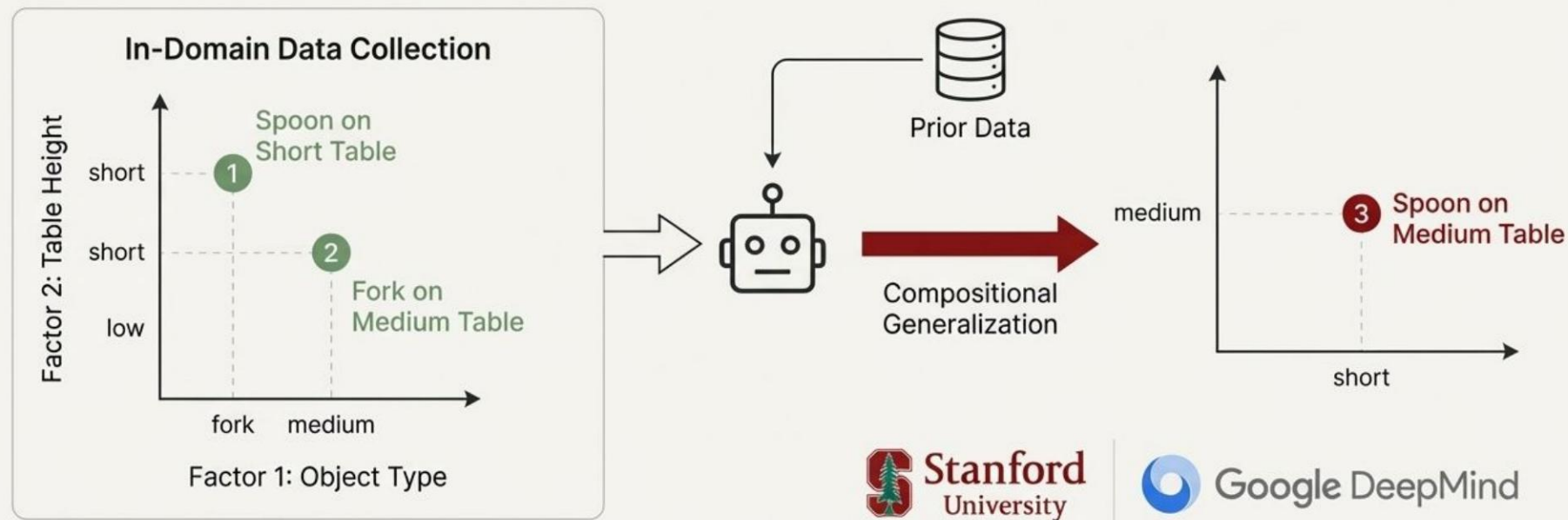


Efficient Data Collection for Robotic Manipulation via Compositional Generalization

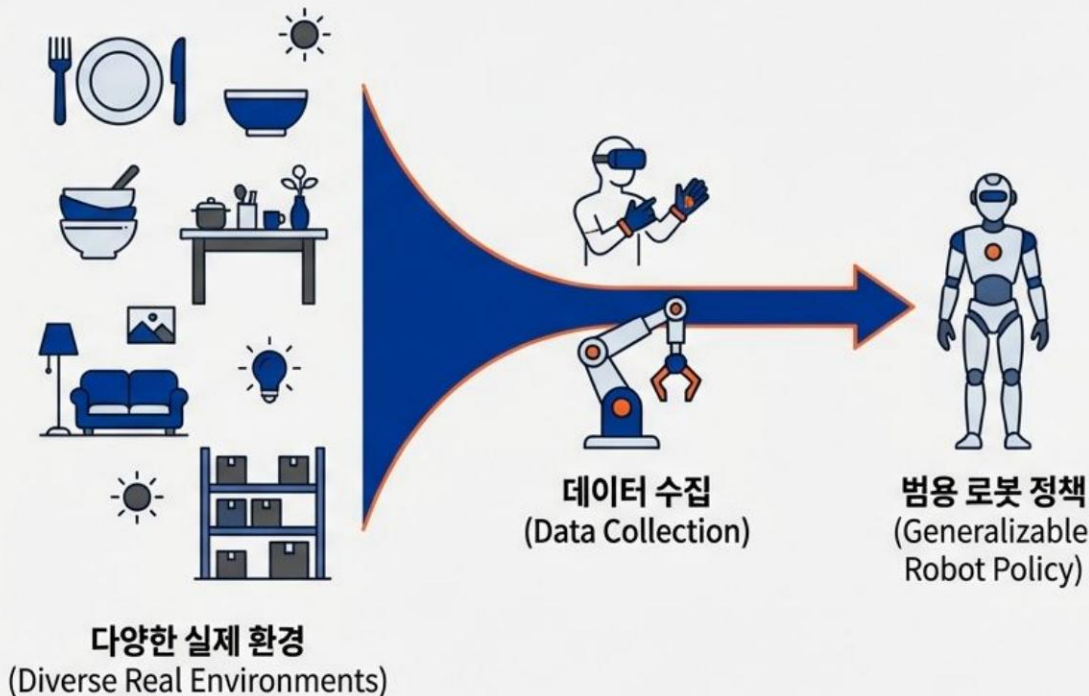
Jensen Gao, Annie Xie, Ted Xiao, Chelsea Finn, Dorsa Sadigh | Stanford University, Google DeepMind



범용 로봇을 훈련시키는 데 가장 큰 병목은 데이터 수집 비용입니다

핵심 포인트

- 로봇이 다양한 실제 환경에 적응하려면 방대한 양의 데이터가 필요합니다. 하지만 로봇 데이터, 특히 인간의 시연 데이터는 수집 비용이 매우 높고 어렵습니다.
- 현재의 대규모 데이터셋(예: BridgeData V2)도 새로운 환경에서는 제로샷(zero-shot) 성능이 부족하여, 결국 해당 환경에 맞는 '인-도메인(in-domain)' 데이터 추가 수집이 불가피합니다.



환경 요소를 모두 다루려는 접근 방식은 기하급수적인 데이터 문제를 야기합니다

로봇이 마주할 환경의 변화 요소를 N 개, 각 요소의 값(예: 물체 종류, 테이블 높이)을 k 개라고 가정해 봅시다. 모든 가능한 조합을 데이터로 수집하려면 $O(k^N)$ 의 노력이 필요합니다. 이는 요소(N)가 몇 개만 추가되어도 데이터 수집량이 비현실적으로 증가함을 의미합니다.

$O(k^N)$ 접근 방식



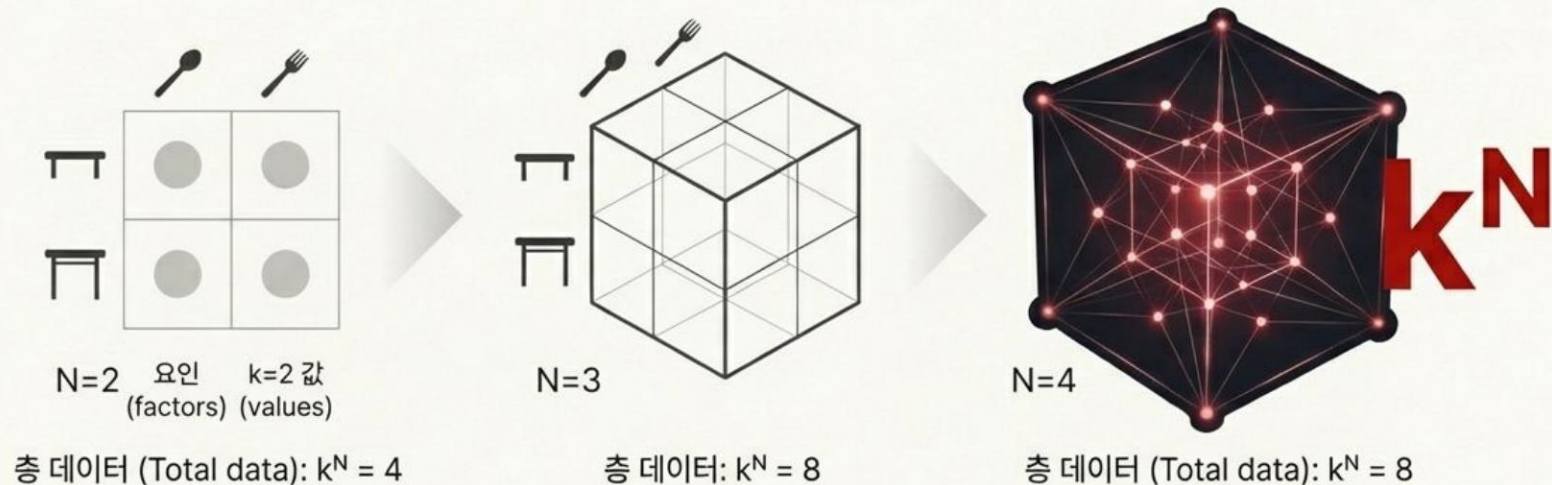
10 Objects x 10 Table Heights = 100 Combinations
10 Objects x 10 Table Heights = 100 Combinations

$O(kN)$ 접근 방식



10 Objects + 10 Table Heights = 20 Examples
10 Objects + 10 Table Heights = 20 Examples

조합의 폭발: 현실적인 데이터 수집의 한계

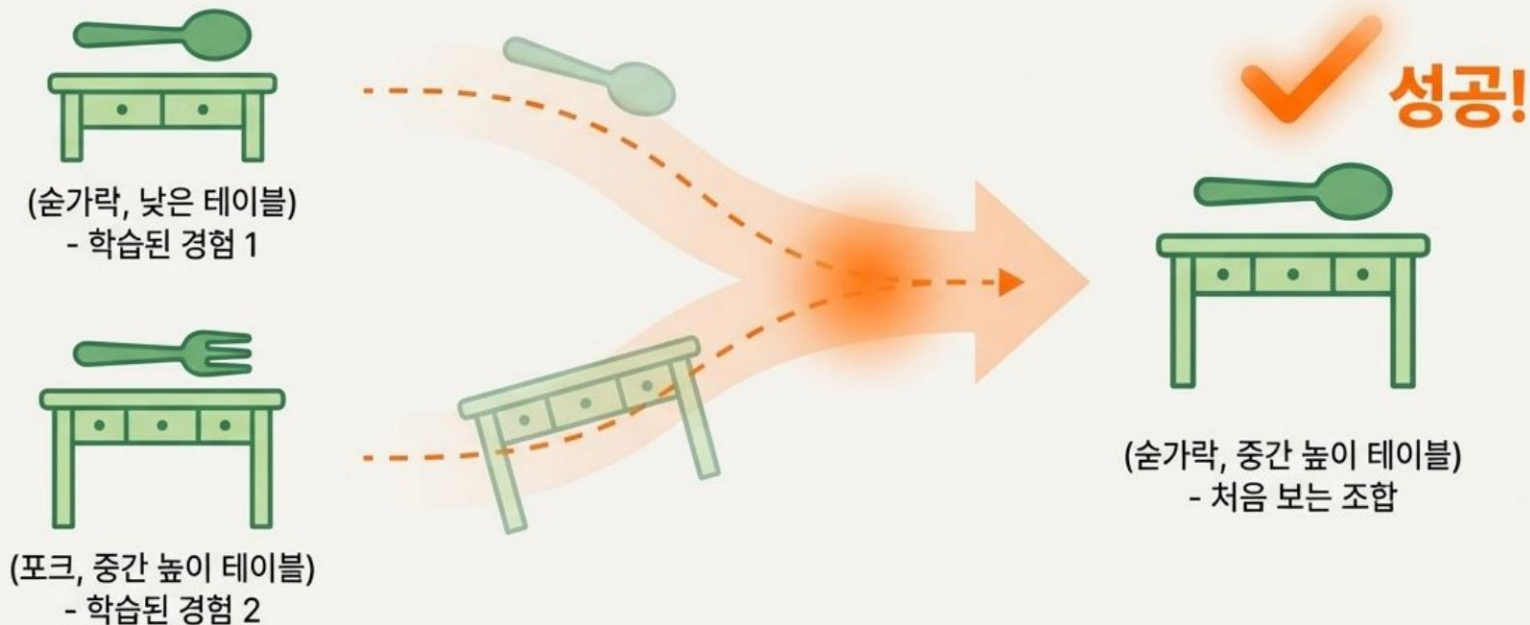


요인이(N) 늘어날수록, 필요한 데이터는 **지수적으로 증가합니다 ($O(k^N)$)**.
이는 곧, 실세계 적용의 불가능을 의미합니다.

모든 개별 요인 값을 커버하는 데는 $O(kN)$ 의 노력이 필요하지만, 모든 조합은 $O(k^N)$ 의 노력을 요구합니다.

로봇이 '개별적으로 학습한 요소'를 '조합'하여 새로운 상황에 대처할 수 있다면? = Compositional Generalization

개별적으로 경험한 환경 요인(factor)들을 조합하여, 한 번도 보지 못한 새로운 환경에 대응하는 능력.



본 연구의 핵심 질문



Q1: 로봇 모방 학습 정책은 실제로 조합 일반화 능력을 보이는가?



Q2: 만약 그렇다면, 조합 일반화를 가장 효과적으로 활용하는 데이터 수집 전략은 무엇인가?

이 질문에 답할 수 있다면

수개월의 데이터 수집
(Months of Data Collection)



스마트한 데이터 수집
(Smart Data Collection)



실세계 로봇 학습에 필요한 **in-domain 데이터 수집 부담**을 획기적으로 줄일 수 있습니다.

제한된 자원으로도 높은 일반화 성능을 가진 로봇 정책을 학습하는
실용적인 가이드라인을 제공할 수 있습니다.

목표 조건 모방 학습(Goal-Conditioned Imitation Learning) 문제 정의

본 연구는 목표 상태 도달 문제를 다음과 같은 튜플로 정의합니다:

$$(\mathcal{S}, \mathcal{A}, \mathcal{T}, H, \mathcal{F}^N, p(f), \rho(s_0|f), \mu(\cdot|f))$$

 $\mathcal{S}$

상태 공간 (로봇 관절각, 물체 위치 등)

 $\mathcal{A}$

행동 공간 (로봇 팔 속도, 그리퍼 개폐 등)

 $\mathcal{F}^N$

요인 공간 (Factor Space). 환경의 N가지 변화 요인을 포착. (예: f_1 = 물체 종류, f_2 = 테이블 높이)

 $p(f)$

요인 조합의 분포. 본 연구에서는 모든 조합에 강건한 정책을 위해 Uniform 분포를 가정.

 $\mu(\cdot|f)$

요인 조합 f 에 따른 목표 상태 g 를 결정하는 함수.

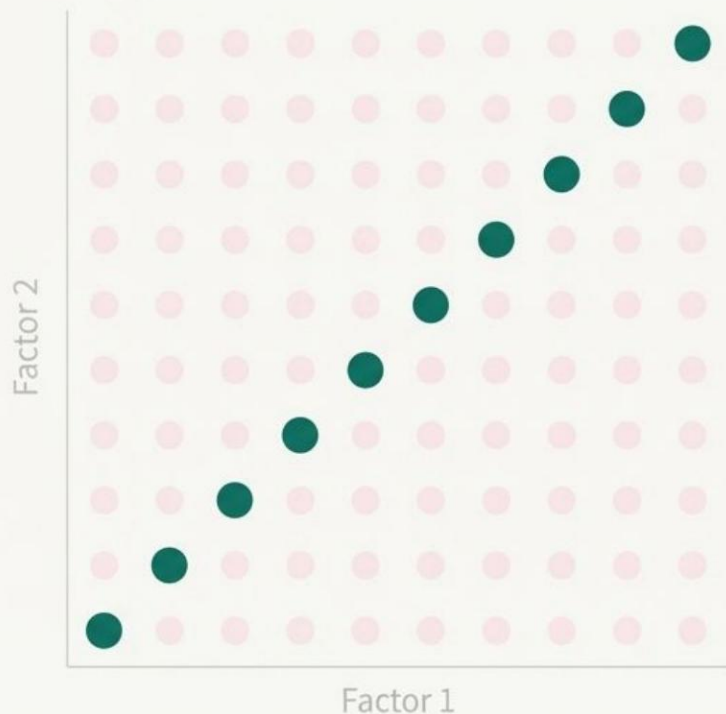
학습 목표는 다음 목적 함수 $J(\pi)$ 를 최대화하는 목표 조건 정책 $\pi(\cdot|s, g)$ 를 학습하는 것입니다.

$$J(\pi) = \mathbb{E}_{f \sim p(f)} [P_{\pi}(s_H = \mu(f))]$$

- **의미:** 다양한 환경 요인(f)이 주어졌을 때, 최종 시점(H)에 목표 상태($\mu(f)$)에 도달할 확률의 기댓값을 최대화합니다.

데이터 수집 전략 1: Diagonal

모든 요인의 각 k 번째 값을 동시에 변경하여, 대각선 패턴으로 모든 요인 값을 한 번씩 커버합니다.



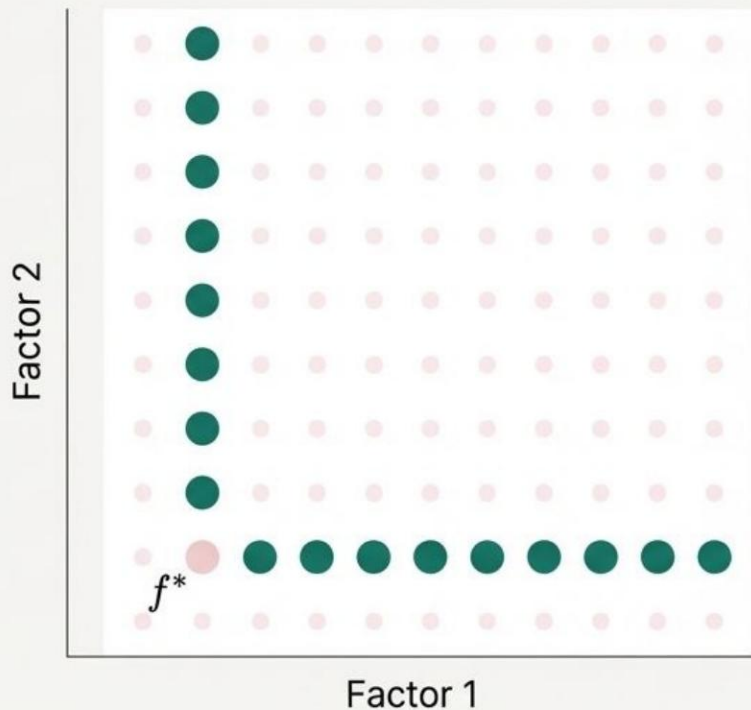
Algorithm 1 Diagonal

Input: scene S , factor values F (N factors $\times k$ values)

```
for  $j \leftarrow 1$  to  $k$  do  
   $f \leftarrow \emptyset_N$   
  for  $i \leftarrow 1$  to  $N$  do  
     $f_i \leftarrow F_{ij}$   
  end for  
  SETFACTORS( $S$ ,  $f$ )  
  COLLECTDATA( $S$ )  
end for
```

데이터 수집 전략 2: L

기준값(f^*)에서 한 번에 하나의 요인만 변경하여 데이터를 수집하고, 각 요인은 개별적으로 모든 값을 탐색합니다.



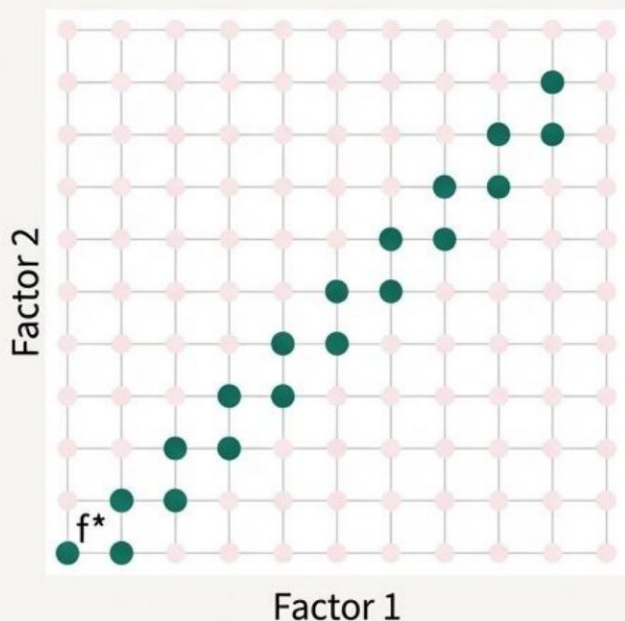
Algorithm 2 L

Input: scene S , factor values F , base values f^*

```
for  $i \leftarrow 1$  to  $N$  do
   $f \leftarrow f^*$  // Reset to base values
  for  $j \leftarrow 1$  to  $k$  do
     $f_i \leftarrow F_{ij}$ 
    SETFACTORS( $S$ ,  $f$ )
    COLLECTDATA( $S$ )
  end for
end for
```

데이터 수집 전략 3: Stair

기본값(f^*)에서 시작하여 요인의 값을 순환적으로 누적 변경하며,
L 전략보다 더 다양한 조합을 효율적으로 커버합니다.



Algorithm 3 Stair

Input: scene S , factor values F , base values f^*

$f \leftarrow f^*$ // Initialize with base values

```
for  $j \leftarrow 1$  to  $k$  do
  for  $i \leftarrow 1$  to  $N$  do
     $f_i \leftarrow F_{ij}$ 
    SETFACTORS( $S$ ,  $f$ )
    COLLECTDATA( $S$ )
  end for
end for
```

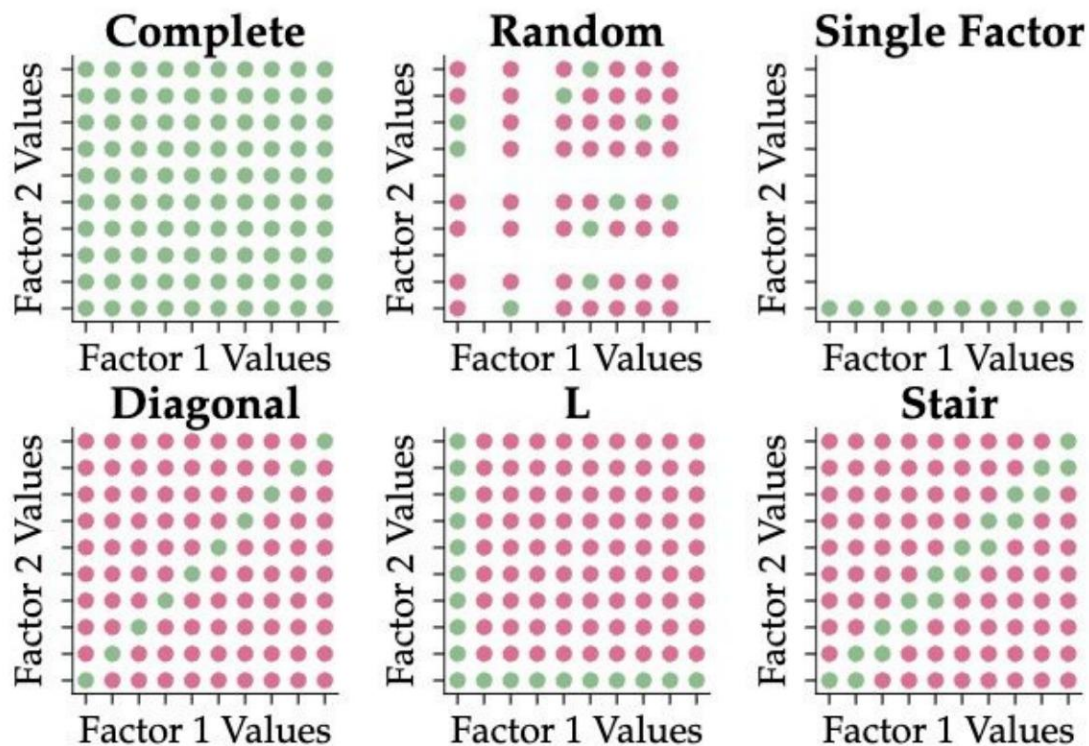


Fig. 2: Visualization of our data collection strategies with $N = 2$ factors. Each axis consists of possible values for a factor. Each green dot indicates that the strategy captures a specific combination of factor values represented by it, and each pink dot represents a combination that compositional generalization may address. We name our strategies based on the patterns in this visualization.

전략 비교: 노력(Effort)과 조합 커버리지

전략 (Strategy)	노력 (Factor Changes)	조합 커버리지 (Combination Coverage)	핵심 아이디어
L	$O(kN)$	낮음 (축 기준)	개별 요인 분리
Diagonal	$O(kN)$	매우 낮음	새로운 값의 조합
Stair	$O(kN)$	중간	점진적 조합 탐색
Random	가변적	무작위	구조 없음
Complete	$O(k^N)$	전체	비현실적
No Variation	상수	단일	기준선

우리의 가설 (Our Hypotheses)



가설 1: 조합 전략의 우수성

Stair, L, Diagonal은
Random보다 우수할 것이다.
(조합 일반화 활용)



가설 2: 효율성의 극대화

조합 능력이 강할 경우, Stair, L,
Diagonal은 Complete 전략의
성능에 근접할 수 있다.



가설 3: Stair의 잠재적 우위

Stair는 더 많은 양과 다양한 조합을
커버하므로 L, Diagonal보다
우수할 수 있다.

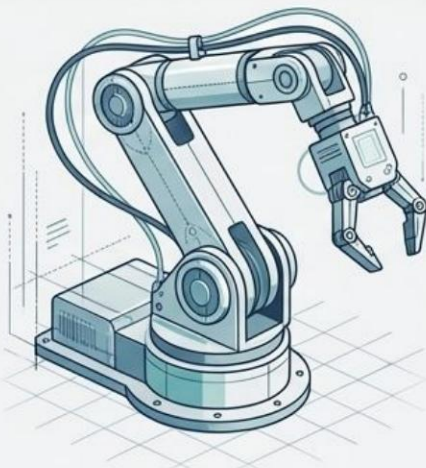


가설 4: 사전 데이터의 역할

기존 로봇 데이터(Prior Robot
Data)를 통합하면 in-domain
데이터의 조합 능력을 강화할 수 있다.

구성 능력 일반화(Compositional Generalization) 검증을 위한 실험 환경 설계

시뮬레이션 및 실제 로봇 환경에서의 체계적 접근



시뮬레이션 환경의 변수 설정

N = 5

5가지 주요 환경 요소를 정의합니다.



물체 위치
(object position)



물체 종류
(object type)



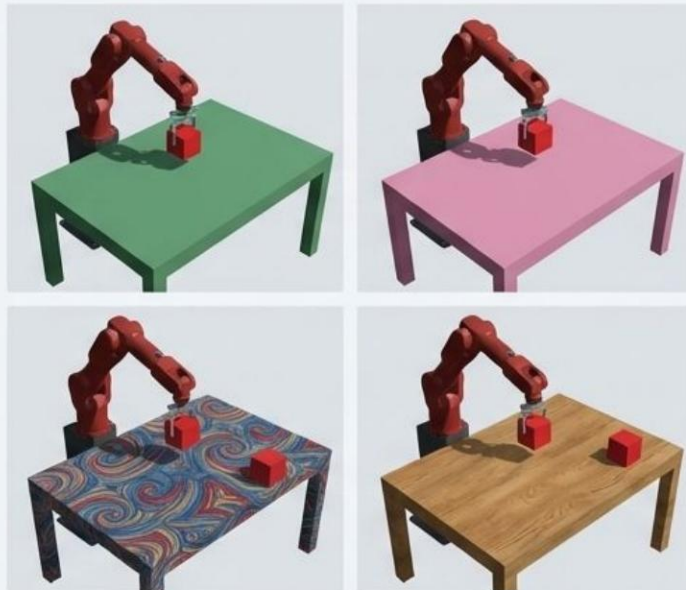
테이블 질감
(table texture)



카메라 위치
(camera position)



방해물
(distractor objects)



k = 10

각 요소는 10개의 고유한 값을
가질 수 있도록 설정됩니다.

실험 시나리오 (Experimental Scenarios)

N=2 설정: 5개 요소 중 가능한 모든 2가지 요소 조합(총 10개)에 대한 쌍별(pairwise) 구성 능력 평가.

N=5 설정: 5개 요소를 모두 동시에 고려하여 더 복잡하고 현실적인 구성 능력 평가.

실제 환경 (Real World)



Setup

Robot: WidowX 250 6DOF 로봇 팔

Environment: 실제 오피스 주방 환경

Task: 포크를 그릇 안에 넣기

Factors Studied

물리적 요소에 중점: 포크 종류, 포크 위치, 그릇 종류, 테이블 높이 등 5가지 핵심 요소를 변화시켰습니다.

Fork Variations



Container Variations



Table Height Variations



Table Texture Variations



실제 환경의 변수 설정

N = 5, k = 4

5가지 주요 물리적 요소에 대해 각각 4개의 변수를 설정합니다.



물체(포크) 종류
(object type)



물체(포크) 위치
(object position)



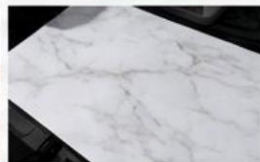
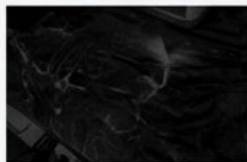
용기 종류
(container type)



테이블 높이
(table height)



테이블 질감
(table texture)



보조 실험 요소
(Additional Factors for
Secondary Experiments):
물체/용기 위치, 물체 방향, 방해물,
카메라 위치

데이터 수집 방법론: 자동화된 정책 vs. 인간 시연

Simulation

**스크립트 기반 전문가 정책
(Scripted Expert Policy)**



자동화된 전문가 정책을 사용하여 대규모의 일관된 데이터를 효율적으로 생성합니다.

데이터 수집 노력(effort)은 '총 요소 변경 횟수(total factor changes)'로 측정됩니다.

Real World

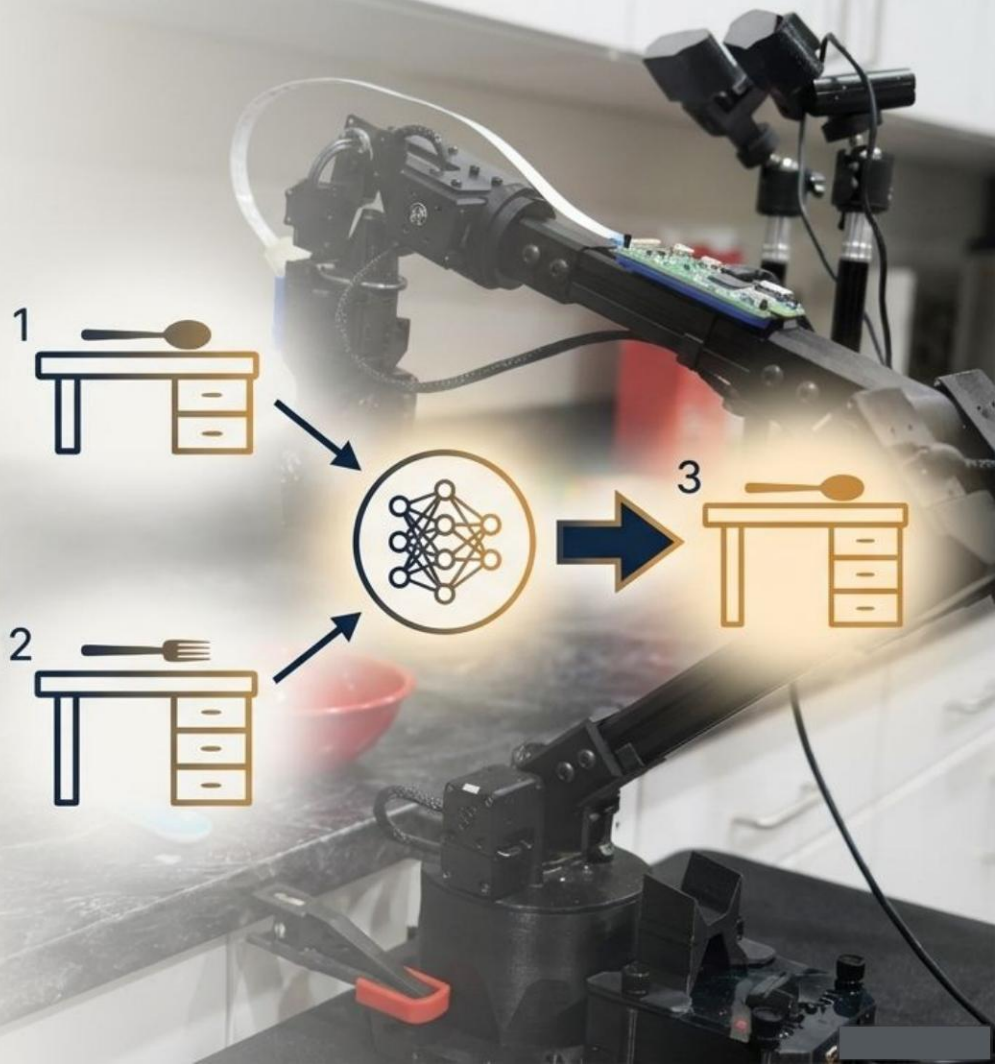
**인간 시연
(Human Demonstrations)**



Meta Quest 2 VR 헤드셋을 이용한 원격 조작(teleoperation)을 통해 숙련된 인간 시연자가 데이터를 수집합니다.

모델 학습 방법론: 조합적 일반화 (Compositional Generalization)를 위한 정책 학습

본 발표는 로봇 정책이 학습 데이터에 없는 새로운 요소 조합에 대응할 수 있도록 하는 특정 모델 학습 전략을 심도 있게 다룹니다.



핵심 전략: 사전 데이터 기반 2단계 학습 파이프라인

대규모 사전 데이터셋(BridgeData V2)의 광범위한 일반화 능력과 인-도메인 데이터의 특수성을 결합하는 2단계 학습 전략을 채택합니다.



일반화 성능 강화를 위한 데이터 증강

- 목표: 학습 데이터의 다양성을 인위적으로 높여 모델의 일반화(generalization) 성능을 향상.
- 기대 효과: 과적합(overfitting)을 방지하고 강건한(robust) 정책을 유도.

적용된 기법 (Applied Techniques)



무작위 이동
(Random Shift)

색상 지터
(Color Jitter)

무작위 자르기 & 크기 조절
(Random Crops & Resizing)

원본 이미지
(Original Image)

Composition, Scaling

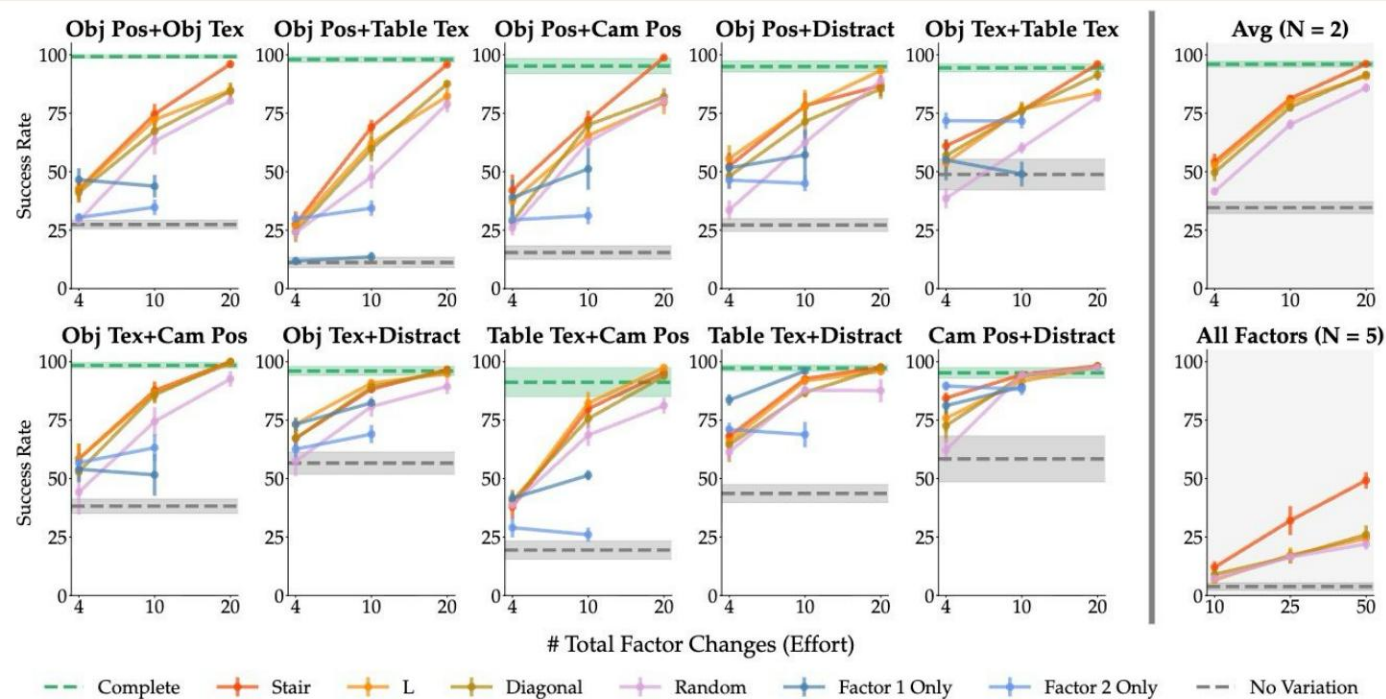


Fig. 4: Simulation results of data collection strategies for *Pick Place*. We report results where $\mathcal{F}^N$ consists of each possible factor pair ($N = 2$), average results across all pairs, and results where $\mathcal{F}^N$ consists of all factors ($N = 5$). All points within the same subplot use the same amount of demonstrations. The strategies that exploit composition (**Stair**, **L**, **Diagonal**) generally outperform **Random**, and often approach **Complete**. **Stair** generally performs the best, especially in the $N = 5$ setting. Error bars represent standard error across 5 seeds.

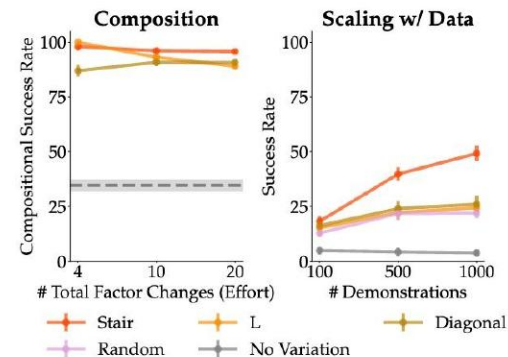


Fig. 5: (left) Compositional success rate of different strategies. (right) Generalization of strategies with increasing dataset sizes.

검증 : 실제 로봇의 조합 능력, 사전 데이터가 결정적이다

Data Strategy	L								No Variation			
Train Method	Bridge				From Scratch				Bridge			
Factor 1 \ Factor 2	Object Type	Container Type	Table Height	Table Tex	Object Type	Container Type	Table Height	Table Tex	Object Type	Container Type	Table Height	Table Tex
Object Pos	8/9	5/9	2/9	5/9	0/9	0/9	0/9	1/9	3/9	2/9	1/9	1/9
Object Type		8/9	8/9	8/9		5/9	5/9	4/9		4/9	3/9	2/9
Container Type			5/9	6/9			4/9	6/9			2/9	3/9
Table Height				4/9				3/9				1/9
Overall	59/90				28/90				22/90			

TABLE I: Real robot pairwise composition results for our “*put fork in container*” task. When leveraging BridgeData V2 as prior data, a policy is able to compose the factor values present in the **L** data to succeed on **59/90** compositional combinations of factor values. Without prior data, the model is unable to compose nearly as effectively, with compositional success rate dropping by roughly half. Prior data alone is also not enough to generalize to these situations, as a policy trained with prior data on **No Variation** also performs poorly.

체계적인 데이터('L' 전략)는 중요하지만,
대규모 사전 데이터(Bridge) 없이는 그 효과가 절반으로 떨어집니다.

검증 : 최적의 학습 방법론은 End-to-End

Data Strategy	L				No Variation		
Train Method	Bridge	R3M	VC-1	From Scratch	Bridge	R3M	VC-1
Object Type + Table Tex	8/9	1/9	0/9	4/9	2/9	0/9	0/9
Object Pos + Table Tex	5/9	0/9	0/9	1/9	1/9	0/9	0/9
Object Pos + Table Height	2/9	1/9	0/9	0/9	1/9	0/9	0/9
Overall	15/27	2/27	0/27	5/27	4/27	0/27	0/27

TABLE II: Additional real robot pairwise composition results for our “*put fork in container*” task with R3M and VC-1.

사전 학습된 시각 인코더를 고정하는 방식은 실패했으며, 사전 데이터와 현재 데이터를 함께 학습(Co-fine-tuning)하는 것이 압도적으로 우수합니다.

최종 검증: 배운 것을 낯선 곳에서 증명하다

Data Strategy		Stair			L			No Variation	
Train Method		Bridge (Co-FT)	Bridge (FT)	From Scratch	Bridge (Co-FT)	Bridge (FT)	From Scratch	Bridge (Co-FT)	Bridge (FT)
No Further Changes	<i>CompKitch</i>	4/5	4/5	0/5	5/5	0/5	0/5	0/5	0/5
	<i>TileKitch</i>	5/5	2/5	0/5	5/5	5/5	3/5	0/5	0/5
Fork Down + Blue Plate	<i>CompKitch</i>	4/5	4/5	0/5	3/5	0/5	0/5	1/5	0/5
	<i>TileKitch</i>	4/5	3/5	0/5	3/5	5/5	5/5	0/5	0/5
Wooden Fork + Pink Bowl	<i>CompKitch</i>	3/5	5/5	0/5	2/5	0/5	0/5	0/5	0/5
	<i>TileKitch</i>	4/5	3/5	0/5	3/5	5/5	2/5	0/5	0/5
Plastic Fork + Higher Table	<i>CompKitch</i>	4/5	0/5	0/5	1/5	0/5	0/5	0/5	0/5
	<i>TileKitch</i>	3/5	0/5	0/5	2/5	0/5	1/5	0/5	0/5
Overall	<i>CompKitch</i>	15/20	13/20	0/20	11/20	0/20	0/20	1/20	0/20
	<i>TileKitch</i>	16/20	8/20	0/20	13/20	15/20	11/20	0/20	0/20
	Combined	31/40	21/40	0/40	24/40	15/40	11/40	1/40	0/40

TABLE III: Out-of-domain transfer results to new kitchens *CompKitch* and *TileKitch*. We find that varied in-domain data from *BaseKitch*, and BridgeData V2 as prior data, are both critical for effective transfer to these new kitchens. **Stair** outperforms **L**, although both achieve significant levels of transfer. Co-fine-tuning generally performs better than only fine-tuning.

CompKitch



CompKitch

TileKitch



TileKitch

체계적 데이터 전략과 사전 데이터의 결합은, 로봇이 한 번도 본 적 없는
부엌에서 도 **77.5%의 높은 성공률**로 임무를 완수하게 했습니다.

연구 결론 및 핵심 기여



Composition 능력 확인

로봇 정책은 상당한 Composition 능력을 보이거나, 물리적 요인 조합은 여전히 난제임.



Prior Data의 중요성

실제 환경에서의 Composition 능력 발현을 위해 Prior Data 활용 및 Co-fine-tuning은 필수적임.



효율적 전략 검증

제안된 전략들은 데이터 수집 노력을 줄이면서 보지 못한 환경으로의 일반화를 가능하게 함.



Stair 전략의 우수성

시뮬레이션과 실제 환경 모두에서 가장 우수한 Composition 및 이전(Transfer) 성능을 보임.

향후 연구 방향 및 논의

- 더 복잡한 과제 및 로봇 플랫폼으로의 확장
- 물리적 요인 간의 상호작용 심층 분석
- 최적의 '사전 데이터' 구성 탐구



Discussion

- 해당 논문은 정책이 다양한 환경 요인에 대한 조합에 일반화가 가능함을 보인다. 그렇다면 이 정책은 **요인들간의 관계**를 가지고 있을까?(e.g. **A가 B 위에 있다**) 만약 가지고 있다면, 관계를 명시적으로 추출하거나 분석할 수 있는 방법은 무엇인가? 명시적으로 추출할 수 있다면, **새로운 요인 조합**, 새로운 형태의 관계도 일반화할 수 있지 않을까?
- Prior data가 compositional generalization에 결정적인 역할을 하는데, 이때 prior data가 로봇 정책에 부여한 이점이 무엇일까? 일반화가 될 수 있도록 지식을 전이한다면, 어떤 지식을 전이할까?
- 고려하지 못한 요인 (방해물, 조명 변화 등)에 대해 선행 데이터가 강건성을 부여한다고 논문에서 언급하는데, 로봇이 이전에 전혀 정의되지 않고 새롭게 발생하는 환경 요인에 대해서도 compositional generalization을 발휘할 수 있을까?