If we define points’ centroid,

then,

Because of , therefore,

The main Steps are below:

1. Calculate two group points’ centroid , and then for every point distract the centroid,
2. Solve the optimization problem,
3. Use SVD to solve this question.

Suppose that,

Therefore, when are full rank, we calculate for t, .

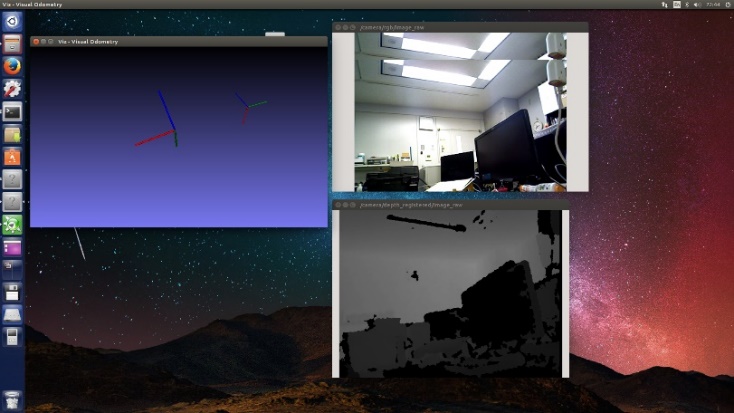
In the Fig. 4 below, we show the camera's motion trajectory based on color and depth maps, which is the change of the coordinate of the current frame relative to the frame of reference.

Fig. 5 The estimation of position using RGB-D image

**4 Experiment**

We use the benchmark for the evaluation of RGB-D visual odometry system. The dataset contains color images, depth maps, and associated ground-truth camera pose information. the data sequence “freiburg1\_desk” contains several sweeps over four desks in a typical office environment. Further, we proposed an evaluation metrics that can be used to assess the performance of a visual odometry system.

Additionally, the ground-truth file included contains the ground truth trajectory stored as a timestamped translation vector and unit quaternion (format: timestamp tx ty tz qx qy qz qw).

|  |  |
| --- | --- |
| Duration | 23.40s |
| Duration with ground-truth | 23.35s |
| Ground-truth trajectory length | 9.263m |
| Avg. translational velocity | 0.413m/s |
| Avg. angular velocity | 23.327deg/s |
| Trajectory dim. | 2.42m \* 1.34m \* 0.66m |

Table 2: Basic information of “freiburg1\_desk” in TUM Benchmark

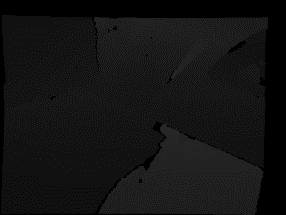
 

Fig. 6 Example of freiburg1\_desk dataset

For a navigation system, additionally the global consistency of the estimated trajectory is an important quantity. The global consistency can be evaluated by comparing the absolute distances between the estimated and the ground truth trajectory.

We calculate the cumulative error of the true trajectory and the estimated trajectory according to the time.

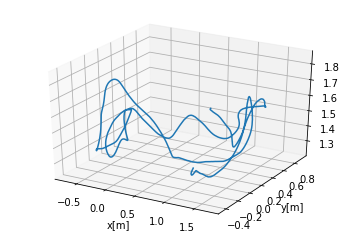


Fig. 4 Visualization of the ground-truth trajectory on the “fr1/desk1” sequence.

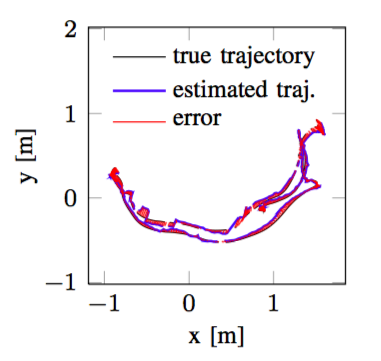


Fig. 5 Visualization of the absolute trajectory error on the “fr1/desk1” sequence.

**5. Conclusion**

We see that the visual odometer can estimate the camera movement and the location of the feature points over a period of time, but this partial approach has obvious drawbacks:

1) Easy to lose. When the key frame moves too fast, it will cause the feature point to be lost. Once lost, we either wait for the camera to turn around, save the reference frame and compare with the new frame, or reset the entire VO to track the new image data.

2) Trajectory drift. The main reason is that the error of each position estimation will be accumulated to the next estimation, resulting in inaccurate long-time trajectory. Larger local maps may alleviate this phenomenon, but it is always there.

**6 Future Works**

Compared with human, a robot’s ability to autonomously execute tasks is still very limited. For the whole navigation system, we hope that, when people set the target location and orientation, the HSR is able to autonomously move to the new position, detecting and avoiding obstacles as needed. It is necessary to have previously created a 3D map of the environment using the equipped Laser Range Finder and Xtion.

Until now, we have only looked at the part of the robot vision odometer. As part of a system that serves the entire family for robot navigation control. Next, using the environment map obtained from xtion, we study the robot's path planning part of the map and expect the optimal path.

In the following, we will conduct research on other aspects of the navigation system along the route proposed in the previous article. We hope that robots can fulfill their tasks according to human instructions as soon as possible so as to serve human beings to reduce their workload. In the future Japan's aging more serious, there is more room for application.

**Reference**

[1] K. Hashimoto, F. Saito, T. Yamamoto and K. Ikeda, "A field study of the human support robot in the home environment," 2013 IEEE Workshop on Advanced Robotics and its Social Impacts, Tokyo, 2013, pp. 143-150.

[2] Krystof Litomisky, Consumer RGB-D Cameras and their Applications, University of California, Riverside.

[3] K. Nagahama, H. Yaguchi, H. Hattori, K. Sogen, T. Yamamoto and M. Inaba, "Learning-based object abstraction method from simple instructions for human support robot HSR," 2016 IEEE International Conference on Advanced Intelligent Mechatronics (AIM), Banff, AB, 2016, pp. 468-475.

[4]J. Sturm, N. Engelhard, F. Endres, W. Burgard and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura, 2012, pp. 573-580.

[5]高翔, 张涛, 颜沁睿, 刘毅, 视觉SLAM十四讲：从理论到实践, 电子工业出版社, 2017

[6]http://www.brain.kyutech.ac.jp/~hma/wordpress/robots/hsr/

[7]https://en.wikipedia.org/wiki/Aging\_of\_Japan