Object Selection using a Spatial Language for Flexible Assembly (Extended Abstract)*

Batu Akan, Baran Çürüklü, Giacomo Spampinato, Lars Asplund
Mälardalens högskola, Box 883, 721 23 Västeras, Sweden
{batu.akan, baran.curuklu, giacomo.spampinato, lars.asplund}@mdh.se

Abstract

In this paper we present a new simplified natural language that makes use of spatial relations between the objects in scene to navigate an industrial robot for simple pick and place applications. Developing easy to use, intuitive interfaces is crucial to introduce robotic automation to many small medium sized enterprises (SMEs). Due to their continuously changing product lines, reprogramming costs are far more higher than installation costs. In order to hide the complexities of robot programming we propose a natural language where the use can control and jog the robot based on reference objects in the scene. We used Gaussian kernels to represent spatial regions, such as left *or* above.

1 Introduction

In order to make industrial robots more favorable in the SME sector, the issues of flexibility has to be resolved. Typically for those SMEs, that have low volume production and frequently changing applications, it is quite expensive to afford a professional programmer or technician, therefore a human robot interaction solution is strongly demanded in order to let the user to program these systems in an intuitive way. Using a high-level natural language, which hides the low-level programming from the user, will enable a task expert who has knowledge in manufacturing process to easily program the robot and let the robot to switch between previously learned tasks. Thus, the goal is to eventually bring robot programming to a stage where it is as easy as teaching the task to a new member of the work team.

One of the problems associated with this idea is that, how the structure of the language can be kept as simple as in daily spoken language, while maintaining the accuracy and precision required for industrial applications. Our daily spoken language is extremely powerful and we make use of spatial terms in order to describe the locations of the objects. The purpose of this paper is to introduce spatial relations in to the world of industrial robot programming, with the hope of making the programming phase more easy and intuitive.

There is a large and diverse literature on spatial representation in humans and other species. Landau and Jackendoff [4] gives a detailed overview of the spatial terms and cognition. They divide spatial processing into object recognition, and specifying paths and locations. In this paper we focus on object recognition and specifying locations for selection of object in cluttered scenes.

2 Architecture

The proposed system has four main parts; Automatic speech recognition, visual simulation environment, reasoning system and spatial commands. In this project we used Microsoft Speech API 5.1 (SAPI 5.1) [3], both to recognize the commands given to the robot and to synthesize speech in order to give feedback to the user. The commands are parsed and passed to the reasoning system which eventually drives the robot to perform the necessary actions and commands requested by the user.

A simulation environment which can simulate a general 6 Degree of Freedom (6DoF) industrial robot has been created in an OpenGL environment in order to simulate the behavior of the robot in a noise free environment where objects' positions and orientations are known. Besides providing a quick test environment, it also acts as a visual feedback to the user. On a higher level of abstraction, commands issued by the user, such as picking up an object is realized through a set of low-level movement functions. Given the position, the orientation of an object and the gripping pattern to be used in order to pickup that object, the Higher Level Movement Commands generates the necessary sequence of low-level movement functions for approaching, grasping and retracting.

In our daily spoken language prepositional terms are very important for us to understand each other, yet they often have a vague meaning. The meaning of spatial terms such as; *left* or *behind* cannot be divided into discrete regions, it is difficult to determine where the notion of *left* starts and where it ends. In order to overcome this problem, we represent the spatial regions using Gaussian kernels, hence turning the problem into a multi-class classification problem. The object is assigned to a region rep-

*This work described in this paper is supported by *Robotdalen* and *Sparbanksstiftelsen*.

Spoken command	*Purpose*
pick up	Picks up a specified object
put selection/it	Puts the object
apply selection	Ends selection phase
it is/they are	Declares new properties for the selection
execute	Begins executing the path
cancel	Clears out the execution queue

Table 1. List of commands to control the robot.

resented by the respective Gaussian kernel based on its Mahalanobis distance to the kernel [2].

Reasoning system functions as the system brain. While interpreting the speech commands given by the user it plans and calls necessary motor functions to realize the commands. As objects are inserted into the simulation environment, or recognized by vision system which is currently under development, information about the objects are asserted into the knowledge-base. Only structural relations about the objects are kept in the knowledge-base, where as, position and orientation are not stored. Spatial relations are computed by the visual simulation environment on demand. The most important predicate in the knowledge-base is the *property* predicate. It is used to define structural information of the objects. It can also be used to define the type of an object; such as defining whether the object is a workbench or a manufacturing object or of any other object type [1]. An example definition for a red, cylindrical, manufacturing object is as follows:

```
object(object12). % object identifier
property(object12, color, red).
property(object12, shape, cylindrical).
property(object12, type, manufacturing).
```

In the proposed language there are commands for object manipulation, object selection, and managing the job queue. Table 1 shows the list of available commands in command set. Manipulation and declaration phrases are followed by a noun phrase describing the structural properties of the object or they are followed by prepositional phrase, giving a reference object and its spatial relation with the *figure* object.

The reasoning system receives text input from automatic speech recognition engine and converts it into Prolog predicates using the set of grammar rules defined by the Natural Language Processor (NLP). These generated predicates capture information about the structural properties of the object such as color, shape or any other property that is defined for that object type as well as cues revealing relative spatial positions of the objects in the scene. A command like: "Pick up a red object which is behind a cubic object.", would translate to Prolog predicates as:

```
object(X), property(X, color, red),
object(Y), property(Y,shape, cubic),
spatialpos(X,Y,behind), pickup(X).
```

3 Experimental Results

In order to demonstrate the usefulness of the proposed system, we designed test cases that utilizes the industrial robot for simple palletizing/depalletizing tasks. The simulated testing environment consists of one ABB IRB-140 robot, one workbench and two conveyor bands on each side of the robot. Figure 1 presents a view of the robot and the working environment around it. In Figure 1 the robot is asked to pick up all blue objects and put them on the conveyor to the left of the robot.

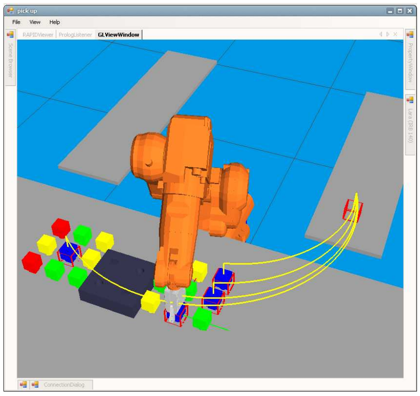

Figure 1. Screenshot of the simulator where the robot is asked to pick and place all the blue objects over the conveyor band.

4 Conclusion

In this paper, we demonstrated a high-level restricted language in order to command an industrial robot for simple pick and place applications. The proposed language can handle attributes of the objects in the environment, such as shapes, colors, and other features in a natural way, it is also equipped with functions for handling spatial information enabling the user to be able to relate objects spatially to static objects, robots, table, etc. as well as to other work objects.

References

[1] B. Akan, B. Curuklu, and L. Asplund. Interacting with industrial robots through a multi-modal language and sensor systems. Seoul, Korea, October 2008. International Symposium on Robotics.

[2] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley Interscience Publication, 2nd edition, 2001.

[3] http://www.microsoft.com/speech. *Microsoft Speech Technologies*.

[4] B. Landau and R. Jackendoff. What and where in spatial language and spatial cognition. *Behavioral and Brain Sciences*, 16:217–265, 1993.