Vision-based Behavioural Modules for Robotic Assembly Systems

Prabhas Chongstitvatana

Department of Computer Engineering
Chulalongkorn University, Bangkok 10300, Thailand
email fengpjs@chulkn.ac.th

Abstract

This work describes how to program robots to work reliably in the presence of uncertainty. Some architectural principles are proposed which address the problem of decomposing robotic assembly tasks into modular units. These modular units are called behavioural modules. The problem of uncertainty is dealt with by encapsulating sensing and variation-reducing strategies inside these modules. This architecture also provide a framework to integrate sensors into a robotic assembly system. Experiments are performed with a working robotic assembly system using vision-based behavioural modules to demonstrate the validity of this approach.

1: Introduction

Reliable assembly with a robot must meet the challenge of dealing with the problem of uncertainty in the real world. Today's assembly robot systems cannot cope with uncertainty nor handle sensing adequately and are in general hard to program. Inevitably, there are variations in the shape and position of the parts. Programming a sequence of robot motions to achieve a task has to take into account these uncertainties. One way to reduce the uncertainty is to use sensors. To get good results the environment and sensors need to be set up carefully and precisely calibrated. The current generation of industrial robots are primarily position controlled devices and therefore sensing systems such as force sensing or visual sensing are still difficult to be integrated into a robotic assembly system.

2:Classical approach to robot programming

Past researches have focus on programming a robot at a high level of description so called *task-level* such as *put peg1 in hole3, move top-plate4 to mate with subassembly9*. From this task level specification the system analyses and generates the sequence of robot motions automatically. To achieve task-level programming, the past researches have focused on the study of supporting functions. These functions are, for example, grasp selection, collision free trajectory planning, motion planning, error detection and recovery [16]. The emphasis is on the modelling of geometrical characteristics of objects and the planning system.

The classical approach to robot programming systems as typified by RAPT [19], TWAIN [16], HANDEY [17] and SPAR [13], can be characterised as based on the geometry of the objects, relying extensively on the exact knowledge of this geometry to generate robot motions. Because robot motions are based on planned motions, the uncertainty in the real world is accommodated by sensing and updating the world model. The problem then lies in the quality of the sensing and how accurately the world model reflects the real world.

The method in the classical approach assumes the world to be perfect, then the uncertainty in the geometry of the parts and their locations are added in and are analysed to determine whether sensing is necessary to reduce uncertainty to an acceptable level. This makes it necessary to do the uncertainty analysis after a plan or partial plan has been created. The results of this analysis are then used to correct or modify the initial plan [2][13]. This cycle of plan, analysis, and modification is repeated until the analysis uncovers no further problems. This makes the production of each robot program a large task since the problem of uncertainty plagues all aspects of a program. Also it is difficult to predict all the necessary facts from the robot internal theory and represention to plan for the corrective motions.

1063-6730/94 $4.00 © 1994 IEEE

3: Behavioural modules

The problem of the classical approach stems from its decomposition of an assembly system. The classical approach decomposes an assembly system into functional modules which are controlled by a central system, with perceptual modules as inputs and action modules as outputs. This work argued that this decomposition is not correct. Instead, it should be broken into many task-achieving units, where the task is some useful accomplishment in the assembly world in question - such as acquiring a part. This is similar to Brooks' argument [3][4] in his mobile robot research in which he proposed the *subsumption* architecture.

Each task-achieving unit can be individually connected sensing to action to achieve a sufficiently fast reaction time to cope with the change in the real world. Each unit pursues its specific goal but co-operate with other units to achieve the desired goal. The units may work in parallel and interact with each other. This enable a system to be more robust by using more than one unit for a given situation. Rather than rely on a centralised world model, the individual unit concentrates on those aspects of the world that are directly relevant to it, i.e. it uses a minimal distributed world model rather than a centralised world model. These task-achieving units are called *behavioural modules*.

The different between this approach and the classical approach is that our planner relies on the execution system to cope with uncertainty. As a consequence it is able to plan in an ideal world, unlike the classical systems which plan from task-level specifications and synthesise sensing and motion strategies down to the robot motion level. From this ideal world plan the execution system carried out the task in the real world using behavioural modules. Behavioural modules are designed, implemented and tested in the real world to perform their tasks reliably. They can be quaranteed to perform their tasks within a certain range of uncertainty.

Behavioural modules facilitate the use of sensors to cope with uncertainties. They abstract away some of the uncertainty management from the user of a robot. This is achieved by hiding the specifics of the use of sensors as much as possible from the planner and by avoiding *explicit* reasoning about uncertainty. The planner therefore deals with an ideal world in which the robot carries out its operations reliably. This leaves the planner to deal with the problem of reasoning about the ordering of the assembly and other high level assembly strategies rather than concerning itself with the actual details of robot motions in the assembly cell.

The decomposition of a task into behavioural modules should be based on the principles that:

1. There is no reliance on a central model of the world for combining sensing and action.
2. Sensing and action should be tightly coupled within the module.
3. It is preferable to pass control via perception of the world rather than by passing parameters.

By avoiding a centralised world model, the problem of updating the world model by sensing in the classical approach is avoided. By a tight coupling of sensing and action, a fast reactive system can be realised. By passing control via perception, the need for an intermediate representation of the state of the world is lessen, in other words the world itself can be used as its own representation. (the argument about representation in this sense can be founded in [5])

4: Vision-based behavioural modules

The type of vision sensing that is used in the systems that employ a world model requires accurate calibration to establish the mapping between the sensor's field of view and a common coordinate system [12,14,20,11,1]. This type of vision system cannot easily cope with changes in camera paramenters, such as changing the position of the camera or changing the view point, zoom, etc. There are some uncalibrated vision systems that can avoid problems normally associated with static cameras. One system [6] is a self-uncalibrated system that exploits the motion of the observer to derive some useful parameters for the purpose of accomplishing the task in hand. The method of visual servo control exists that uses image features (i.e. image areas and centroids) as feedback control signals, eliminates a complex interpretation of image features to derive world coordinates [21].

Based on this background, we design and implement vision-based behavioural modules. The goal is to use them for robotic assembly tasks such as acquiring parts or mating parts. Our principle is to try and replace, as far as is possible, calculations by sensing. This equates to preferring perception over representation. Several techniques are used that contribute to this end. Frequent vision sensing, for example, allows motions in the world to be viewed as linear approximations, and only short term predictions of motion need be made for following a moving target. The tracking algorithm adopted is similar to that used by others [15] [7]. The system is purposefully made as calibration free as possible. Relative quantities are used and self-calibration is done while carrying out assembly tasks. This approach is possible because adequate sensing allows self-calibration. The camera position need not be known. The system does not rely on explicit models of the objects nor does each assembly agent need to know about the other(s). The realisation of vision-based behavioural modules will be explained in the next section.

5: Experiments

We have done three experiments to demonstrate the concept of of behavioural modules. In the first experiment we integrated visual sensing into a robotic assembly system. The experiment is based on an existing system SOMASS (Soma Assembly System) [18], which originally did not use sensors, but whose architecture was designed to facilitate their incorporation. The task of acquiring a part is implemented using a vision-based behavioural module [8]. (fig.1)

Figure 1 : The SOMASS system with vision sensing

The SOMASS system is a complete and integrated planning and execution system which performs the assembly of Soma shapes. Given the desired final shape of an assembly, the system plans the sequence of operations that will put the component parts together. SOMASS is divided into two parts: the symbolic planner system and the execution system. The symbolic planner uses an abstract representation of the task and is not concerned with the details of the real world. It generates plans (fig.2 is a part of a plan) which are then carried out by the execution system. The execution system deals with variations and uncertainty in the location and dimensions of the parts.

```
; This is plan56 of the chair assembly using the soma4 set
  ....
; ---------- The placing of fork2 ----------
  CALL reach(b3.get)
  CALL zget(b3.get:RZ(0))
; - Straight case.
  CALL zmanip(table, RZ(-270):RY(90),0,-1,2,RZ(0),0,0,3)
  CALL zput(b3.put:RZ(0))
;
 ...
; ---------- The placing of lell ----------
  CALL reach(b1.get)
  CALL zget(b1.get:RZ(-90))
; - No regrasping required.
  CALL zput(b1.put:RZ(-90):RZ(0))
.END
```

Figure 2 : A plan generated from an automatic planner

The main behavioural modules in the system are: *reach, zget, zmanip* and *zput*. The execution of the highest level behavioural modules is sequencial most of the time, i.e. doing the task step by step according to the plan. But inside a module there are several submodules that run simultaneously. The *reach* module moves the robot hand to the part using a self-calibrating vision. The *zget* module then is used to pick up the part from that location. The *zmanip* module performs a regrasp operation to change the orientation of the part (when necessary) and the *zput* module puts down the part into the final assembly.

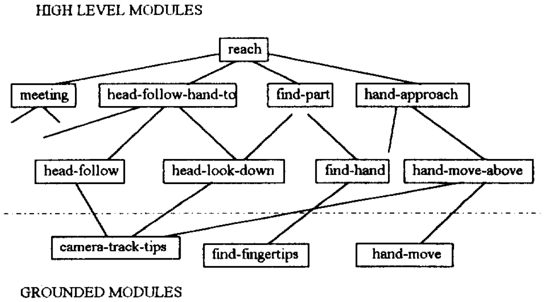

Figure 3 : Hierarchical structure of behavioural modules

The *reach* module will be examined in more details. The *reach* module is composed of several submodules (fig.3). The *meeting* module moves the hand and the mobile camera to a meeting place where the visual tracking can start. The *head-follow-hand-to* module moves the hand while keeping the mobile camera tracking it until the hand reaches a location near a part. The *findpart* module looks underneath the hand to find the part. The *hand-approach* module moves the hand until it is directly above the part using visual-servo method (see [8] for details). At the termination of *reach,* the hand is in the position to pick up the part using the *zget* module.

The communication between these modules is through the world. The *head-follow-hand-to* module uses the hand to lead the mobile camera to the part without passing any location parameters to the vision system. The camera stops at the right place where the vision system can detect the part. This enables the vision system to work without having to know the location of the part as *a priori.* Most of the time the part itself is used to communicate between modules without using any common coordinate system. For example, the *zget* module picks up the part at the point where the *reach* module left the hand. The tight coupling between the sensing and action in the *head-follow* module (sensing the hand motion and moving the camera) and in the *hand-approach* module (sensing the hand distance from the part and moving the hand) eliminates the need for a precise calibration between the camera and the robot hand.

The second experiment [10] extended the result from the first one. The limitation of the single camera system in the previous experiment is that the sensing gives only two-dimensional data, therefore in order to resolve depth ambiguity along the line of sight some *a priori* knowledge must be available. This experiment introduces the second camera and does not require that knowledge (fig.4). This two camera system in self-calibrated (the camera positions need not to be known). We developed an approximation of Jacobian that relates the motion of the hand in the images to the commanded motion. A simple task of stacking blocks is demonstrated. The result showed that the task was performed reliably and the accuracy of the system did not depend on knowing the kinematics of the robot.

The third experiment [9] added mobility to the two camera system. This enables it to change its view. This increases the range of assembly tasks that can be performed using visual sensing. Additional behavioural modules are designed to move the camera head around the object of interest without calibration between the position of the camera head and the object. A camera keeps its fixation point on a target in the image while moving the camera head. This experiment demonstrates the use of a mobile camera head to find good views that are suitable for assembly operations. Visual feedback is also used to perform a part-mating operation which requires a close range visual sensing (fig.5). The result showed that moving cameras to change view points can be achieved without calibration, thus preserving the *no centralised world model* principle.

6: Conclusion

How to program robots to work reliably in the presence of uncertainty?

This work suggests the use of a competent execution system to deal with the inevitable uncertainty of the real world. This has a profound effect on the architecture of robotic systems. The sensing operations are embedded in the run-time system, which allows the programming (or task planning) to be carried out in an ideal world. This work proposes the criteria for decomposing a task into modular units which are called behavioural modules. Behavioural modules are task-achieving units. Programming a robot in terms of behavioural modules leads naturally to task-level programs. Experiments show that behavioural modules can encapsulate the essential information about the world and communicate without relying on a centralised world model and without a global coordinate system. The tight coupling of the sensing and action inside individual modules is an important idea in coping with the uncertainty of the real world. Sensors and other manipulation strategy to reduce uncertainty can be used without putting the burden on to the user.

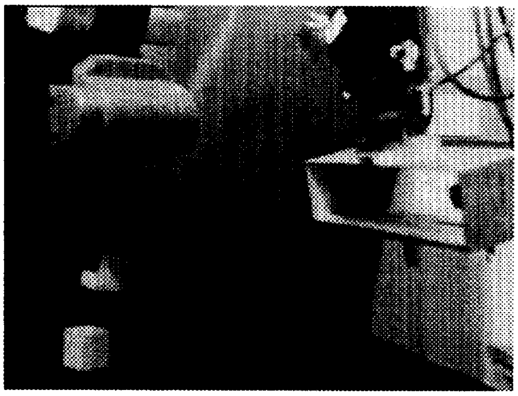

Figure 4: Two camera system on a mobile head

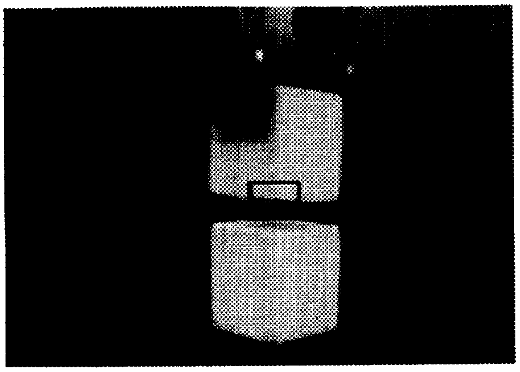

Figure 5: Part-mating operation using visual feedback

References

1. R.L. Andersson. *A robot ping-pong player.* MIT Press, 1988.
2. R.A. Brooks. Symbolic error analysis and robot planning. *Int. Jour. of Robotic Research, 1*(4):29-68, 1982.
3. R.A. Brooks. A robust layered control system for a mobile robot. *IEEE Jour. of Robotics and Automation*, RA-2(1) :14-23, 1986.
4. R.A. Brooks. Achieving artificial intelligence through building robots. MIT AI Lab., AI memo 899, 1986.
5. R.A. Brooks. Intelligence without representation. *Jour. of Artificial Intelligence*, 47:139-159, 1991.
6. R.A. Brooks, A.M. Flynn, and T. Marill. Self calibration of motion and stereo vision for mobile robot navigation. MIT AI Lab., AI memo 984, 1987.
7. C.M. Brown (Ed.). Rochester robot. University of Rochester, Computer Science, technical report 257, 1988.
8. P. Chongstitvatana and A. Conkie. Behaviour-based assembly experiments using vision sensing. In A. Colin, and E. Emil, (Eds.), *Advances in Machine Vision.* World Scientific Press, Singapore, pp.329-342, 1992, also Department of Artificial Intelligence, Edinburgh University research paper (DAI RP) 466.
9. P. Chongstitvatana and A. Conkie. Active mobile stereo vision for robotic assembly. *Proc. of 23rd Int. Sym. on Industrial Robots,* Barcelona, Spain, pp. 393-397, 1992, also DAI RP 590.
10. A. Conkie and P. Chongstitvatana. An uncalibrated stereo visual servo system. *Proc. of the British Machine Vision Conference,* Oxford, pp. 277-280, 1990, also DAI RP 475.
11. T. Heikkila, T. Matsushita, and T. Sato. Planning of visual feedback with robot-sensor co-operation. *Proc. of 1988 IEEE Int. Workshop on Intelligent Robots and Systems*, Tokyo, 1988.
12. B.K. Horn and K. Ikeuchi. Picking parts out of a bin. MIT AI Lab., AI memo 746, 1983.
13. S.A. Hutchinson and A.C. Kak. Spar: A planner that satisfies operational and geometric goals in uncertain environments. *AI magazine*, 11(1):31-36, 1990.
14. H. Inoue and M. Inaba. Hand-eye coordination in rope handling. In Brady M., and R. Paul (Eds.), *Robotics Research 1.* MIT Press, pp.163-174, 1984.
15. V. Jones. Tracking: an approach to dynamic vision and hand-eye coordination, Ph.D. thesis, university of Illinois, Urbana Champaign, 1974.
16. T. Lozano-Perez, and R.A. Brooks. An approach to automatic robot programming. MIT AI Lab., AI memo 842, 1985.
17. T.Lozano-Perez, J.L. Jones, E. Mazer, P.A. O'Donnell, W.E.L. Grimson, P. Tournassoud, and A. Lanusse. Handey: a robot system that recognises, plans, and manipulates. *Proc. of IEEE Int. Conf. on Robotics and Automation*, Vol.2, pp.843-849., 1987.
18. C.A. Malcolm. Planning and performing the robot assembly of SOMA cube construction. M.Sc. thesis, Department of Artificial Intelligence, University of Edinburgh, 1987.
19. R.J. Popplestone, A.P. Ambler, and I. Bellos. An interpreter for a language for describing assemblies. *Artificial Intelligence*, 14(1):79-107, 1980.
20. S. Sakane, T. Sato, and M. Kakikura. Model-based planning of visual sensors using a hand-eye action simulation system: HEAVEN, *Proc. of Int. Conf. Advance Robotics*, pp.163-174, 1987.
21. L.E. Weiss. Dynamic visual servo control of robots: an adaptive, image based approach. Ph.D. thesis, Carnegie Mellon University, 1984.