Enhancing the usability of the MANUS manipulator by using visual servoing

ir. A.H.G. Versluis, ir. B.J.F. Driessen, ing J.A. van Woerden (TNO TPD)
dr.ir. B.J.A Kröse (University of Amsterdam)

TNO TPD, PO-BOX 155, 2600 AD Delft, The Netherlands
Email: versluis@tpd.tno.nl, tel: +31 15 2692762, fax: +31 15 2692111

Introduction

Manus is a wheelchair mounted manipulator, meant to assist severely handicapped people in carrying out all day living tasks, such as eating, drinking, scratching etc. The manipulator has six rotational degrees of freedom for positioning and orienting the gripper, one degree of freedom for opening and closing the gripper, and one (optional) degree of freedom for lifting the entire manipulator. MANUS is designed to operate in an unstructured environment, where the user is responsible for driving the robot to the required position (telemanipulation). Compared with industrial robots, MANUS has low accuracy and low repeatability[1]. These deficiencies have to be compensated by the end-user who guides the system to the required position, based on visual observations (dotted lines in fig.1).

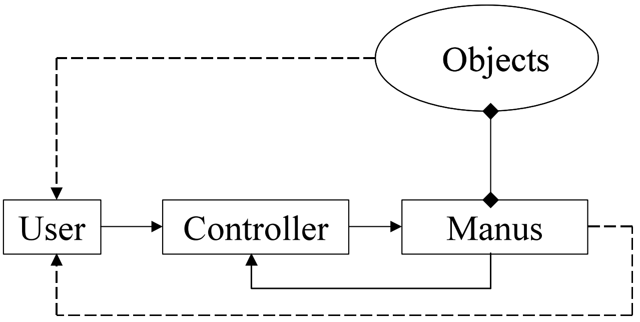

Figure 1 Traditional MANUS control scheme.

Although users are able to manipulate many objects using this control architecture, the cognitive load on the end-user while accomplishing a certain task can be serious (especially when accurate motions are required). In this research it is investigated how a camera system, mounted on the gripper of the MANUS (eye-in-hand configuration) can help the end-user in carrying out a certain task (see fig.2). The MANUS philosophy that the end-user should always be 'in control' will be a design constraint for the visual servo: it is not the intention to build a *fully autonomous* system. The following constraints should be met:

- the visual servo has to work in strong collaboration with the end-user
- the system must operate in unstructured / unknown situations, allowing only very limited usage of a-priori information.

In this paper, several approaches to visual servoing are considered and a solution for integrating visual servoing in the MANUS controller is proposed. The performance of the visual servo is demonstrated in a test case where the objective is to pick up a colored beaker.

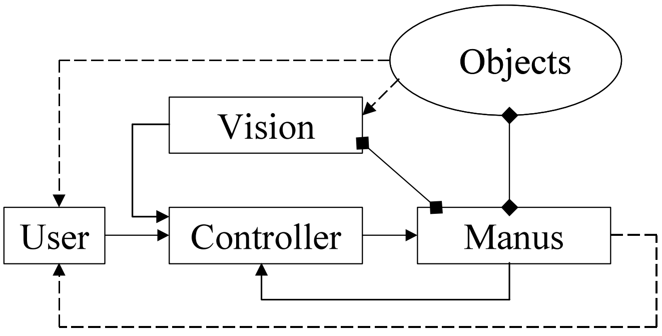

Figure 2 Vision based control scheme.

Visual servoing

Two widely used architectures for visual servoing are *image based visual servoing* (IBVS) and *position based visual servoing* (PBVS) (see fig. 3) *[1-5]*.

---

[1] This is a result from the high friction and backlash, caused by the design choice to put the motors and gearbox in the main shaft of the manipulator, in order to keep the size and weight of the MANUS low.

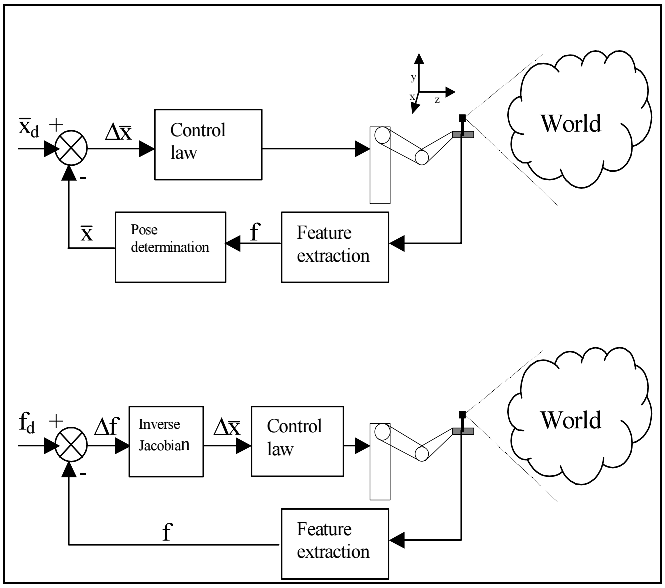

Figure 3 PBVS (upper) vs. IBVS (lower)

In PBVS a 3D world-model is constructed by the vision system. This 3D world model is used by a path-planner, which will guide the arm to the desired position. Calculating a 3D world model can be carried out using different techniques (stereovision, laser triangulation, usage of a-priori knowledge about the objects, etc.). These techniques are not preferred, since they need extra hardware, or explicit a-priori information (which is in most cases not available in unstructured environments).

In IBVS, the required position of the manipulator is defined in terms of image features (e.g. required location of corners of a cube). The visual servo continuously calculates the actual values of these features, and calculates a correcting control action in feature space. Feature errors are translated to robot co-ordinates (defined in Cartesian space or joint space). This requires the calculation of an image Jacobian (J), defined in the following equation:

$$\Delta f = J(\theta).\Delta x,$$

where Δf represents the position error in feature space, and Δx represents the position error in robot space.

As stated earlier, the Manus has to be able to function in unstructured environments and with various objects, putting high constraints on the vision system *[6]*. To reduce the complexity of these constraints, the objects can be equipped with markers. Although this approach has proven to work well *[7]*, it limits the versatility of the system.

In our approach, the visual servoing system is used in two manners. When Manus far away from an object, only generic image features are used (e.g. size of object, position of center of gravity, etc). The visual servo is responsible for driving the gripper close to the object to manipulate. When the manipulator is ‘close’ to the object, either the end-user takes over the control of the visual-servoed degrees of freedom, or a second visual servoing algorithm is activated, for fine-positioning the gripper. For this, a-priori information can be used to identify the object and the position of the gripper with respect to the object. Only the first manner of use is described in this paper.

Control Architecture

The control system must support the combination autonomous control (via visual servoing), and direct user control, which may be defined in different co-ordinate systems. Figure 4 shows the scheme used.

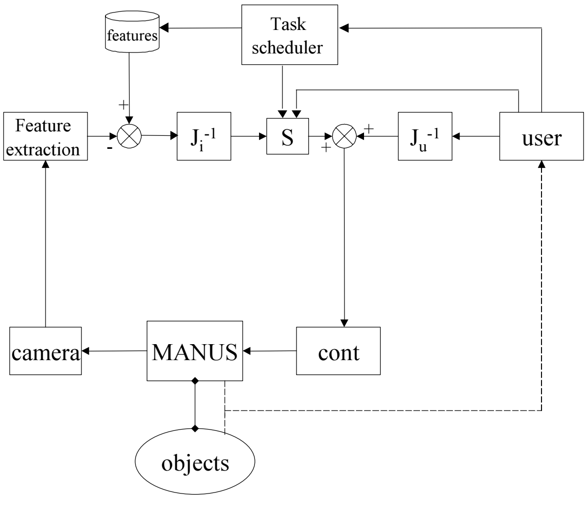

Figure 4 Visual control architecture.

Whenever a user wants to execute a certain task, a task scheduler selects a set of image features from a database. Different features may be selected during different stages of the task. The features are compared with the measured features, resulting in a feature error. The inverse image Jacobian J_i^{-1} translates feature errors to Cartesian errors. These errors are multiplied by a selection matrix S, which is a diagonal matrix of ones and zeros. A one in element S_{ii} implies that the i^{th} Cartesian DOF is actively used for visual servoing.

In parallel, the end-user may drive the robot in his/her preferred co-ordinate system. The specified user-velocity is transformed to the robot co-ordinate system, and added to the output of the visual controller.

This architecture has the advantage that tasks need not be carried out completely by the visual controller. Instead, it is possible to merge user inputs with visual servo inputs, allowing some DOFs to be controlled by the user, and some DOFs to be controlled by the visual controller. Note that when the entire selection matrix S is set to 0, the traditional MANUS controller is realized.

Vision system

In the first stage of task execution, we wish to rely on generic features, applicable for a wide range of objects. These features should supply servoing information, i.e. information that can be used to control the selected DOFs of the manipulator. An example of a generic feature is the center of mass of an object, the size of objects, or the angle of the 'long axis' of an object. Since the system operates in unstructured environments, with changing illumination, the quality of the extracted features will strongly vary. In order to cope with this, confidence measures are required, which measure the reliability of the calculated features. During control the features with the highest confidence measures can be selected. Note that the number of features must be at least equal to the number of actively controlled DOFs. If no alternative features can be selected, the visual servo must stop, and the end-user will be informed (e.g. through an audio signal).

The vision system is equipped with a low cost color camera and wide-angle lenses were used, so that the calculation of features would not fail at close proximity. Occlusion is dealt with by using the confidence measures. For segmentation of an object normalized color information is used, providing brightness independent color segmentation. From the segmented image, the center of mass, the size and the angle of the long axis are calculated. More possibilities exist to use the features for visual servoing. For example, the center of mass can be used for controlling the X an Y position of the gripper, or for controlling the yaw and pitch orientation angles. Naturally, this requires a proper selection matrix S, and image Jacobian J_i.

Experiments

Several experiments have been done using different selection matrices (and consequently different Jacobians). The task was to pick up a colored beaker.
The experiments served two goals:

1. To investigate the stability of the proposed visual control architecture.
2. To investigate the usability issues of the combined visual / user control system.

The results of one typical experiment are shown in figures 5 and 6. In this experiment, the user controlled the Z-direction (i.e. the distance between the gripper and the beaker) manually, whilst the visual servo controlled the X, Y and roll DOFs. Yaw and pitch angles remained constant.

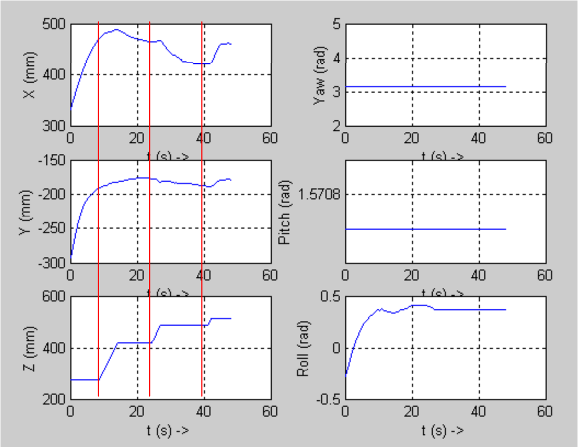

Figure 5 Results of experiments

Figure 5 shows the pose of the MANUS during task execution, calculated from the motor angles. As explained earlier, the MANUS has a significant amount of backlash. This means that the position of the motors differs from the actual pose of the gripper. Consequently, a moving motor does not automatically imply that the gripper is also moving.

This is also shown in figure 6, (which applies to the same experiment). Three signals are given. The first signal is the Y co-ordinate of the gripper, calculated from motor co-ordinates. The second signal shows the Y co-ordinate measured in the camera image. The third signals shows the input of the end-user. It can be seen that though the motor is moving, the real gripper position is relatively constant.

Other experiments confirm this outcome, and show that the proposed architecture resulted in a stable control performance. The final paper will contain a thorough discussion about these experiments.

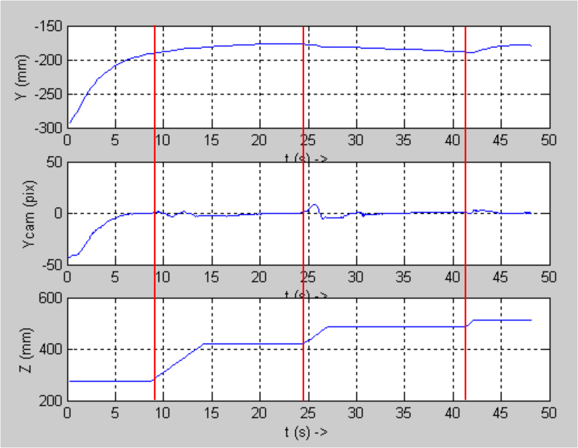

Figure 6 Motor position vs. gripper position

Conclusions and Future Work

We proposed a visual control architecture that allows semi-autonomous visual control of the Manus manipulator.

Using the control architecture it is possible to select DOFs that will be controlled by visual servoing, whilst other DOFs are directly controlled by the end-user.

The first experiments show that the total controller is stable. Differences between the motion of the motors and the motion of the gripper can be explained by the amount of backlash that exists in the manipulator.

More experiments will be done to investigate the best value of the selection matrix S for different tasks. This will be ascertained by user trials. Subsequently, the feature tracking algorithm should be improved. Furthermore, the second stage of the visual servoing algorithm (close to the object) should be developed. In this stage a flexible way of using a-priori information will be introduced.

References

REFERENCES

[1] "A tutorial on Visual Servo Control", Hutchinson S., Hager G., Corke P., IEEE transactions on Robotic and Automation, Vol. 12, No. 5, October 1996

[2] "A new approach to visual servoing in robotics.", Espiau B., Chaumette F., and Rives P., IEEE Transactions on Robotics and Automation, 8(3):313—326, June 1992.

[3] "A Modular System for Robust Hand-Eye Coordination Using Feedback from Stereo Vision.", Hager G., IEEE Transactions on Robotics and Automation, 13(4) pp. 582-595, 1997.

[4] "Dynamic sensor-based control of robots with visual feed- back", Weiss L.E., Sanderson A.C., Neuman C.P., IEEE Journal on Robotics and Automation, RA-3, 404-417, 1987.

[5] "Visual control of robot manipulators – A review", Corke P., K. Hashimoto(ed.), World scientific; Singapore, 1993

[6] "Motion Based Tracking for Dynamic, Unstructured Environments", Woodfill J., Zabih R., Computer Science Department, Stanford University, 1997.

[7] "Interactive Controlled Robotic System FRIEND to Assist Disabled People", Martens C., ICORR 2001 proceedings, p. 148, 2001