

자유도-4 로봇 팔을 위한 ROS 기반 Pick-and-Place 동작 제어

김영주^o

^o신라대학교 컴퓨터소프트웨어공학부

e-mail: yjkim@silla.ac.kr^o

ROS-based Pick-and-Place Motion Control for a Robot Arm of 4 Degrees of Freedom

Young-Ju Kim^o

^oDivision of Computer Software Engineering, Silla University

● 요약 ●

본 논문은 ROS 프레임워크를 기반으로 4-자유도를 가진 로봇 팔의 Pick-and-Place 동작 제어를 구현하고, 틱택토 게임에 적용한 사례를 제시한다. 로봇 팔의 Pick-and-Place 동작 제어는 움직임 궤적 계획, 충돌 회피 그리고 역기구학 모델링 연산들과 이를 이용한 복잡한 제어 과정을 요구한다. ROS 프레임워크는 간단한 인터페이스 통해 로봇 팔의 동작을 용이하게 제어할 수 있도록 일련의 연산들과 제어 동작을 통합하여 MoveIt 패키지를 제공하고 있으며, 본 논문은 이 패키지를 기반으로 4-자유도의 로봇 팔에 대한 동작 제어 모듈을 구현하였다. 또한 이를 틱택토 게임에 적용하여 로봇 팔을 적절히 제어함을 확인하였다.

키워드: 로봇 팔(Robot Arm), Pick-and-Place 동작, ROS(Robot Operating System), 움직임 궤적 계획(Motion Trajectory Planning)

I. Introduction

로봇 팔의 Pick-and-Place 동작은 로봇 팔의 모든 동작을 구성하는 기본 동작으로서 동작 제어를 위한 기본 제어 기술을 요구한다. 로봇 팔의 동작 제어는 움직임 궤적 계획, 충돌 회피 그리고 역기구학 모델링에 대한 연산을 필요로 하며, 각각의 연산에 대해 다양한 알고리즘이 개발되어 사용되고 있다[1,2,3]. ROS 프레임워크는 이러한 연산들과 이를 이용한 일련의 복잡한 제어 과정을 용이하게 사용할 수 있도록 MoveIt 패키지를 지원하고 있으며, 로봇 팔의 제어에서 우수한 성능을 발휘하는 것으로 평가되고 있다[4,5].

본 논문은 ROS 프레임워크를 활용하여 4-자유도인 로봇 팔의 Pick-and-Place 동작 제어를 구현하고, 이를 테스트하기 위해 틱택토(Tic-Tac-Toe) 게임에 적용하고, 정상적으로 동작함을 확인하였다.

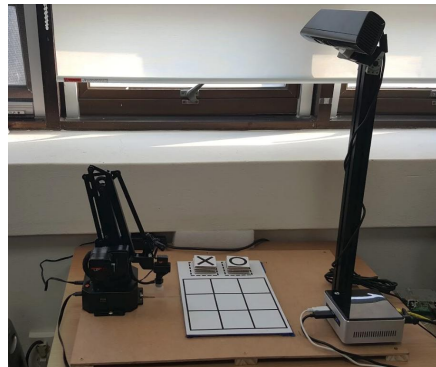


Fig. 1. Test-bed System Architecture

II. Test-bed System Architecture

실험 시스템은 Fig. 1과 같이 UFactory SwiftPro 로봇 팔, 마이크로소프트 Kinect 카메라, Intel NUC-15 PC 그리고 3x3 틱택토 게임보드로 구성된다.

III. ROS-based Pick-and-Place Motion

1. Tic-Tac-Toe Game Algorithm

본 논문의 틱택토 게임은 게임자와 컴퓨터가 게임을 수행하며, 게임자가 먼저 직접 'X' 마커를 집어 원하는 게임보드 위치로 두면 컴퓨터가 다음 'O' 마커의 위치를 결정하고 로봇 팔이 'O' 마커를 집어 지정된 위치에 가져다 두도록 진행된다.

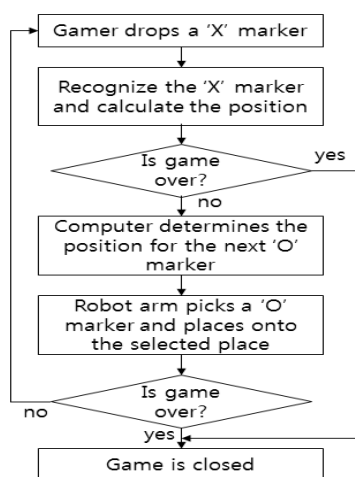


Fig. 2. Tic-Tac-Toe Game Algorithm

2. Recognition of 'X' Marker

본 논문은 마커 모양의 단순성을 감안하여 OpenCV의 특징점 매칭 연산을 이용하여 마커 인식 연산을 구현하였다.

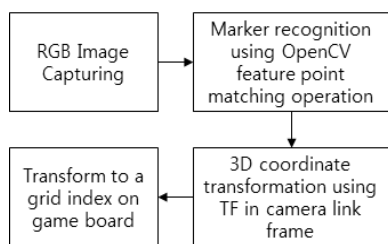


Fig. 3. 'X' Marker Recognition Algorithm

3. Determination of 'O' Marker Position

본 논문은 MinMax 알고리즘을 이용하여 틱택토 게임을 구현하였다. 게임 알고리즘에 대한 자세한 기술은 논문의 관점에서 벗어나므로 생략하도록 한다. 게임 알고리즘에 의해 게임자가 놓은 'X' 마커에 대해 가장 게임을 이길 확률이 높은 곳은 다음 'O' 마커의 위치로 결정한다.

4. Pick-and-Place Motion of 'O' Marker

'O' 마커의 구체적인 Pick-and-Place 동작은 Fig. 4와 같고, 로봇 팔의 이동 제어를 위한 구체적인 알고리즘은 Fig. 5와 같다.

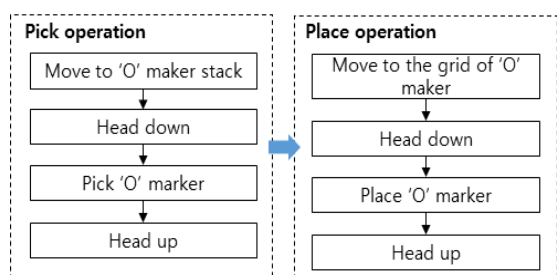


Fig. 4. Details of Pick-and-Place Motion

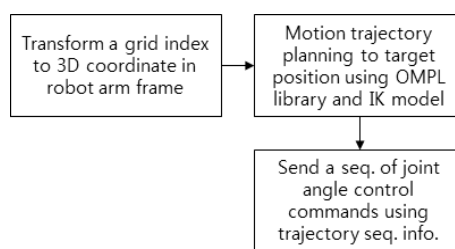


Fig. 5. Motion Trajectory Planning Algorithm

IV. Experiments & Conclusions

본 논문은 수차례의 틱택토 게임 수행을 통해 구현된 로봇 팔 동작 제어 모듈이 정상적으로 동작하고 있음 확인하였다. 결론적으로 본 논문은 ROS 프레임워크를 활용하여 4-자유도 로봇 팔의 동작 제어 모듈을 구현하고 틱택토 게임에 적용하여 정상적으로 제어가 이루어짐을 확인하였다. 향후에는 이동 중에 감지되는 장애물을 회피하는 기능을 구현할 예정이다.

REFERENCES

- [1] S. Cambon, R. Alami, and F. Gravot, "A Hybrid Approach to Intricate Motion, Manipulation and Task Planning," *Int. J. Rob. Res.*, vol. 28, no. 1, pp.104-126, Jan. 2009.
- [2] Wang, W., Suga, Y., Iwata, H., and Sugano, S., "Solve inverse kinematics through a new quadratic minimization technique," *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pp.306-313. 2012.
- [3] I. A. Sucas, Mark Moll, and Lydia E. Kavraki, "The Open Motion Planning Library," *IEEE Robotics & Automation Magazine*, Vol. 19, No. 4, pp.72-82, Dec. 2012.
- [4] M. Quigley, K. Conley, Brian P. Gerkey, J. Faust, T. Foote, J. Leibs, Rob Wheeler, and A. Y. Ng., "ROS: an open-source robot operating system," In *ICRA Workshop on Open Source Software*, 2009.
- [5] I. A. Sucas and S. Chitta, "MoveIt!," <http://moveit.ros.org>