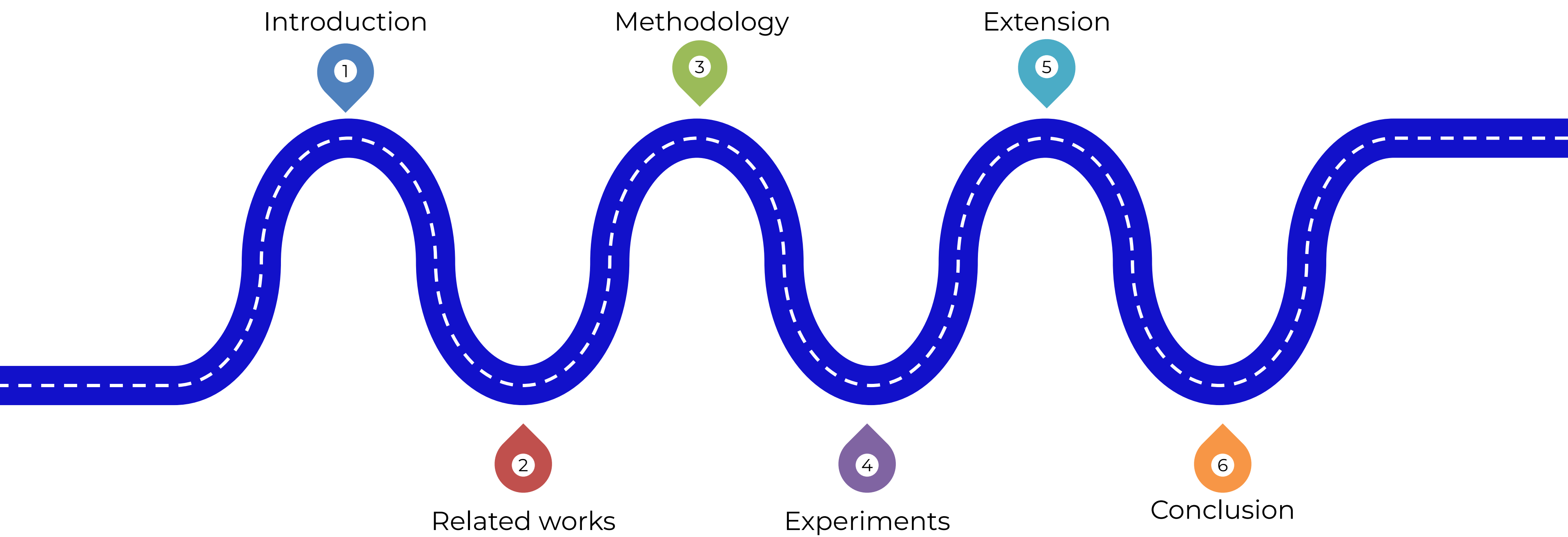
Point Cloud Affordance Highlighter

Amirali Changizi - s324771
Meelad Dashti - s328715
Kimia Dorrani - s329154

Politecnico di Torino

Roadmap



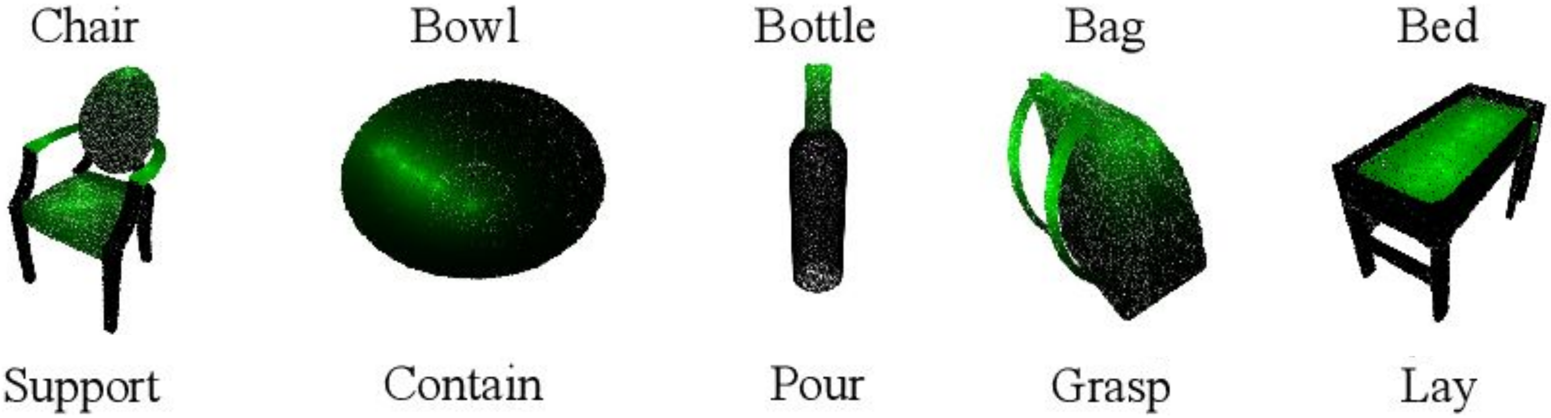
Introduction
1
Related works
2
Methodology
3
Experiments
4
Extension
5
Conclusion
6

Introduction

Using Neural Fields and Vision-Language Models for Unsupervised Affordance Detection

What are Affordances?

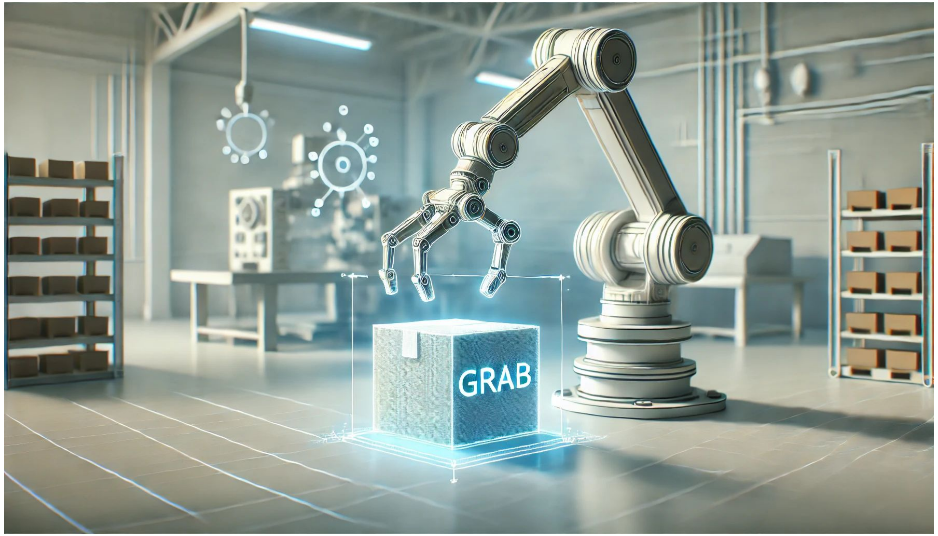
- Regions on objects that support meaningful interactions
  - Bag → Grasp
  - Bed → Lay
  - Bottle → Pour
  - Chair → Support



Why is Affordance Detection Important?

- Critical for robotics and human-object interaction

- Enables:
  - Autonomous manipulation
  - Scene understanding
  - Human-robot collaboration



Current Challenges

- Reliance on labeled datasets
- Limited to specific object categories
- Complex geometric features
- Limited generalization

Related Work

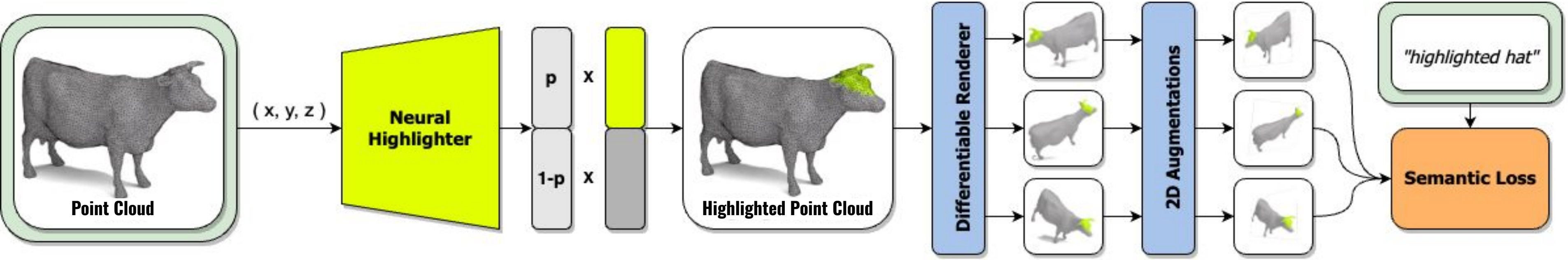
- Traditional Affordance Detection
- 3D AffordanceNet
- Interaction-driven 3D Affordance Grounding (IAG)
- Vision-Language Models in 3D (e.g., CLIP-based)

Our Approach

- Label-free pipeline using:
  - Neural fields
  - Pre-trained vision-language models (CLIP)
  - Differentiable point cloud rendering

- Benefits:
  - No manual annotations needed
  - Works with real-world sensor data
  - Flexible across different 3D formats

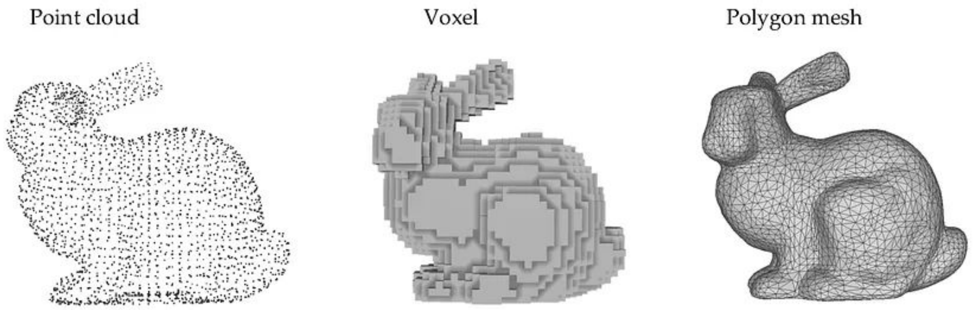
Pipeline Overview



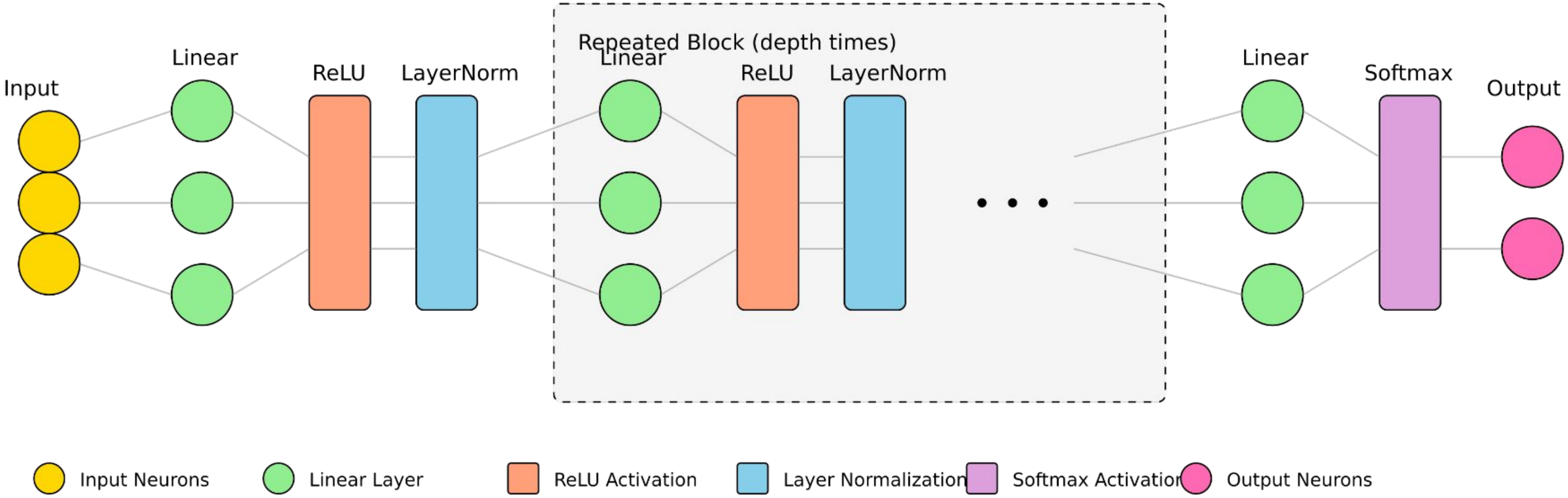
Point Cloud Processing

- Three Potential Approaches:
    1. Direct Point Cloud Rendering (Selected)
    2. Mesh Approximation
    3. Voxel-based Conversion

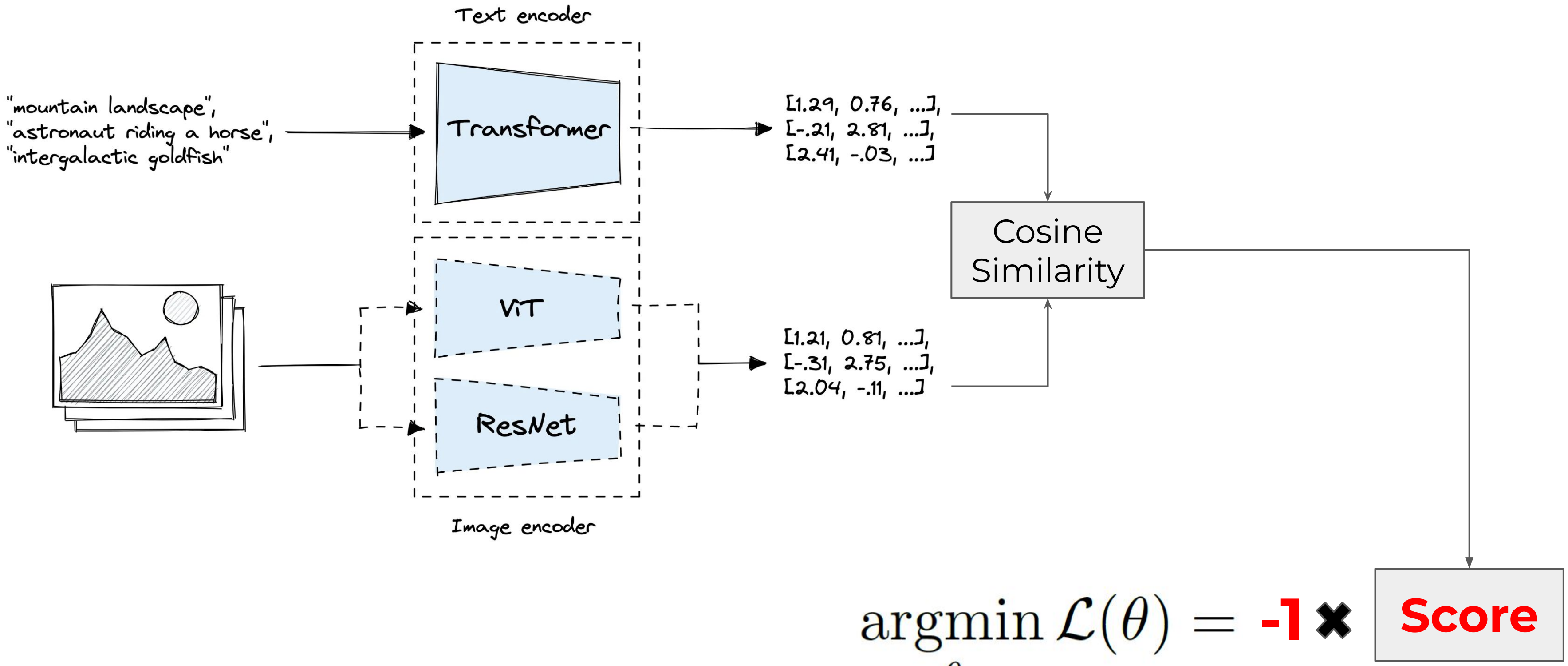
- Why Direct Rendering?
    - Preserves original geometry
    - Avoids reconstruction artifacts
    - Computationally efficient



Neural Highlighter Network



Semantic Loss



$$\underset{\theta}{\operatorname{argmin}} \mathcal{L}(\theta) = -1 \times \text{Score}$$

Experiments : 3D AffordanceNet

🎯 **Objective:** Pipeline evaluation across three experiments

📈**Grid Search:** Systematic exploration of Hyperparameters

🗨️**Prompt Strategies:** Basic, Action, Affordance-Specific

🗂️ 22,949 shapes, 23 classes, 18 affordances

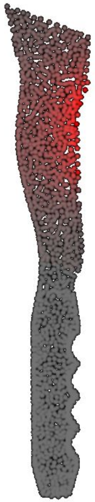
📊 **Validation:** 5 objects, **Test**: 5 objects

🔢 **Input:** 2,048 points per model

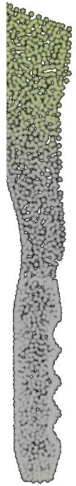
🔍 **Evaluation Metrics:** IoU, AIoU, mean IoU

Experiment 1 : Single Class and Affordance Pair

🎯 **Objective:** Evaluate the **cut** affordance for the **knife** class.



Ground Truth



Predicted

1️⃣ **Basic:** A 3D render of a gray Knife with highlighted cut regions.

2️⃣ **Action:** A 3D render indicating the parts of the gray knife that can be used to cut.

3️⃣ **Affordance Specific:** A 3D render of a gray knife with highlighted regions showing the sharp blade edge and cutting tip, emphasizing the main cutting surface and pointed end.

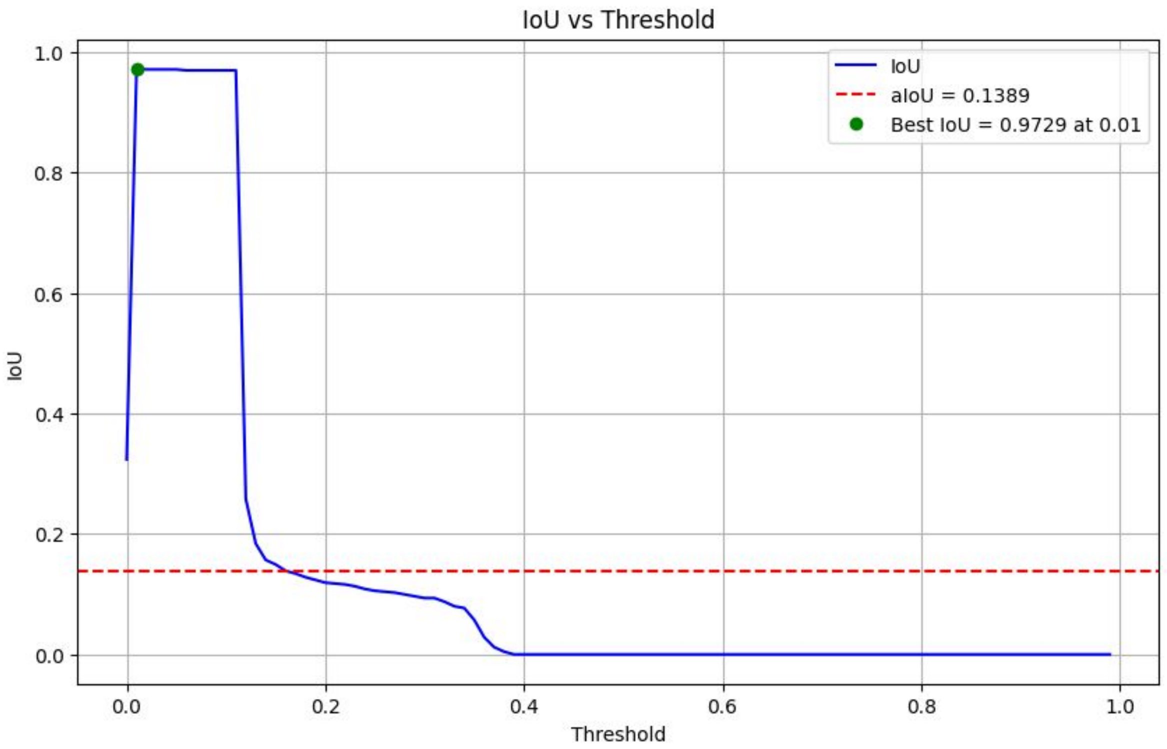
CONFIG 3

Results : Single Class and Affordance Pair

Config	Shape	Prompt Strategy	Thres hold	Learning Rate	Depth	Augment ations	Views	IoU	aIoU
Config 1	d7	Basic	0.01	0.001	4	1	2	0.938	0.109
Config 2	24	Affordance-s pecific	0.94	0.001	4	1	2	0.394	0.391
Config 3	**1e**	**Action**	**0.1**	**0.001**	**4**	**3**	**4**	**0.703**	**0.117**
Config 4	3a	Action	0.02	0.001	4	3	2	0.712	0.044
Config 5	**d7**	**Basic**	**0.1**	**0.001**	**4**	**3**	**2**	**0.972**	**0.1389**

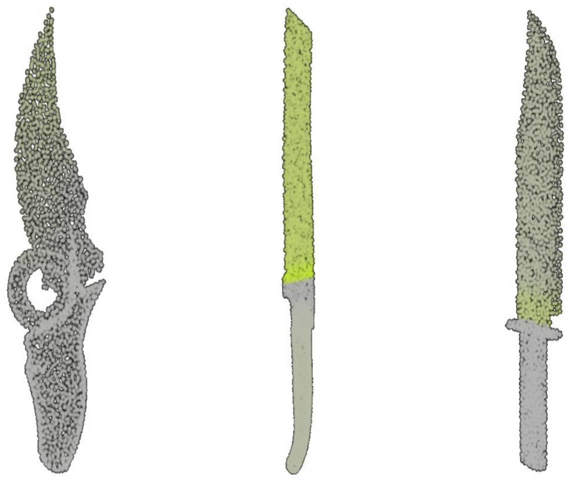
*Configs 3 and 5 were selected as the best-performing configurations based on their IoU, aIoU, and strong alignment with ground truth during visual inspections. This evaluation involved analyzing a total of **240 renders** to identify the top-performing configurations.*

IoU vs. aIoU: Evaluation Metrics



Average IoU computed across a range of thresholds, from 0.0 to 0.99 in 0.01 increments, providing a comprehensive evaluation of segmentation performance.

Test Set Observations: Generalization Performance



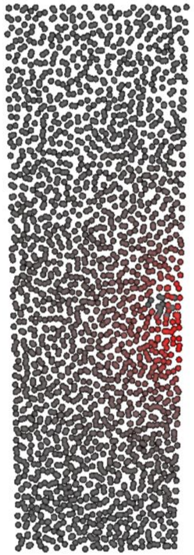
obj2 obj3 obj5

Object	IoU
2	0.4820
3	0.8630
5	0.3534
Mean IoU	**0.339**

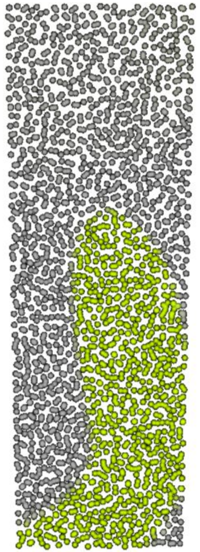
Results for Config 3 (**Action Prompt**) - Knife Class with fixed threshold 0.1

Experiment 2 : Single Class with Multiple Affordances

🎯 **Objective:** Evaluate the model's ability to generalize across multiple affordances (**openable**, **pushable**, **pull**) within the **door** class.



Ground Truth



Predicted

Config 2

1️⃣ **Basic:** *A 3D render of a gray door with highlighted {affordance type} regions.*

2️⃣ **Affordance Specific:**

Openable: *A 3D render of a gray door with highlighted hinge regions and handle areas that enable opening movement.*

Pushable: *A 3D render of a gray door with highlighted flat surface regions designed for pushing.*

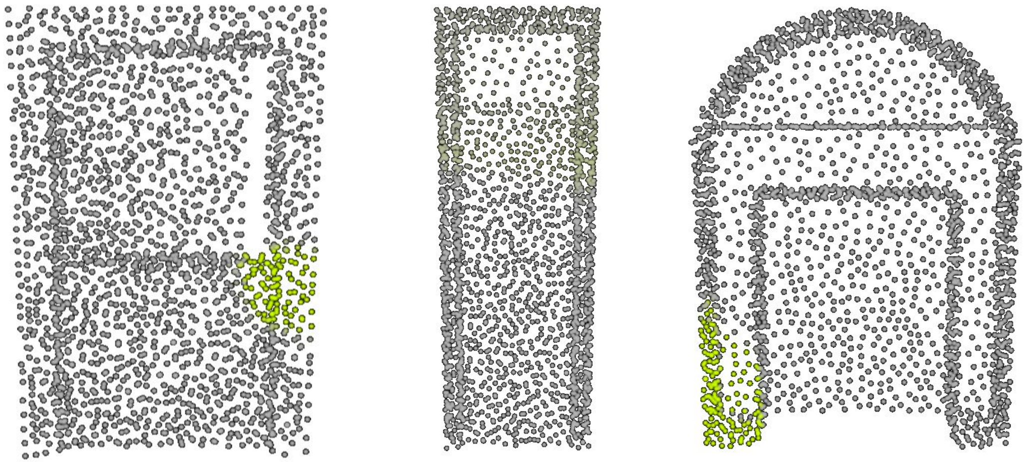
Pull: *A 3D render of a gray door with highlighted regions showing handles, grip spots, or edges used for pulling.*

Results : Single Class with Multiple Affordances

Config	Affordance	Prompt Strategy	Thres hold	Learning Rate	Depth	Augment ations	Views	IoU	aIoU
Config 1	Pushable	Basic	0.3	0.0001	5	3	2	0.215	0.1623
Config 2	Openable	Basic	0.1	0.0001	5	3	2	0.6694	0.3837
Config 3	Pull	Basic	0.1	0.001	4	3	2	0.6670	0.6619

The best configurations for each affordance were selected based on IoU, aIoU, and visual inspections. After analyzing 640 renders, the overall poor performance led us to test all three configurations further.

Test Set Observations: Generalization Performance

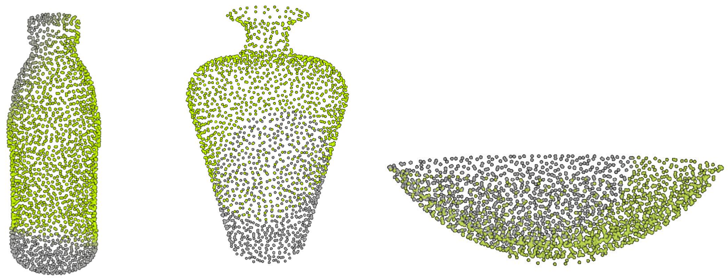


Config	Mean IOU
Config Pushable	0.0930
Config Openable	0.0270
Config Pull	0

Visual results for the pushable, openable, and pull affordances in the test set. Predictions were often misaligned or incomplete.

Experiment 3: Generalization of single affordance over multiple Classes

🎯 **Objective:** Evaluate the model's ability to generalize a single affordance **contain** across diverse object classes **vase**, **bowl** and **bottle.**



Bottle Vase Bowl

1️⃣ **Basic:** A 3D render of a gray {shape class} with highlighted contain regions.

2️⃣ **Action:** A 3D render indicating the parts of the gray {shape class} that can be used to contain.

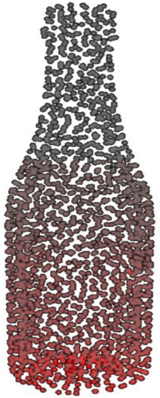
3️⃣ **Affordance Specific:** A 3D render of a gray {shape class} with highlighted regions showing the sharp blade edge and cutting tip, emphasizing the main cutting surface and pointed end.

Results : Generalization of single affordance over multiple Classes

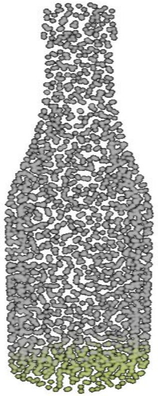
Shape class	Prompt Strategy	Threshold	Learning Rate	Depth	Augmentations	Views	IoU	aIoU
Bottle	**Affordance specific**	**0.1**	**0.001**	**4**	**3**	**2**	**0.9043**	**0.6939**
Vase	Affordance specific	0.1	0.001	4	1	3	0.652	0.648
Bowl	Affordance specific	0.1	0.001	5	1	3	0.9155	0.295

The bottle configuration for the affordance contain was selected as the best-performing configuration based on their IoU, aIoU, and strong alignment with ground truth during visual inspections. This evaluation involved analyzing a total of ***412 renders*** *to identify the top-performing configurations.*

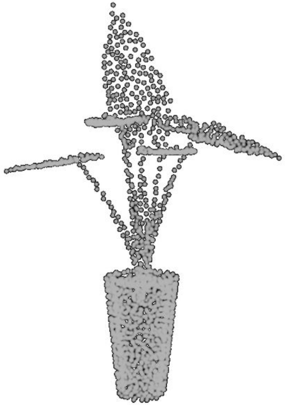
Test Set Observations: Generalization Performance



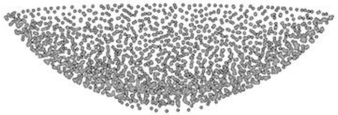
Ground Truth



Predicted



IoU= 0.345

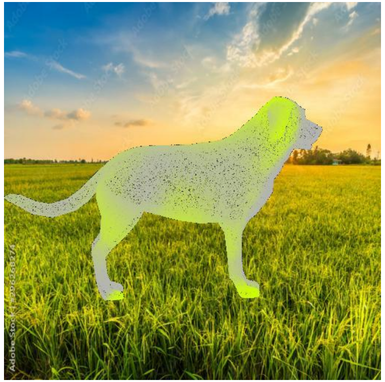


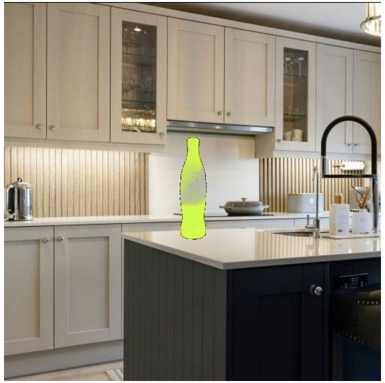
no meaningful affordance detection

Extension of the pipeline: Adding background and augmentation

Background
- No Background
- Outdoor
- Indoor

Augmentation
- Default Transform
- Viewpoint Transform
- Lighting Transform
- Balanced Transform





Augmentation types and backgrounds

Augmentation Type	Background	mIoU
Balanced	**No Background**	**0.4924**
	Outdoor 1	0.3011
	Outdoor 2	0.3430
	Indoor 1	0.1657
	Indoor 2	0.1861
Viewpoint	No Background	0.5360
	Outdoor 1	0.3330
	Outdoor 2	0.4790
	Indoor 1	**0.6639**
	Indoor 2	0.3654

Augmentation Type	Background	mIoU
Lighting	**No Background**	**0.5881**
	Outdoor 1	0.3342
	Outdoor 2	0.5799
	Indoor 1	0.4261
	Indoor 2	0.0669
Default	**No Background**	**0.6790**
	Outdoor 1	0.3263
	Outdoor 2	0.4708
	Indoor 1	0.3851
	Indoor 2	0.1921

Comparison of augmentation strategies and background types tested on object class knife with affordance cut using config 5 over 3 shapes by mIoU.

LiDAR

LiDAR

KIRI Engine Application

A 3D render indicating the parts of the gray **chair** that can be used to **sit**.

Alternative Backbones

- **OpenCLIP: Poor Results**
- **OpenShape: Resource Intensive**

open_clip

An open source implementation of CLIP.

Conclusion

3D Highlighter architecture shows promise for label-free affordance detection.

Performance depends heavily on hyperparameter tuning and effective prompt engineering.

Future work should focus on improving generalization and using alternative backbones

Thank you

References

[1] Wenzheng Chen, Huan Ling, Jun Gao, Edward Smith, Jaakko Lehtinen, Alec Jacobson, and Sanja Fidler. Learning to predict 3d objects with an interpolation-based differentiable renderer. In Advances in Neural Information Processing Systems(NeurIPS), volume 32, 2019. 1

[2] Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws for contrastive language-image learning. In 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), page 2818–2829. IEEE, June 2023. 8

[3] Dale Decatur, Itai Lang, and Rana Hanocka. 3d highlighter: Localizing regions on 3d shapes via text descriptions. In 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 20930–20939, 2023. 1

[4] Sheng Deng, Xun Xu, Chaozheng Wu, Ke Chen, and Kui Jia. 3d affordancenet: A benchmark for visual object affordance understanding. In 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 1778–1787, 2021. 1

[5] Minghua Liu, Ruoxi Shi, Kaiming Kuang, Yinhao Zhu, Xuanlin Li, Shizhong Han, Hong Cai, Fatih Porikli, and Hao Su.= Openshape: Scaling up 3d shape representation towards open world understanding, 2023. 8

[6] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. arXiv preprint arXiv:2103.00020, 2021. 1

[7] Yiheng Xie, Towaki Takikawa, Shunsuke Saito, Or Litany, Shiqin Yan, Numair Khan, Federico Tombari, James Tomp- kin, Vincent Sitzmann, and Srinath Sridhar. Neural fields in visual computing and beyond. Computer Graphics Forum, 41(1):1–12, 2022. 1

[8] Yuhang Yang, Wei Zhai, Hongchen Luo, Yang Cao, Jiebo Luo, and Zheng-Jun Zha. Grounding 3d object affordance from 2d interactions in images, 2023.