대형 언어 모델 기반 로봇 Bin Picking 작업 계획

차성훈, 김승준, 오윤선

한양대학교

{cktjdgns98, rlatmdwnseo, yoh21}@hanyang.ac.kr

LLM-based Task Planning in Robotic Bin Picking

Seonghun Cha, Seungjun Kim, Yoonseon Oh

Hanyang University

요 약

본 논문은 상자 (bin)에 다양한 물체들이 혼재되어 있는 환경에서 사용자가 언어 명령으로 지정한 타겟 물체를 로봇이 효율적으로 집어 옮기는 bin picking 시스템을 제안한다. 제안된 시스템은 인식 모듈과 계획 모듈로 구성되며, 인식 모듈은 객체 탐지 (Object detection) 및 인스턴스 분할 (Instance segmentation) 모델을 통해 물체의 정보를 인식한다. 계획 모듈은 인식된 물체 정보를 바탕으로 장면 그래프 (Scene graph)를 구성하고, 대형 언어 모델 (Large Language Model)을 활용해 사용자의 언어 명령에 대응되는 하위 작업 계획을 생성한다. 실험 결과, 사고 사슬 (Chain-of-Thought) 프롬프팅과 함께 인컨텍스트 (In-context) 학습을 수행한 LLM이 높은 최적화 성능을 보였으며, 실제 UR5e 로봇을 사용한 전체 시스템 검증에서는 0.75의 성공률로 bin picking을 수행하였다.

Ⅰ. 서 론

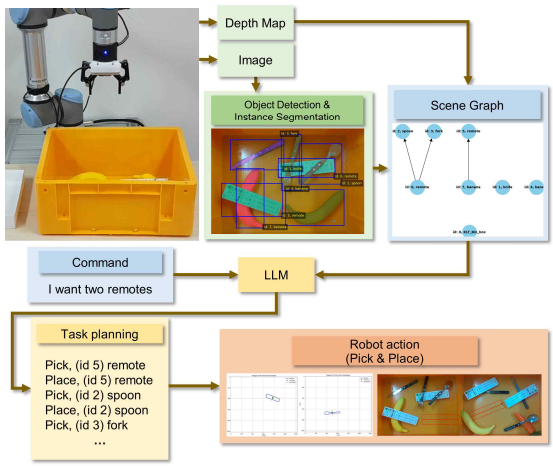
Robotic bin picking은 로봇이 상자 내부에 무작위로 쌓인 물체들을 인식하고 선택하여 특정 위치로 옮기는 작업으로, 다양한 산업 환경에서 자동화를 통한 효율성 향상을 위해 사용된다. 일반적으로 bin picking 시스템은 데이터 수집, 물체의 자세 추정, 그리고 집기 기술에 대해 연구되어왔다[1]. 대부분의 연구들은 다양한 물체들이 혼재되어 있는 상자 (bin) 내에서 개별 물체를 인식하고 안정적으로 집는 데 초점을 맞추며, 물체의 클래스나 집는 순서를 엄격하게 고려하지 않는다. 그러나 물류 산업에서는 현장 특성상 소비자의 주문에 따라 원하는 물체를 알맞은 개수로 선별하여 담아야 하기 때문에, 각 물체를 클래스별로 분류하는 작업과 타겟 물체를 집기 위한 적절한 작업 계획 생성이 필요하다. 이러한 작업 없이 기존 bin picking 시스템만으로 물류 산업에 적용하려면 사전에 각 품목을 여러 상자에 분류해야 하는 비효율적인 문제가 있다.

이러한 문제를 해결하고자, 본 연구에서는 상자에 다양한 물체들이 혼재되어 있는 환경에서 사용자가 원하는 타겟 물체에 대한 언어 명령이 주어지면, 그래프로 표현한 환경 정보를 기반으로 LLM (Large Language Model)을 통해 작업 계획을 생성하고, 계획에 따라 로봇 동작을 수행함으로써 효율적인 bin picking 시스템을 구축한다.

Ⅱ. 본론

2.1 환경 및 시스템

상자 내에는 여러 클래스의 물체들이 혼재되어 있으며, 동일 클래스의 물체도 존재한다. 또한, 사용자는 상자 내 물체의 목록을 알고 있으며, 타겟 물체가 아닌 물체들은 상자 외부로 옮길 수 없다고 가정한다. [그림 1]은 본 연구에서 구축한 bin picking 시스템을 나타내며, 크게 인식 모듈과 계획 모듈로 구분된다. 카메라의 상단 뷰에서 캡쳐한 RGB 이미지와 depth map이 시스템의 환경 정보로 입력되고, 타겟 물체와 개수에 대한 정보가 언어 명령으로 주어진다. 환경 정보를 그래프 형태로 나타내기 위해 인식



[그림 1] Bin picking 시스템

모듈에서는 각 물체에 대한 bounding box, class, mask 정보를 추론한다. 계획 모듈에서는 이러한 정보와 depth map을 함께 사용하여 물체들 간의 상하 관계를 그래프로 나타낸 후, 이를 기반으로 하위 작업 계획을 생성한다. 로봇은 생성된 계획을 바탕으로 동작을 수행하여 타겟 물체만을 다른 상자로 옮기게 된다. 이 때 로봇이 한 번의 pick and place 동작을 수행할 때마다 환경 정보를 재인식하여 변화하는 환경에서 하위 작업에 해당하는 물체의 위치를 정확히 파악하고 안정적으로 집을 수 있게 하였다.

2.2 인식 모듈과 계획 모듈

인식 모듈에서는 이미지를 이용하여 객체 탐지 (Object detection)과 인스턴스 분할 (Instance segmentation)을 수행한다. 객체 탐지 모델은 Co-DETR[2], 인스턴스 분할 모델은 SAM[3]을 사용하였다. 가장 먼저 객체 탐지를 수행하여 물체의 바운딩 박스와 클래스에 대한 정보를 얻고, 출력된 바운딩 박스와 이미지를 SAM에 입력하여 각 물체의 마스크 정보

를 추출하였다. 그리고 동일한 물체를 구분하기 위해 탐지된 물체에 고유 id 번호를 부여하였고, 재인식을 진행할 때마다 이전 바운딩 박스, 클래스, id 정보를 참조하여 고유 id를 추적하도록 하였다.

계획 모듈에서는 먼저 인식 모듈로부터 추출된 정보와 depth map을 활용하여 규칙 기반 장면 그래프 (Scene graph)를 생성하였다. 이 과정에서 다음의 세 가지 규칙을 사용하였다: 1) 두 물체의 마스크가 인접해 있다면, 이들 물체는 겹쳐진 상황으로 간주한다. 2) 두 물체가 인접했을 때, 마스크 간의 경계선을 기준으로 근처 픽셀들의 depth 평균값을 사용하여 물체 간의 상하 관계를 결정한다. 3) 특정 물체의 마스크 아래에 두 개 이상의 물체가 존재할 경우, 이들 물체가 서로 인접해 있으면 depth 값을 비교하여 바로 아래에 있는 물체만을 그래프 간선으로 연결한다. 이렇게 장면 그래프를 생성한 후, 사고 사슬 (Chain-of-Thought)[4] 프롬프팅과 함께 인컨텍스트 (In-context) 학습이 수행된 LLM에 그래프 정보를 텍스트로 입력하여 하위 작업 계획을 생성하였다. 이 때, 높은 추론 능력을 가진 GPT-4[5] 모델을 사용하였다. 또한 하위 작업의 형태는 pick, place와 같은 로봇 동작과 대상 물체가 순차적으로 나열되도록 설정하였다.

2.3 실험 결과

본 연구에서는 여러 물체들이 혼재되어 있는 환경 정보가 그래프로 주어졌을 때, 사고 사슬 프롬프팅과 함께 인컨텍스트 학습이 수행된 GPT-4 모델이 하위 작업 계획을 생성하는데 효과적인지 평가하였다. 비교 모델로는 추가 학습을 진행하지 않은 Zero-shot GPT-4와 7가지 대표적인 예시를 통해 인컨텍스트 학습을 진행한 Few-shot GPT-4로 설정하였다. 실험에 사용된 그래프는 Isaac Sim 시뮬레이터를 이용하여 상자에 평균 10개의 물체를 랜덤으로 떨어뜨려 구성하였고, 그래프의 구조와 간선의 복잡성에 따라 Easy, Medium, Hard 난이도로 나누어 각각 20개씩 수집하였다. Easy는 깊이 2 이하의 단순 선형 간선으로 구성된 그래프, Medium은 깊이 3 이상의 관계를 포함하면서 단순 선형 간선으로만 구성된 그래프, Hard는 깊이 3 이상의 관계를 포함하며 복잡한 간선을 가지는 그래프로 정의한다. [표 1]은 그래프의 난이도에 따른 세 가지 모델의 성능과 출력된 동작 시퀀스 길이의 평균을 최적 동작 시퀀스 길이의 평균과 비교한 결과를 나타낸다. Feasibility는 실행 가능한 작업 계획 생성 성공률을 나타내고, Optimization은 최적화된 작업 계획 생성 성공률을 의미한다. Zero-shot GPT-4 모델과 Few-shot GPT-4에 사고 사슬 추론을 적용한 모델을 비교한 결과, feasibility에 대해선 비슷한 성능을 보였으나, optimization에서는 Few-shot GPT-4에 사고 사슬 추론을 적용한 모델이 가장 높은 성능을 보였다. 평균 동작 시퀀스의 길이를 비교했을 때, Zero-shot GPT-4 모델은 최적 동작 시퀀스에 비해 불필요한 동작 시퀀스를 포함하는 경우가 많았으나, 사고 사슬 추론을 적용한 모델은 최적 계획을 생성하도록 유도되어 대부분의 경우 최적화된 작업 계획을 생성하였다. 인컨텍스트 학습만 진행한 모델은 그래프의 난이도가 어려워질수록 feasibility와 optimization 성능 모두 감소하였다. 대부분의 실험에서 동작 시퀀스의 길이가 짧게 출력되었고, 난이도가 어려워질수록 필요한 동작이 생략된 경우가 많았다. 이는 충분한 예시가 주어지지 않아 모델이 주어진 컨텍스트를 제대로 이해하지 못했기 때문이다[6]. 또한 본 연구에서는 구축한 bin picking 시스템을 통해 언어 명령으로 주어진 타겟 물체를 성공적으로 집어 옮길 수 있는지 UR5e 로봇을 사용하여 평가하였다. [표 2]는 bin picking 시스템의 정량적 평가 결과를 나타낸다. 실험 환경은 Co-DETR 모델이 학습된 COCO 데이터셋에서 로봇 매니퓰레이터가 다룰 수 있는 6종류의 물체를 선별하고, 중복을 허용하여 총 7개의 물체를 랜덤으로 떨어뜨려 구성하였다. 상자 내부 물체들 중 랜덤으로 물체를 지정하여 언어 명령을 주었을 때, 타겟 물체를 집어 옮기는데 0.75의 성공률을 보였다. 실패 사례는 대부분 로봇 동작 과정에서 물체가 미끄러지거나 상자와의 충돌로 인해 발생하였고, 인식 모듈에서 간혹 예측 오류가 발생하였다.

Method	Feasibility			Optimization			|A|/|O|
	Easy	Medium	Hard	Easy	Medium	Hard	
Zero-shot GPT-4	0.9	0.95	0.9	0.5	0.7	0.65	4.4/3.5 (+0.9)
Few-shot GPT-4	1.0	0.75	0.55	0.95	0.7	0.55	3.2/3.5 (-0.3)
Few-shot GPT-4(+CoT)	1.0	0.95	0.85	**1.0**	**0.95**	**0.75**	**3.6/3.5 (+0.1)**

[표 1] 난이도별 Feasibility 및 Optimization 성능과 출력 동작 시퀀스 길이의 평균 (A) 대비 최적 동작 시퀀스 길이의 평균 (O)에 대한 정량적 결과

Method	Success Rate
Ours	0.75

[표 2] Target bin picking에 대한 시스템의 정량적 결과

Ⅲ. 결론

본 논문에서는 상자에 다양한 물체들이 혼재되어 있는 환경에서 사용자가 원하는 타겟 물체에 대한 언어 명령이 주어지면, 하위 작업 계획을 생성하고 로봇 동작으로 타겟 물체를 집어 옮기는 bin picking 시스템을 제안한다. 인식 모듈과 계획 모듈을 구축하였으며, 그래프 난이도별 작업 계획 생성 성능과 전체 시스템에 대한 성능 평가 결과를 제시하였다. 향후 연구에서는 물체들이 복잡하게 혼재되어 있는 환경에서 안정적인 파지 방법을 개발하여 전체 시스템의 성능을 더욱 향상시키고자 한다.

ACKNOWLEDGMENT

본 논문은 2023년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원 (No. RS-2020-II201373, 인공지능대학원지원(한양대학교)과 2024년도 ㈜씨메스사의 지원을 받아 수행된 연구임.

참 고 문 헌

[1] Cordeiro, Artur, et al. "Bin picking approaches based on deep learning techniques: A state-of-the-art survey." 2022 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC). IEEE, 2022.

[2] Zong, Zhuofan, Guanglu Song, and Yu Liu. "Detrs with collaborative hybrid assignments training." Proceedings of the IEEE/CVF international conference on computer vision. 2023.

[3] Kirillov, Alexander, et al. "Segment anything." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2023.

[4] Wei, Jason, et al. "Chain-of-thought prompting elicits reasoning in large language models." Advances in neural information processing systems 35 (2022): 24824-24837.

[5] Achiam, Josh, et al. "Gpt-4 technical report." arXiv preprint arXiv:2303.08774 (2023).

[6] Liu, Jiachang, et al. "What Makes Good In-Context Examples for GPT-3?" arXiv preprint arXiv:2101.06804 (2021).