3D Point Cloud Affordance Detection for Robotics Applications

Application Level Presentation

Lasse Huber-Saffer

Master Seminar Intelligent Robotics
Technical Aspects of Multimodal Systems
Department of Informatics
University of Hamburg

June 5, 2025

Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Table of Contents

Manipulation Task Planning

- Scenario
  - Robot arm, camera
  - Unseen environment
  - Textual task specification
- Goal: Plan trajectory for manipulation task

Motivation

- Many tasks require fine-grained understanding of scene and prompt
  - "Pick up the knife"
    - Avoid touching the blade
  - "Put the flower into the vase"
    - Avoid touching the petals
    - Find vase opening
  - "Lift the handbag"
    - Grip on handle to avoid dropping contained items

$\Rightarrow$ Object detection alone not sufficient

- **Emergent task: Visually identify task-specific actionable areas**

3D Affordance Detection

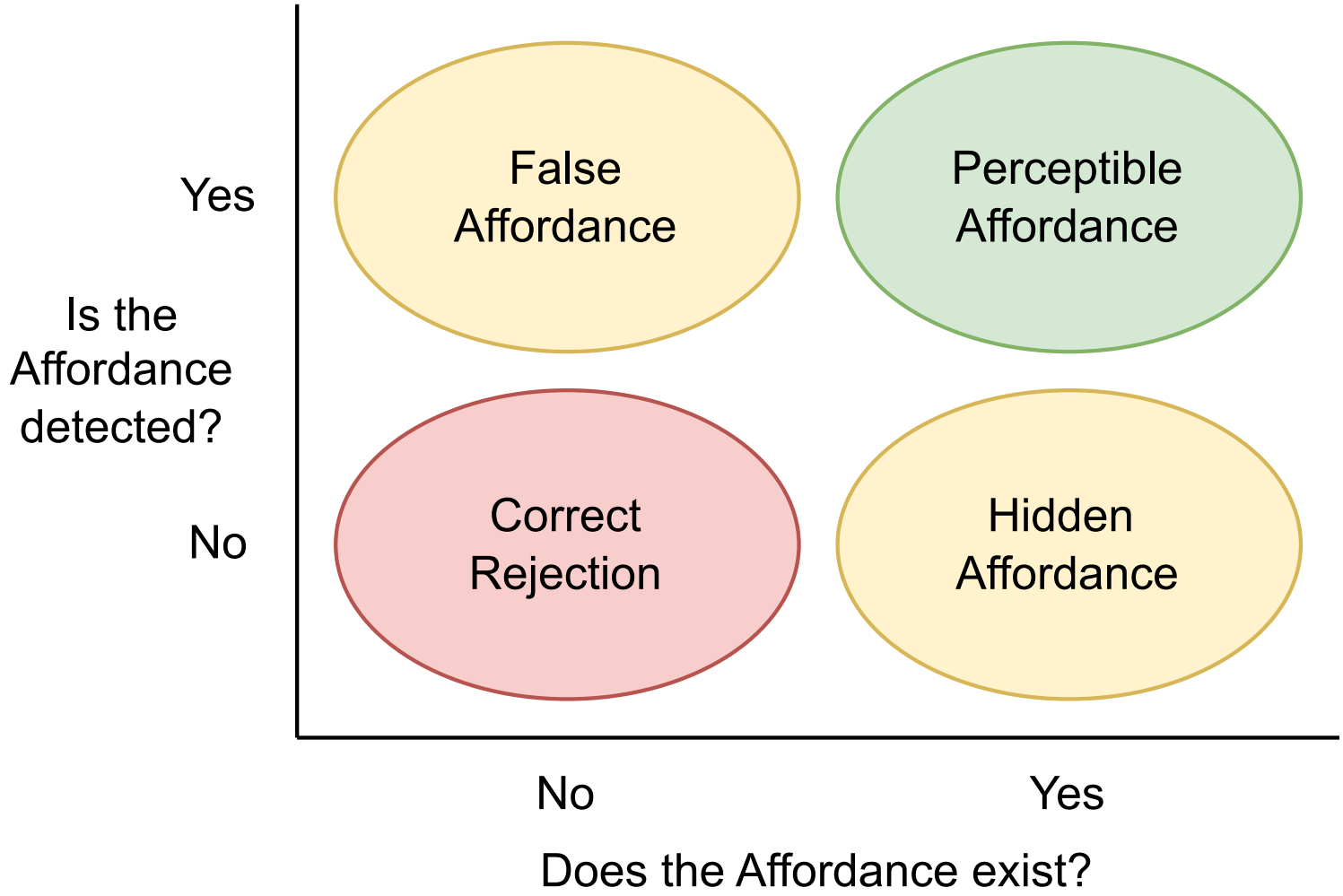
- Input: 3D representation of objects/environment
  - i.e., depth camera point cloud
- Output: Affordance labels for individual regions
  - Affordance = *Possible action exhibited by object or environment*[1]

Figure: Example affordance labels[2]

[1] Gibson, 1966

[2] Nguyen et al., IROS 2023

Affordance Types[3]



[3] Gaver, CHI 1991

Challenges

- Affordances depend on observer's capabilities
  - Different gripper/robot types facilitate different actions
- False affordances
  - Touch blade of knife ⇒ damage robot
  - Pick up flower by the petals ⇒ damage object
- Incomplete information & hidden affordances
  - ! Sensors only produce partial scene representations
  - "Pull the green lever" ⇒ What if it isn't visible?
  - "Open the drawer half-way" ⇒ How far can it extend?
  - "Pick up the heaviest cube" ⇒ How heavy are the cubes?
  - May require multi-step exploration/experimentation with continuous feedback

Closed Vocabulary Approaches

- Predetermined closed set of affordances
  - i.e., "grasp", "lift", "pour", "cut"
- Generalizability limited by dataset scope
  - Known affordance types limit possible actions
  - Known object types limit viable environments
  - ! Broadening the scope drastically complicates data collection
- Typically object-level
  - Disregard contextual information of scene

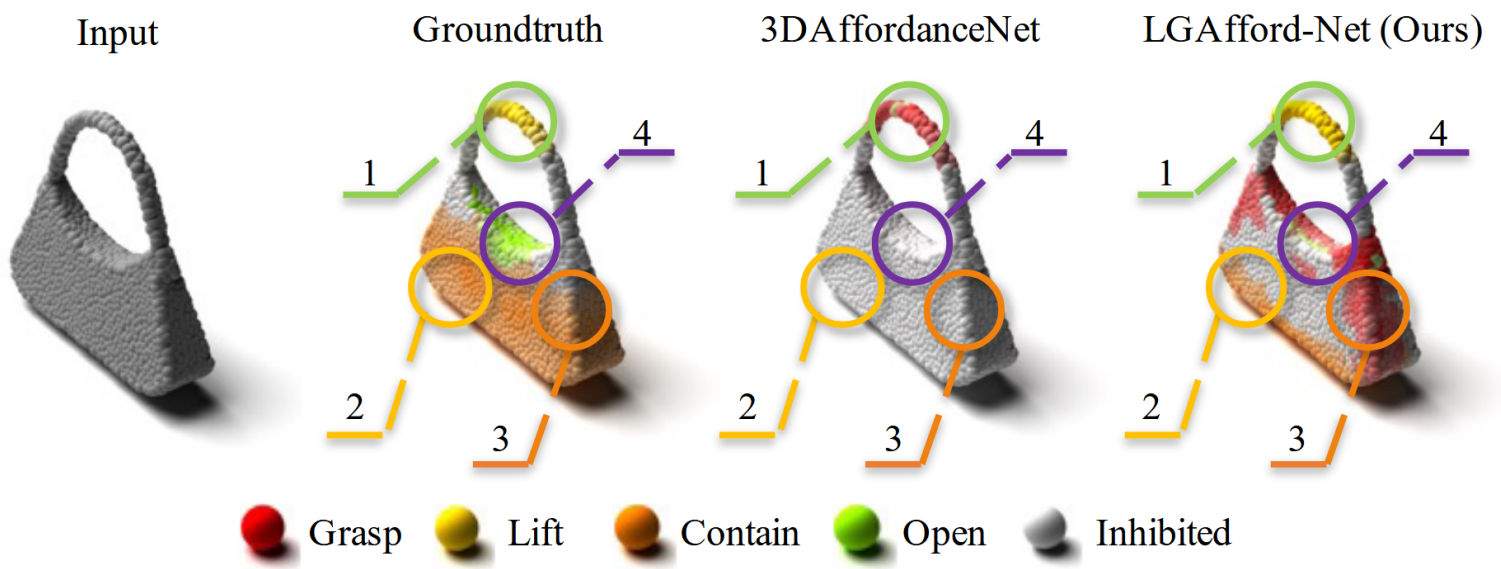
3D AffordanceNet[4]

- Object-level dataset
  - 23 object types
  - 18 affordance types
  - 23k full-shape point clouds
  - 56k affordance annotations
- Common benchmark for 3D affordance detection
  - 2 baseline methods (PointNet++, DGCNN)

[4] Deng et al., CVPR 2021

LGAfford-Net[5]

- Trained on 3D AffordanceNet dataset
- Emphasis on local geometry
- Improved accuracy and object type generalizability



[5]Tabib et al., CVPR 2024

Open Vocabulary Approaches

- Generalization to open-set text prompts
- Varying degrees of complexity
  - Unseen affordance labels ("increase volume")
  - Referring expressions ("open the smallest green glass bottle")
  - Physical understanding ("place on a stable surface")
  - Freeform instructions ("First, unpack the shopping bags and place the items in their correct fridge compartments. Then prepare a vegetarian sandwich for me.")
- Requires profound understanding of prompt and environment

  $\Rightarrow$ **Previously infeasible**

Foundation Models

- Large machine learning models
- Pre-trained on extensive and diverse datasets
  - Costly dataset collection, training and inference
- Achieve human-level understanding of their tasks
  - **Recent breakthrough!**
- Impressive transfer capabilities
  - Zero-shot $\Rightarrow$ Directly use pre-trained model
  - Few-shot $\Rightarrow$ Fine-tune on small dataset

Foundation Models (Cont.)

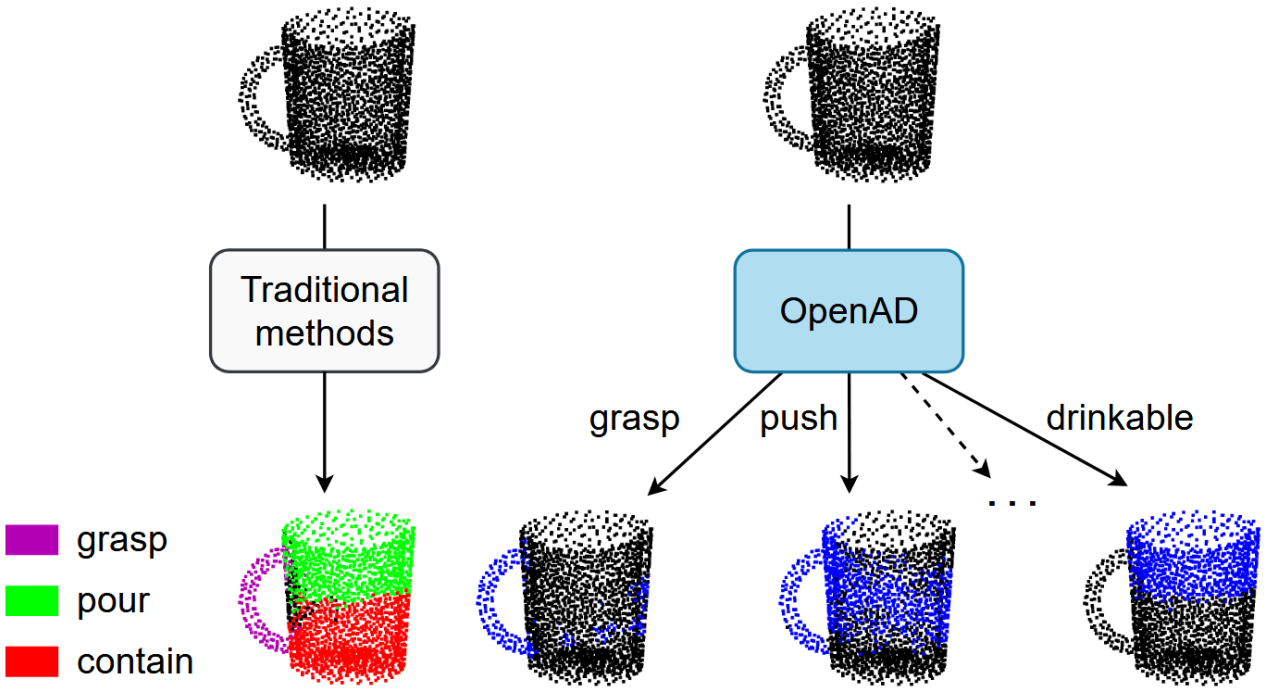
- Examples
  - Language: GPT[6], LLaMA[7], DeepSeek-R1[8]
  - Vision: SAM 2[9], DINO-X[10]
  - Multimodal: LLaVA[11], GPT-4V[12], CLIP[13]

⇒ **Useful for affordance detection and other robotics applications**

[6] Radford et al., OpenAI, 2018
[7] Touvron et al., arXiv, 2023
[8] Guo et al., arXiv, 2025
[9] Ravi et al., arXiv, 2024
[10] Ren et al., arXiv, 2024
[11] Liu et al., NeurIPS 2023
[12] Achiam et al., OpenAI, 2023
[13] Radford et al., arXiv, 2021

OpenAD[14]

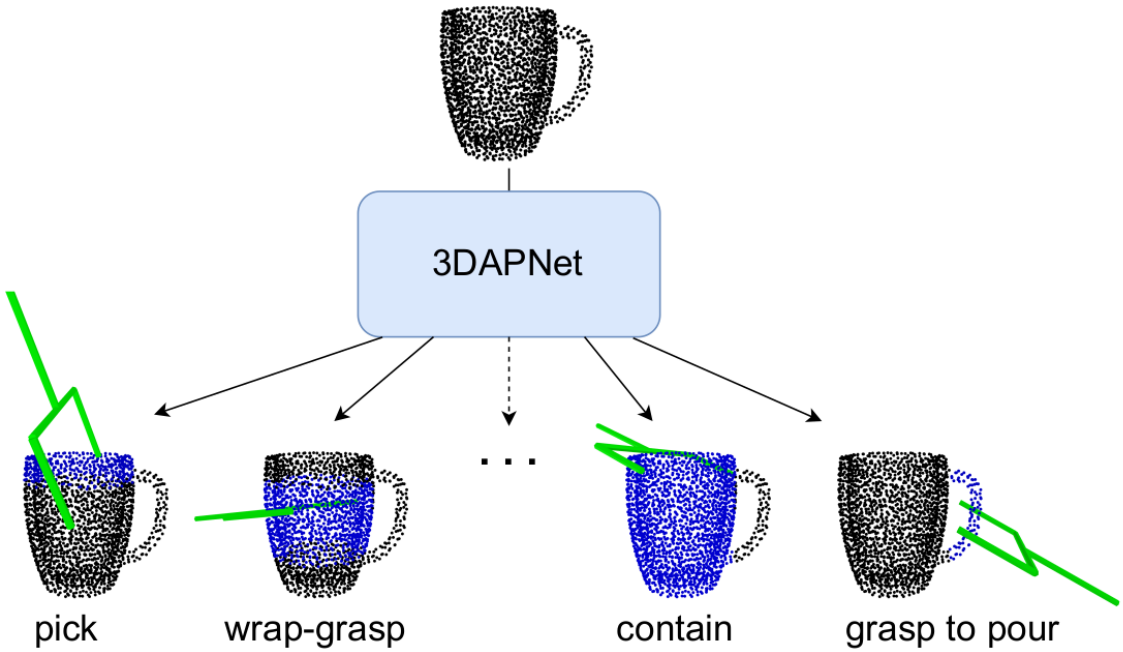
- Generalization of closed vocabulary approaches
- Prompt scope: Unseen affordance labels
  - i.e., "drinkable", "display", "take a seat"
- Inherits text encoder from CLIP
- Improved multi-affordance detection



[14] Nguyen et al., IROS 2023

3DAPNet[15]

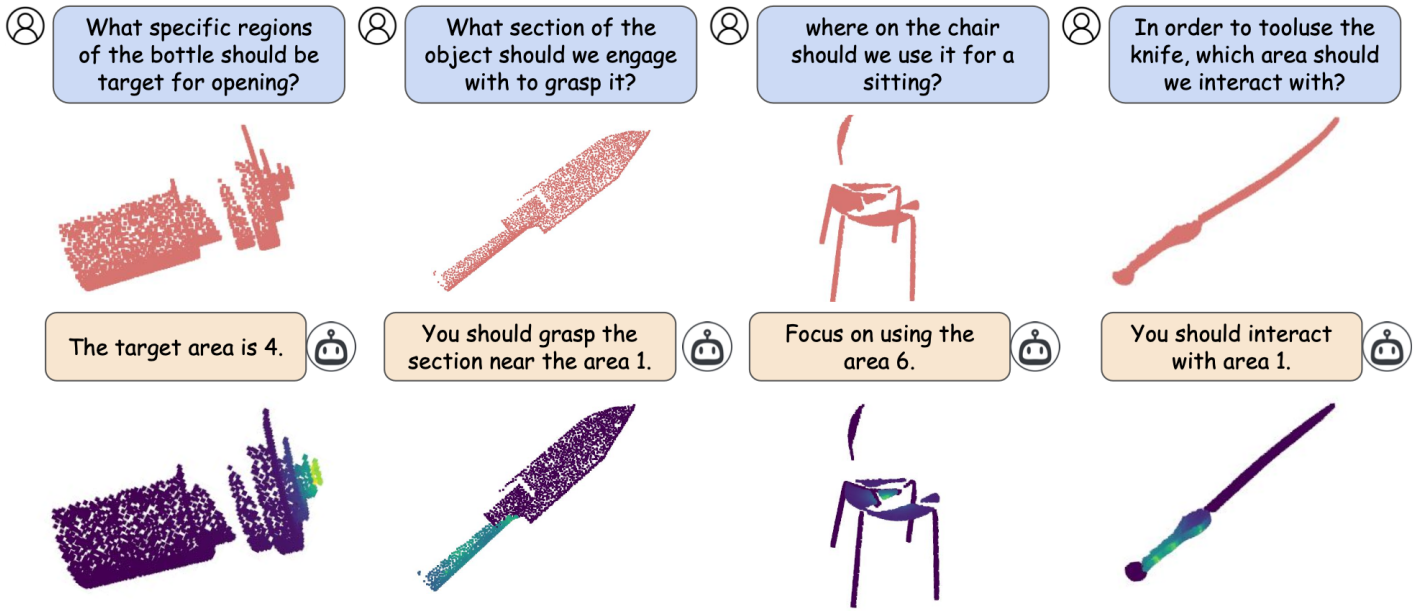
- Evolution of OpenAD
- Prompt scope: Unseen affordance labels
- Simultaneously predicts affordance and 6-DoF grasp pose
- Improved benchmark performance



[15] Nguyen et al., ICRA 2024

PAVLM[16]

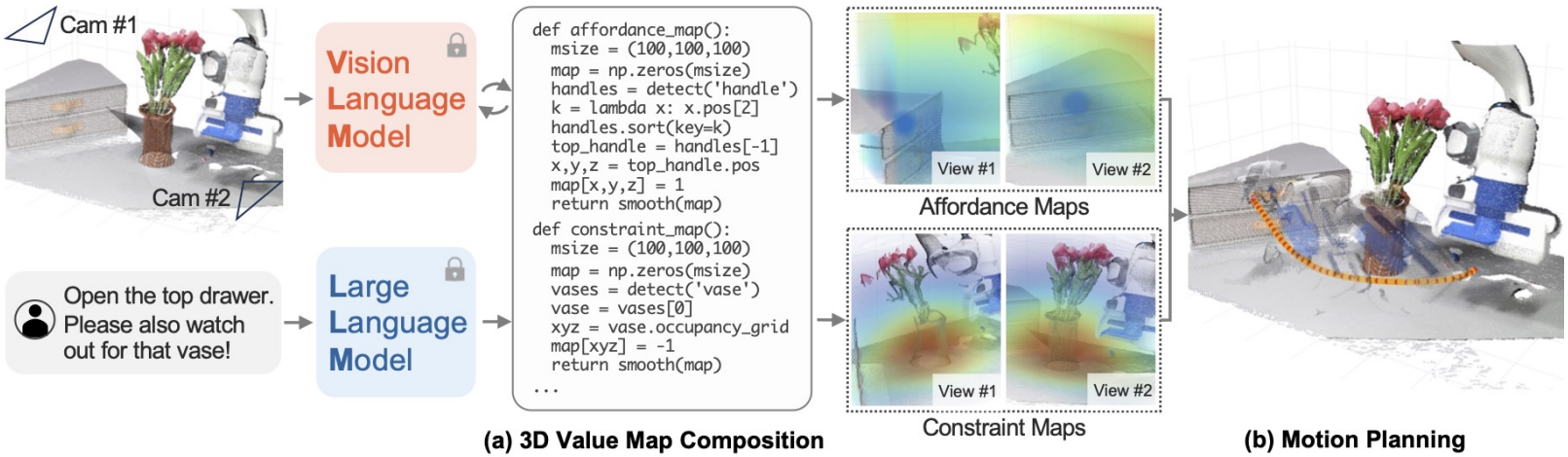
- Prompt scope: Freeform instructions
  - i.e., "Which part of the scissors should be used for grasping?"
- Utilizes separate LLM and VLM components
- Improved benchmark results and open-world generalization capabilities
- Handles partial point clouds particularly well



[16] Liu et al., arXiv, 2024

VoxPoser[17]

- Prompt scope: Freeform instructions
- Separate LLM and VLM components (**zero-shot transfer!**)
- Predicts task-specific cost/reward function on scene voxel grid
- Synthesizes trajectories for task
- Closed-loop visual feedback enables robustness to dynamic environmental changes



[17] Huang et al., CoRL 2023

CoPa[20]

- Prompt scope: Freeform instructions
- More advanced VLM component
  - ⇒ No need for separate LLM
- Pose candidate generation using GraspNet[18]
- Motion planning module solves for spatial constraints of relevant object parts
  - ⇒ How should they move in relation to eachother?
- Broad zero-shot generalization capabilities
- Integration into high-level planning method like ViLa[19] enables solving of long-horizon tasks

[18] Fang et al., CVPR 2020
[19] Hu et al., arXiv, 2023
[20] Huang et al., IROS 2024

CoPa (Cont.)

- Results
  - Coffee machine demo (Video)
  - Romantic dinner demo (Video)

Summary

- Complex manipulation tasks require fine-grained object understanding
- Legacy approaches offer limited generalizability
  - Affordance vocabulary
  - Object types
- Recent approaches leverage pre-trained foundation models
  - Task-oriented scene understanding
  - Open-set textual task understanding
  - Object part segmentation
  - Grasp pose candidate generation
- Applicability of high-level planning and continuous feedback
  - Explore missing/ambiguous information
  - Adapt to dynamic changes