Automation using Robot Programming by Demonstration Technique

Dr. Shantanu K. Dixit[1] Tushar K. Patil[2]

[1]Department of Electronics & Tele-Communication Engineering [2]Department of Electronics Engineering

[1,2]Walchand Institute of Technology Solapur, Maharashtra, India-413 006

Abstract— Presently Programming the robot is a becoming daunting task. Inaccurate, inconsistent and unexpected results may be occurred because of Direct Programming of the robots & also it may take lots of time for programming. Also due to the costs involved in the development and maintenance of robot programming technique, automatic programming techniques and Robot Programming by (Human) Demonstration (RPD) techniques are currently attracting a lot of interest. Paper aims to build a robotic arm, which is provided with the sensors to sense and record parameters and the motors to repeat an action that is demonstrated by the human, so that accurate and efficient program is being generated. RPD is an intuitive method to program a robot. A human demonstrator with his/her hand, teaches an action to the robot, by physically interacting with it to perform an action. This allows the measurement and recording of the human's positional and other parameters that are relevant to perform the demonstrated action.

Key words: Robot Programming by Demonstration (RPD), learning from Demonstration (LfD), Industry Automation, Robot Teaching and Learning

I. INTRODUCTION

Robots are becoming more and more intelligent as technology advances in the areas of CPU speed, sensors, memories etc. The Robot Programming by Demonstration (RPD) is becoming popular in the Automobile and many industries due to the costs involved in the development and maintenance of robot programs. Here, the User has an implicit knowledge and information about the action or task to be performed but doesn't have usually the Programming Skills to reconfigure or program the robot. Thus, Demonstrating an action or task through the examples allows us to not to explicit Program or to do the coding of the Robots. The RPD strategies may include the teach-in, guiding or play back methods in which the new task or action teaching to the Robot arm is done and the same task or action is then performed by the Robot arm[1]. But currently, the Task or action is demonstrated to the robot multiple times, then the process of Generalization is applied to the data sets that are retrieved and then the Task or action is performed. The development of Robots in industries naturally brought a growing interest in robot programming by demonstration. As a robot is supposed by its nature to adapt to new environments, the algorithms used for its control should be flexible.

The absence of this Problem domain knowledge requirement leads to the policy development to be done by the Non-technical persons also, satisfying a need that increases as robots become more commonplace. Furthermore, Demonstration has an attractive feature of being a good medium for communication from humans those who are using Demonstration to teach actions to other humans. With this RPD, training datasets are consisting of the demonstration of the sequence of the actions. There are 2 ways for the Learning from Demonstration. First one is learning by observation. In this, the Robot is observing the demonstration continuously and generates the data. For this Robot may take help of the external sensors like cameras. The second one is learning by experience in which the Robot is performing actions along with the Demonstrations. But, the most important thing is that Learning from experienced demonstrations. Here, the demonstrator is demonstrating actions and from which Robot is developing the Policies that can be used for final robot task execution. As the sensors are located on the robot system, with the help of it robot is learning the action or the task.

In this paper, a robot arm is built on which the motors and the sensors are present. Sensors do the task of sensing and the motors will perform the action that is demonstrated by the demonstrator. Detailed block diagram is explained later.

A. *Definition and Terminology:*

RPD is a way to teach the robot new states and actions by demonstrating the sequence of state-action pairs to the Robot arm to avoid direct programming of the Robot arm. The demonstration based Learning methods are well known by the various other terms such as Programming by Demonstration (PbD), Learning from Demonstration (LfD), imitation & mimicry, learning by Showing etc. Though some of Definitions of Terms can be understood, their use differs in various articles.

The RPD concepts can be implemented for industrial robots not only to perform only a certain set of actions but also to perform various other set of actions just by simply demonstrating the state-action pairs to the Robot arm. The task of Teaching to the robot can be done by the novice or the Non-technical person. This concept is not limited to the autonomous robots but can be used for the humanoid Robots.

In order to do the Programming by Demonstration, there are 2 problems that must be solved[2],

(1) What to imitate - here, robot has to extract the required features from the 1 or multiple Demonstrations,
(2) How to imitate - it consists of the problem solution with which the Demonstrated task is reproduced by the Robot arm.

B. *Basic Flow:*

The flow of the work can be generalized and is shown in figure 1. As it is shown that, Very first the Task or sequence of actions are demonstrated to the Robot arm[5]. So during Demonstration, The User or Teacher may directly or indirectly present the task or sequence of state-action pairs to the Robot arm. Here, direct or indirect is nothing but, The Demonstration is given by the Teacher on actual robotic System or by using other similar kind of systems in which the training data should be Mapped from the Demonstrated

All rights reserved by www.ijsrd.com

system space to the final robotic System Space. One more thing that must be cleared at this point is that, User may go for a single or multiple Demonstrations which may affects the type of final Policy Derivation Technique used for the Final execution of the Demonstration by the actual Robot System. While demonstrating the task, the sensors that are located into the system, Responds according to the Demonstration. So, training data is then recorded directly or recorded after the Generalization of the data set. Once the Training data sets are received from the Demonstration(s), the policy is then derived from the data sets with which the Task is finally performed by the Robot arm.

The figure 1 is shown just only for a single Demonstration. If the Multiple Demonstrations are used, then the change will be the feedback. For this case, we may get accurate results. Because, a single demonstration doesn't give us accurate results. In multiple Demonstrations, we may take the help of the history and the current datasets so that we can update the policy as soon as the new Training data sets are available. So, as the number of Training data sets is increased, we get consistent and accurate results and the intended action can be performed very well. The demonstrator is demonstrating the task with his/her own hands to the robot arm. During this, Sensors from the Robot arm will sense the required positional, orientation parameters and is then replicated by the robot after the Generalization.

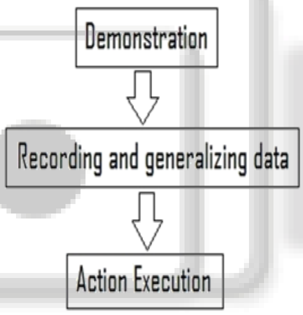

Fig. 1: Basic flow

Generalization is required and is generally applied to multiple Demonstrations. For Single Demonstration there will not be use prior knowledge. But for multiple demonstrations, history is used. In generalization, it is deciding which states must occur after the other ones and also to remove the unwanted state-action pairs from the final execution sequence.

II. DERIVATION OF STATES

If the task includes dealing with the multiple objects of different shapes in front of the robot, then the objects location information is considered as either absolute or relative positions with the other objects already placed. No doubt, some of the objects are placed absolute, but we may go for relative location information. Here, not only the object is having relative or absolute information about object but also it can have relative or absolute orientation information with reference to the other objects. [3] Paper includes the information about the state derivation. In that, they have used k-means clustering algorithm to generate the states of the objects with their relative or absolute information.

III. TASK GENERALIZATION

Once a single or multiple demonstrations are given by the user to the robot arm, now the task of generalization should be applied on all of the demonstration sequences. As said in the previous discussion that, generalization is required to apply if and only if multiple demonstrations are Present. Single demonstrations don't give us guaranty of accurate and precise results. But, this also doesn't imply that very large number of demonstrations should be applied on the robot arm. Enough finite number of demonstrations are sufficient. It's not necessary to have very large number of Demonstrations. This generalization process can be applied to any automated robot systems such as home automated robots etc.

The general process for the generalization can be given as follows,

(1) The teacher will give multiple demonstrations to the robot arm which are considered as a sequence of actions,
(2) By referring to the multiple sequences, the constraints are derived.
(3) At last, with the help of derived constraints and all of the demonstrated sequences, the final execution sequence is derived.

The above steps can be made clear by referring to the diagram. According to the steps given, the diagram bellow is shown. Here, the each variable represents an action or a task. For the generalization process to begin, the multiple demonstrations are required to be given to the robot arm. The diagram is shown just as an example. It easily clears from the diagram that, very first the Robot should generate the sequences of Demonstrations. Here, there are 3 demonstrations given to the robot. So, there will be 3 sequences generated.

There are 2 unwanted variables included in the last 2 sequences i.e. M & W. Those should be eliminated from the final execution sequence. The second step is to derive the constraints. The constraints are derived by considering all the demonstrations. As every sequence is starting with a variable A and it is less than other variables, the constraints like A<F, A<G … and so on are written. Likewise, all the possible constraints are written. Once done with the second step, by using demonstration sequences and the constraints, the Final execution sequence is generated. This is what the sequence that is followed by the robot arm.

There are 2 unwanted variables included in the last 2 sequences i.e. M & W. Those should be eliminated from the final execution sequence. The second step is to derive the constraints. The constraints are derived by considering all the demonstrations. As every sequence is starting with a variable A and it is less than other variables, the constraints like A<F, A<G … and so on are written. Likewise, all the possible constraints are written. Once done with the second step, by using demonstration sequences and the constraints, the Final execution sequence is generated. This is what the sequence that is followed by the robot arm.

All rights reserved by www.ijsrd.com

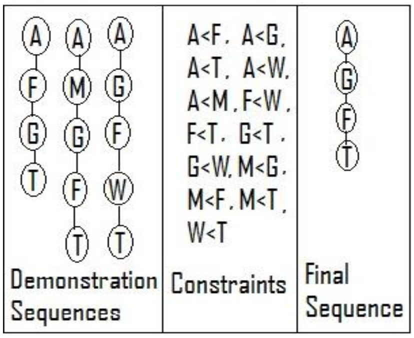

Fig. 2: Generalization Process

The above process is ignoring all of the possible conflicts that are coming into the Generalization process. From above diagram, the G is coming after the F in the first demonstration and in the second & third demonstration; G is coming before the F. So, it is solving such type of conflicts also. If the constraints that are derived by the robot are correct, then the final execution sequence will be as it is intended by the Demonstrator or the Teacher. And if the demonstrations are increased, then we can update the constraints. In turn, it will also update the final execution sequence. Once, the execution sequence is derived, the robot is free to execute the final sequence as many times as we want. For performing the Generalization of the data, we may take the help of the Neural Network. With the Neural network, we can implement various algorithms and it is not shown in this paper.

The advantage of generalization is thus that,

(1) It removes the unwanted action variables from the final sequence, and
(2) To know the action variables that are common to each of the Demonstrations etc.

We have seen one of the easiest method for the generalization of the demonstrated sequences. There is other method for the generalization process i.e. longest common Subsequence (LCS)[4]. In the LCS method, the longest common sequence from the given demonstrations is taken out. No doubt, the Common actions will be there in the final execution sequence but also the other actions are appearing as alternate ways into the final execution sequence. But, this type of method is not good for the Manipulation tasks. So, it is rarely used method [4].

IV. Block Diagram

The below diagram consists of micro-controller that is interfaced with the Robot arm. Robot arm includes the various types of sensors like rotation angle sensors, velocity sensors etc. and the Motors like servo motor. We can have the accelerometer sensor to detect the tilt angle of rotation of the motors.

The demonstrator with his / her own hands will teach a particular simple action to the Robot arm. While teaching action, the motors of a robot arm will gets rotated by some angle. These angles are then measured with the help of Rotation angle sensors. Angles of all the motors are sensed continuously, recorded into the Memory till the teacher teaches the action. Generalization is applied on the data. After getting all the required values recorded, the program is created for that action by the controller and is repeated by the Robot arm.

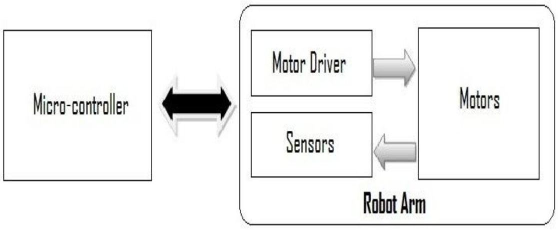

Fig. 3: System Block Diagram

Here, we can have the velocity sensors and the other type of sensors located on the body of the Robot arm. The camera can be used for the imitation purpose. There is a method of using 2 cameras to monitor the demonstration process which is based on Stereo vision[6]. In that, the output from the two camera sensors are merged to form the final image, then the final image is used to track the object and the Demonstration is performed[7]. The thing that must be remembered is to check whether the micro-controller is having that many ports to interface all the type of Sensors and the motors to the Robot arm.

A. *Planner:*

With the Task generalization process, the final execution sequence is generated. Now, here it comes the turn of the Path planner. This is what a process of finding the actual path to be followed by the robot arm. The path planner must be carefully designed so as to perform the demonstrated task.

```
arm Down 0d 30d 0d
arm none 0d 0d 30d
arm Up 0d 30d 0d
arm Left 60d 0d 0d
arm Down 0d 30d 0d
arm none 0d 0d -30d
arm Up 0d -30d 0d
```

Fig. 4: Final sequence in terms of Macros

In reality, we know how many actions that the robot can perform. For example, with the simple arm having 2 or 3 motors on it, the robot can move up, down, Rotate Left, Rotate Right, front and back. So, as soon as the controller comes to know that some action such as moving right is sensed with the help of the Sensors with the location information; the generalized sequence is generated which is then followed by the robot arm. But, after the Generalization, once again the final execution sequence with the action and the location and position information is then generated by this path planner. This final generated real sequence is temporarily stored in memory or can be stored in a file. This sequence is then executed by the robot arm.

Path planner generates a final execution real sequence that consists of the Macro names and the positional or location. The Final sequence for example that can be generated by the planner as shown in the above diagram

V. Conclusion

As there is no any overhead of coding the robots, it allows us to apply the methods to industrial robots so that the robot can be operated by Non-technical persons. Also it can be used in Home automation Systems to perform the various actions just only by teaching the robot.

All rights reserved by *www.ijsrd.com*

Furthermore, the work can be extended that will offers the 'Plug and produce' technology of the Manufacturing systems to be free from the re-configurability.

VI. ACKNOWLEDGEMENTS

We would like to take this opportunity to thank one and all who have provided their valuable advice, without their guidance this work would not have been a success, we have to thank who have helped us directly or indirectly since they have given us more than just guidance.

Our profound thanks to Dr. S R Gengaje, Head of the Department, Department of Electronics Engineering, Walchand Institute of Technology, Solapur, for his invaluable advice and constant encouragement to complete this work in a successful manner

I would like to convey our sincere thanks to management and supportive staff of Walchand Institute of Technology, Solapur, for encouraging us to come up with this paper work.

REFERENCES

[1] Wikipedia- www.google.co.in.
[2] Brenna D. Argall, Sonia Chernova, "A survey of Robot learning from Demonstration", Robotics and Autonomous Systems, Elesevier, Robotics and Autonomous Systems 57 (2009) 469-483.
[3] Staffan Ekvall and Danica Kragic, "Robot Learning from Demonstration: A Task-level Planning Approach". ARS International, Vol. 5, No. 3 (2008).
[4] Monica N. Nicolescu, "Natural Methods for Robot Task Learning: Instructive Demonstrations, Generalization and Practice", University of southern California, 2002.
[5] S. MÜNCH, J. KREUZIGER, "Robot Programming by Demonstration (RPD) - Using Machine Learning and User Interaction Methods for the Development of Easy and Comfortable Robot Programming Systems", Proceedings of the 24th International Symposium on Industrial Robots (ISIR '94).
[6] Sylvain Calinon and Aude Billard, "A Probabilistic Programming by Demonstration Framework Handling Constraints in Joint Space and Task Space", Learning Algorithms and Systems Laboratory (LASA), Switzerland.
[7] Yusuke MAEDA, Nanako ISHIDO, "Teaching of Grasp/Graspless Manipulation for Industrial Robots by Human Demonstration", IEEE/RSJ, Switzerland, October 2002.

All rights reserved by *www.ijsrd.com*