UBFC
UNIVERSITÉ
BOURGOGNE FRANCHE-COMTÉ

Robotics Engineering

Technical Survey Report

Fine Positioning (Visual Servoing)

Authors :
Selma BOUDISSA
Danie Jianah SONIZARA
Meldrick REIMMER

Master students in Computer Vision
University of Burgundy

November 5th, 2018

Contents

1 Introduction **2**

2 Objectives **3**

3 Visual Servoing Techniques **4**
3.1 Position Based Visual Servoing (PBVS) 4
3.2 Image based Visual Servoing (IBVS) 5
3.2.1 Detecting and tracking AR Tags 5
3.3 2 1/2 D Visual Servoing 6
3.4 Shortest Path Visual Servoing (SPVS) 6

4 Results & Discussions **7**
4.1 PBVS 7
4.2 IBVS 8
4.3 2.5D VS 8
4.4 Shortest path Visual Servoing (SPVS) 9

5 Conclusion **10**

A Visual Servoing Platform (ViSP) **12**

1 Introduction

This is a survey on Visual Servoing regarding its techniques. In this report, we examined several techniques in order to achieve a task given being Fine Positioning. Our robot is to be able to fine position autonomously at a specific location. In order to achieve this, we compared the results based on each technique in the quest of finding the best suit solution for our task.

Visual Servoing (VS) also known as vision-based robot control, is a method which uses feedback information extracted from a vision sensor such as a camera to control the motion of a robot. Its techniques are commonly used for positioning a robot in a desired position (typically with the goal of grasping some object) or following targets in movement.[1]

The task of Visual Servoing (VS) for robot manipulators is to control the pace of the robot's end-effector by extracting real-time visual features of the image. In other words, the robot should be able to extract from the image all the environment's features it needs to execute its task. As we use only one camera, the previous task can be done in two different ways: - Eye-in-hand or endpoint closed- loop control, the camera is attached to the last joint of the manipulator close to its end-effector and observing the relative position of the target - Eye-to-hand or end-point-loop control, the camera is fixed in the world and looking to the end-effector and observing the target.

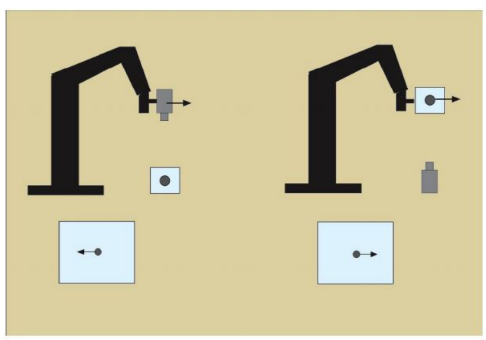

Figure 1: Camera Position. Left side being Mounted (Eye-in-Hand) and right side being Fixed (Eye-to-Hand)[3]

The first paper published about VS is from 1979 by SRI International labs.[2] Visual Servoing is a very large field of research in Robotics, and to build such system the knowledge from different research domain (Computer Vision, robot modeling ...) is essential.

In a visual servoing system, the robot should be able to extract from the image, all the environment's features which it needs to execute its task.

2 Objectives

The main purpose is to use information acquired by vision sensors (cameras) for feedback control of the pose/motion of a robot. The task given is for us to make our robot to position itself at a given location to enable another task to be done. We will be examining various techniques used and related papers to see which technique will be more accurate to solve our task.

We performed the following steps in being able to find the suitable solution to the task given.

- Discovering the various techniques
- Examining on the techniques
- Discussing on the results techniques
- Choosing one technique to solve the task.

3 Visual Servoing Techniques

In this section we will be describing some of the techniques used in Visual Servoing.

3.1 Position Based Visual Servoing (PBVS)

In scientific literature, the Position Based Visual Servoing (PBVS) is one of the main Visual Servoing techniques. With the position based method, a 3D camera calibration is required in mapping the 2D data of the image features to the Cartesian space data. This is to say that intrinsic and extrinsic parameters of the camera must be evaluated. In position-based visual servoing (PBVS), full 3D object pose is estimated using a model of the object, and the control error is then defined in terms of the current and desired poses. This method makes it possible to attain straight line motion resulting in a minimum trajectory length in 3D. PBVS provides a better response to large translational and rotational camera motions.

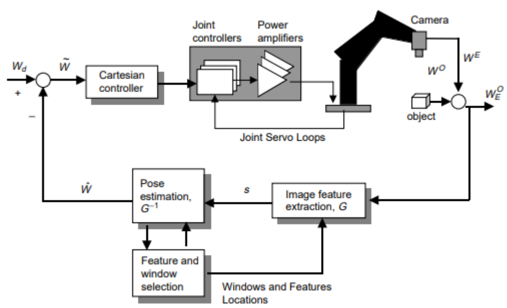

Figure 2: Structure of Position Based Visual Servoing (PBVS) [4]

The scheme above describes the structure for the PBVS technique which has common points with IBVS. The feature and window selection square in both techniques use current information on the camera, the joint-servo interface simplify the integration and the portability of the visual servo-control. For the PBVS the features extracted by the vision system are used for the pose estimation which is compared to the desired relative pose to calculate the relative pose error. PBVS is more sensitive to camera and object model errors, it provides no mechanism for regulating the features in the image space. [4] provided a feature-selection and switching mechanism. Pose estimation is a key issue in PBVS.

3.2 Image based Visual Servoing (IBVS)

This technique consists of the extraction of the features of the image without interpretation. Those features are primitives such as line or points. In the IBVS control the error signal is defined directly in terms of image feature parameters.

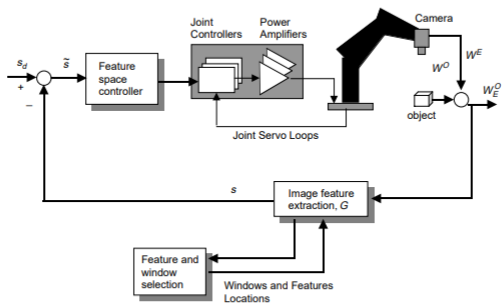

Figure 3: Structure of Image Based Visual Servoing (IBVS)[4]

In the above figure the robot is controlled directly by the image parameters. These feature parameters extracted is compared to the desired feature parameter.[4]

The advantages of the IBVS technique no prior information is required and also no 3D reconstruction of the target is necessary which reduce the processing time of the algorithm used, increase the sample rate and decrease the computer

3.2.1 Detecting and tracking AR Tags

The AR Tags can be an example of image used for the IBVS. The detection of an object will be much easier by adding a tag on it. A detailed description of this technique is presented in the 'ROS example book volume 2, chapter 10' [5] has a follower application for the robot, which is easier to follow and more robust under different conditions of lighting and/or background.

We can create our own AR tag by using the package 'rbx2_ar_tags' by running the creqteMarker included in the 'qr_track_alvar' package we will have created an image with the resulted AR tag. The implementation is fully described in [5] chapter 10. The previous technique describe in the ros book volume 1 was using face detection and/or color. The main advantage is using AR tag the robot will be able to detect it under any lighting conditions or angle of view from the camera.

3.3 2 1/2 D Visual Servoing

This method combines PBVS and IBVS it is an hybrid approach which supposed to avoid the drawbacks of the main techniques such as the problem of local convergence for the control law in the IBVS. [6]

It is based on the estimation of the partial camera movement from the actual to the desired position. This approach is well established from an analytic point of view. The controls camera rotational and translational degree of freedom (DOF) are decoupled.

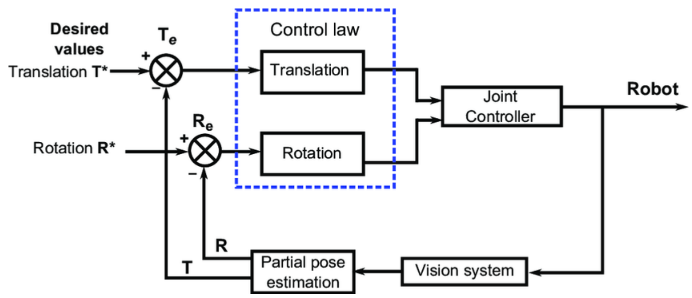

Figure 4: Structure of the 2 1/2 D visual servoing process[7]

At the first time to recover the relative pose between the current and the desired position we need to use a pose estimation technique which involve lots of complex mathematical calculation through a rotation matrix in order to extract the unit vector of the rotation axis.

3.4 Shortest Path Visual Servoing (SPVS)

The shortest path visual servoing is a method presented by Kyrki and Kragic [8] with the goal to solve the inherent problems of image-based and position-based servoing. This approach guarantees a shortest Cartesian trajectory and object visibility. The analysis of the trajectory can be difficult, mainly due to the partitioned control. However, knowing the Cartesian trajectory is essential to avoid the risk of joint limits. By finding the shortest path servoing, this strategy minimizes the path length by using straight line trajectory and avoids the possibility the target leaves the camera field of view.

4 Results & Discussions

4.1 PBVS

In position-based visual servoing (PBVS) a model of the object is used to estimate the full 3D object pose. The computational complexity of pose estimation is one of the main disadvantages of this method. Another well-known problem, pointed out by Chaumette [9] ,is the constraint on the visibility of the object.That is the target may leave the camera's field of view

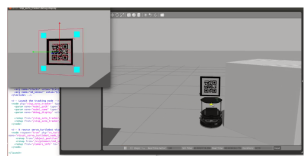

Figure 5: Simulation in the Gazebo world Posed Based Visual Servoing

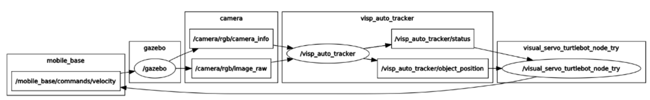

Figure 6: The resulted rqt_graph of the simulation using gazebo

The starting position of the robot allows the camera to frame a wide area of the workspace, when the target is recognized by the vision system, the controller starts to govern the robot till the target is reached. The limit of the simulation is the position of the marker within the gazebo world. In fact, the marker position as an influence on the robot detection: if it's too far or too close the robot won't be able to detect the QR code and then won't be able to rotate around the area to detect it which can be explained by the lightning conditions and angle of view from the camera. In the above picture the marker's position was good enough to allow the robot to rotate and analyze the area which enable the detection of the tag and reach the desired position in front of it.

The rqt_grph has been obtained from the gazebo simulation using the package 'visp_auto_tracker.

4.2 IBVS

Such as the PBVS this approach also have been advantages and disadvantages. IBVS introduces high coupling between translation and rotational motions of the cameras which bring to a camera retreat problem, solved in the hybrid VS technique (2.5D VS). Also the system does not allow local convergence for the control law.

Despite the previous disadvantages there is a good point about this approach no 3D reconstruction of the target is necessary leading to a reduction of the processing time of the algorithm used and an augmentation of the sample rate which solve part of the computer vision problem.

4.3 2.5D VS

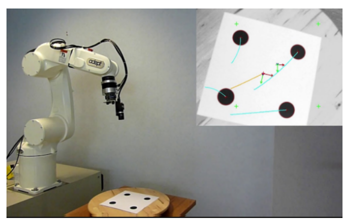

Figure 7: The result obtained in [10] using 2.5D VS approach

This technique has many advantages such as:

- the control of camera being decoupled for the rotation and the translation which solve the camera retrieved problem
- no image singularities and local minima are necessary
- a full pose estimation is not required

Despite the advantages presented as the other VS techniques there is also drawbacks such as:

- the convergence is limited to the neighborhood of the desired position
- it does not ensure that the target won't leave the camera's view field.
- this approach is very sensitive to the noise

4.4 Shortest path Visual Servoing (SPVS)

However, comparison between IPVS, PBVS and SPVS was done based on the simulation experiments presented by Kyrki and Kragic [8] in their paper in which they proposed the following control task: optical axis rotation, optical axis translation, camera y-axis rotation, feature plane rotation and general motion".

The experiment revealed that : - on optical-axis rotation revealed that the SPVS method does not suffer from convergence problems with large optical axis rotations, - on y-axis rotation, where each of the methods has a different control function, shows that comparable performance between all methods. In feature plane rotation, the Shortest Path method converges faster than PBVS. However, for all methods the image features remain in the field of view throughout the servoing, and all are therefore able to converge. In the general motion task, it is shown that the visibility efficiently can be controlled with only marginal difference in performance while PBVS exhibits the problem of image features leaving the field of view .

5 Conclusion

To conclude, we have been able to see the different solutions available to achieve our task. All the techniques described, theoretically, can give satisfactory results, but each of the techniques has its advantages, inconveniences and limitations.

The 2.5D VS and the Shortest Path shows good results in theory, but it also involves a huge understanding principally on mathematical formulas.

Given the time frame for this project and based on the comparison of each technique, we decided to implement the IBVS and/or PBVS to achieve our goal. Theses methods were chosen based on the environment, we will be implementing the technique and the hardware specifications available at hand. We will try each technique to see which of the two fits best the task. If one technique should have a major drawback we will then implement the other. At the end, we be able to achieve the task given.

References

[1] R. Santos,”Evaluation of Visual Servoing techniques for Robotic Applications”2007

[2] ”Visual Servoing”,2018.

[3] Prof. Alessandro De Luca,”Visual Servoing”.

[4] J-S. Farrokh,”Visual Servoing:Theory and Applications”,pp. 15-5.

[5] R. Patrick Goebel,”ROS by example Volume 2, Packages and Programs for Advanced Robot Behaviors”

[6] E. Malis, François Chaumette, S. Boudet ”2 1/2 D Visual Servoing”.

[7] Naresh Marturi,”Block diagram depicting 2 1/2 D visual servoing”

[8] Kyrki, V., Kragic, D., and Christensen, H., Oct. 2004, “New shortest-path approaches to visual servoing” in IEEE Int. Conf. on Intelligent Robots and Systems, Sendai, Japan, pp. 349–355.

[9] F. Chaumette, “Potential problems of stability and convergence in image-based and position-based visual servoing,” in The Confluence of Vision and Control, ser. Lecture Notes in Control and Information Sciences, no. 237. Springer-Verlag, 1998, pp. 66–78

[10] ”2.5-D Visual Servoing: cog c*tc thetaUz as feature”, video on youtube (https://www.youtube.com/watch?v=RX2R2GLs3YE)

Appendix A Visual Servoing Platform (ViSP)

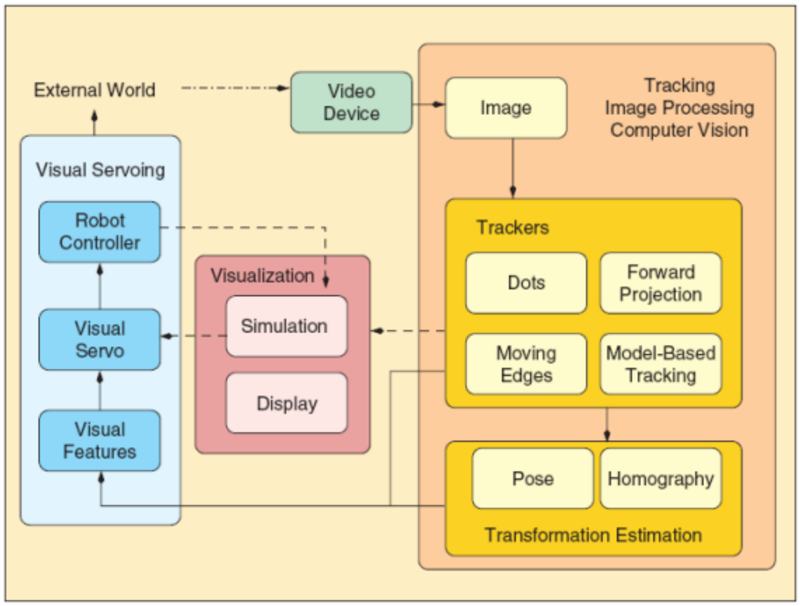

Figure 8: Structure architecture for ViSP library