Exploring Correspondences Between Gibsonian and Telic Affordances for Object Grasping

Aniket Tomar and Nikhil Krishnaswamy

AREA-2 Workshop, August 9, 2022, Galway, Ireland

Colorado State University

Outline

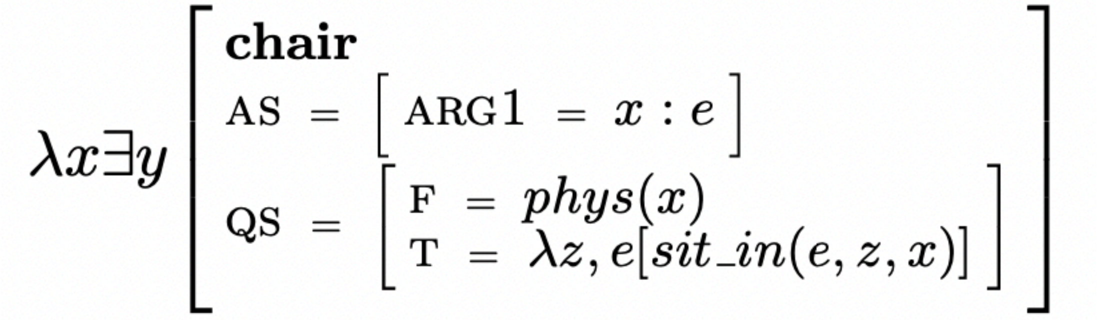
- Telic vs. Gibsonian Affordances
- MeshCNN
- Dataset and Annotation
- Model Performance
- Interchangeability Between Embedding Spaces
- Discussion

Gibson: Affordances

- *Affordance* (n.) - the functional and ecological relationship between organisms and their environments (Gibson, 1977)
  - To say an object “affords” and action is to say an object facilitates the action being taken with it
- Predicting affordances has proven to be a topic of interest to AI, esp. robotics
- HOI datasets for image and video recognition provide a similar task to evaluate against (Chao et al., 2015; Goyal et al., 2017; Chao et al., 2018)
- Methods to learn affordances from visual features exist (Fang et al., 2018; Nagarajan et al., 2019; Xiao et al., 2019)
- Affordance prediction is still challenging
  - Humans learn about affordances by using objects or watching them being used, rather then from text (e.g., “cups contain things,” “spoons are used for stirring)
  - Such information is considered obvious and may be sparse in corpora

Gibsonian vs. Telic Affordances

- **Gibsonian** affordances: behaviors afforded due to object structure
  - Grasp-able, hold-able, pick up-able, throw-able, etc.
- Pustejovsky (2013) introduced the notion of the **telic** affordance, or behavior conventionalized due to object’s typical use or purpose
  - Downstream of the GL telic role

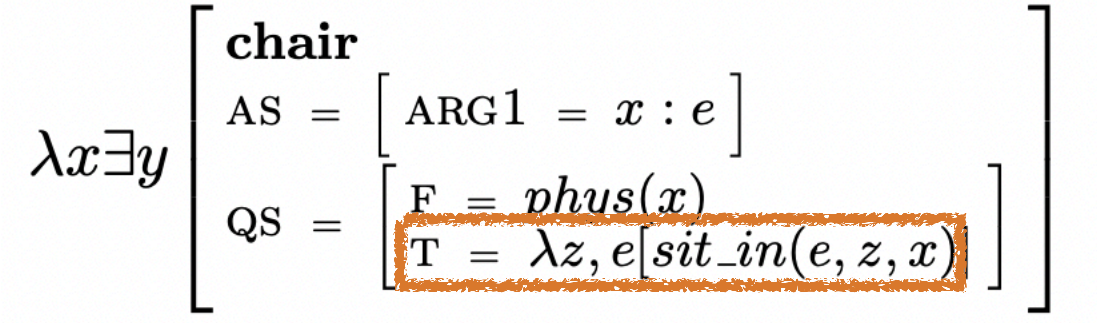


$$\lambda x \exists y \begin{bmatrix} \textbf{chair} \\ \text{AS} = \begin{bmatrix} \text{ARG1} = x : e \end{bmatrix} \\ \text{QS} = \begin{bmatrix} \text{F} = phys(x) \\ \text{T} = \lambda z, e[sit\_in(e, z, x)] \end{bmatrix} \end{bmatrix}$$

  - Gibsonianly, a chair might be grasp-able, slide-able, etc.
    It was *made* to be sit in-able

Gibsonian vs. Telic Affordances

- **Gibsonian** affordances: behaviors afforded due to object structure
  - Grasp-able, hold-able, pick up-able, throw-able, etc.
- Pustejovsky (2013) introduced the notion of the **telic** affordance, or behavior conventionalized due to object’s typical use or purpose
  - Downstream of the GL telic role

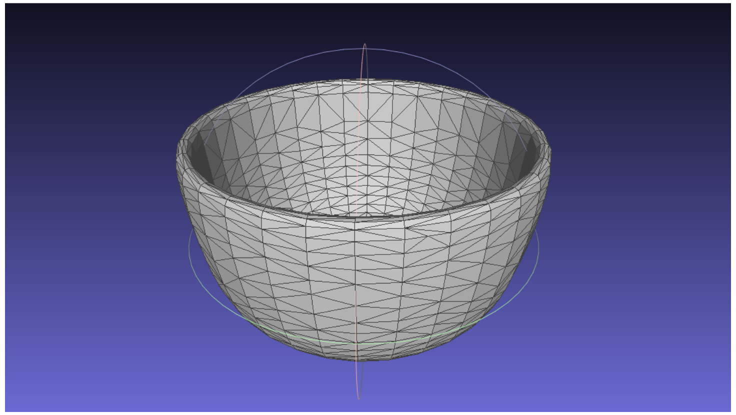


$$\lambda x \exists y \begin{bmatrix} \textbf{chair} & & \\ \text{AS} & = & \begin{bmatrix} \text{ARG}1 \;=\; x : e \end{bmatrix} \\ \text{QS} & = & \begin{bmatrix} \text{F} \;=\; phys(x) \\ \boxed{\text{T} \;=\; \lambda z, e[sit\_in(e, z, x)]} \end{bmatrix} \end{bmatrix}$$

  - Gibsonianly, a chair might be grasp-able, slide-able, etc.
    It was *made* to be sit in-able

Gibsonian vs. Telic Affordances

- If Gibsonian and telic affordances are different, are there nonetheless direct correspondences between them that can be automatically exposed?
- We explore the ability of machine learning methods to learn correspondences between objects based on their geometries, and based on human annotations of object use
    - **Focus on one behavior only: grasping**
- We then train classifiers on two datasets representing the different affordance types and use an affine transformation technique explore correspondences between the embedding spaces of the two classifiers

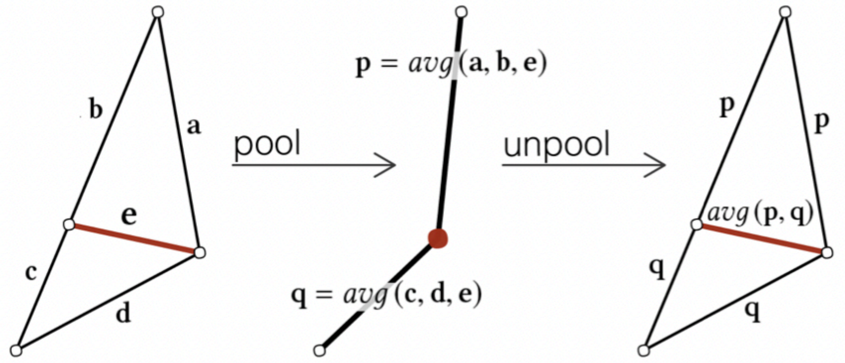


MeshCNN

- Key to this work is MeshCNN (Hanocka et al., 2019)
  - Adaptation of convolutional neural networks for the analysis of 3D triangular meshes
  - Though polygonal meshes explicitly and efficiently capture both surface and topology, few attempts have been made to use such data for ML inference
  - High-quality 3D polygonal mesh data is difficult to acquire at scale
  - Advances in capturing, synthesizing, or reconstructing 3D mesh or point cloud data (e.g., Yu et al., 2020; Jain et al., 2021; Lin et al., 2021; Ye et al., 2021; Michel et al., 2021; Hanocka et al., 2020a; Metzer et al., 2021) suggest that such data is useful for AI tasks

MeshCNN

- Convolution operators apply to unambiguously-ordered features in a neighborhood
- Because images are pixel grids, this creates an inherent neighborhood and ordering
  - Not the case in 3D triangular meshes
- MeshCNN authors define a neighborhood for each edge that consists of:
  - Edges contained in faces incident on that edge
  - Vertices ordered counter clockwise
  - Input features of the edge: dihedral angle, two inner angles, two edge-length ratios between edge and perpendiculars for each face from the edge
- Enabled custom pooling operation that collapses incident edges to a point
- MeshCNN has demonstrated performance on mesh segmentation and classification, using fewer parameters and compute time than comparable methods

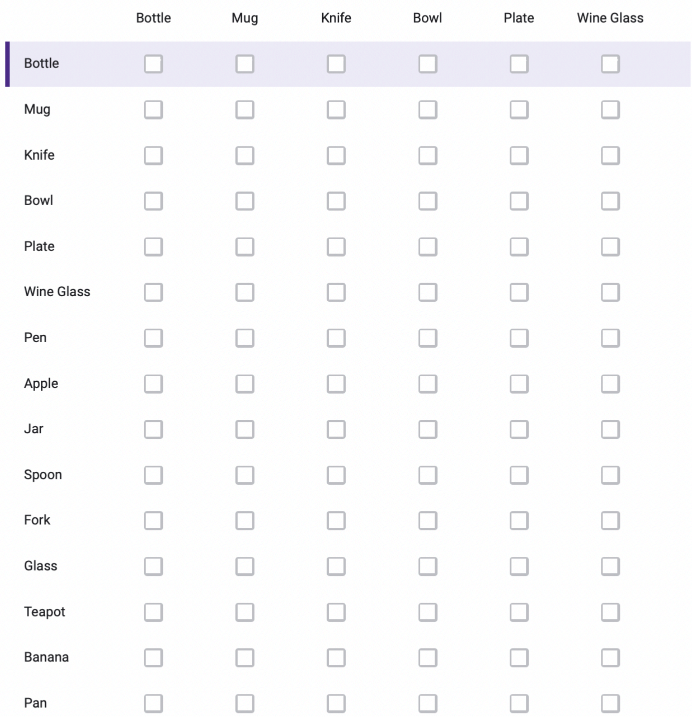


Affordance Classification Task

- Task: If MeshCNN can classify meshes by type, can it classify groups of meshes by type (e.g., grasp method)?
- Select test set of common household object graspable with one hand:
  - *bottle, mug, knife, bowl, plate, wine glass, pen, apple, jar, spoon, fork, glass, teapot, banana, pan*
  - Common kitchen objects represent a common problem set in this domain (e.g., Damen et al., 2018)
- Similarities in the mesh should indicate similarities in the grasp method
  - e.g., cylindrical objects grasped similarly, objects with handles grasped similarly

Human Annotation

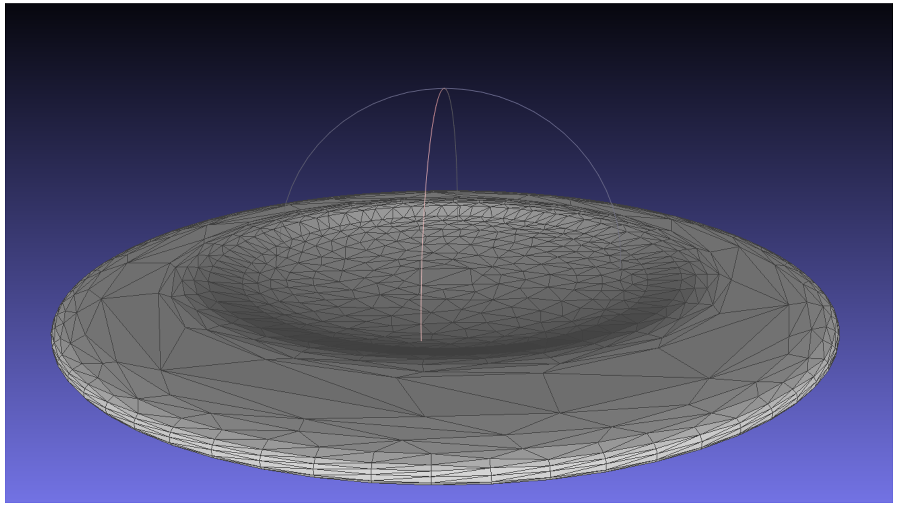
- Created a survey to elicit canonical grasp pose for each object
- Multiple-choice questionnaire: “*Consider how your hand is posed while grasping each object for typical use. Then, for each object, select all other objects which are grasped using a similar hand pose.*”
- Participants could select up to 15 objects, to group multiple objects in grasp classes
  - 28 participants (∴ 28 15D k-hot vectors)
  - Responses filtered using standard statistical techniques like Z-score filtering and normalization
- Computed Kraemer’s kappa (Kraemer, 1980) to account for multiple annotators and of choices
- κ ≈ .32, indicating “fair agreement” (Landis and Koch, 1977)



	Bottle	Mug	Knife	Bowl	Plate	Wine Glass
Bottle	□	□	□	□	□	□
Mug	□	□	□	□	□	□
Knife	□	□	□	□	□	□
Bowl	□	□	□	□	□	□
Plate	□	□	□	□	□	□
Wine Glass	□	□	□	□	□	□
Pen	□	□	□	□	□	□
Apple	□	□	□	□	□	□
Jar	□	□	□	□	□	□
Spoon	□	□	□	□	□	□
Fork	□	□	□	□	□	□
Glass	□	□	□	□	□	□
Teapot	□	□	□	□	□	□
Banana	□	□	□	□	□	□
Pan	□	□	□	□	□	□

Mesh Dataset

- For each object, collected 40 3D meshes from public repositories
- Converted all to .obj format, standardized to ~8,000 faces, ~15,000 edges each
- Filter meshes to remove anomalies, e.g., non-manifold or non-Euclidean meshes
- Visual inspected each mesh for identity
- Validated each mesh with MeshCNN itself on dummy classification task



Deriving Grasp Classes

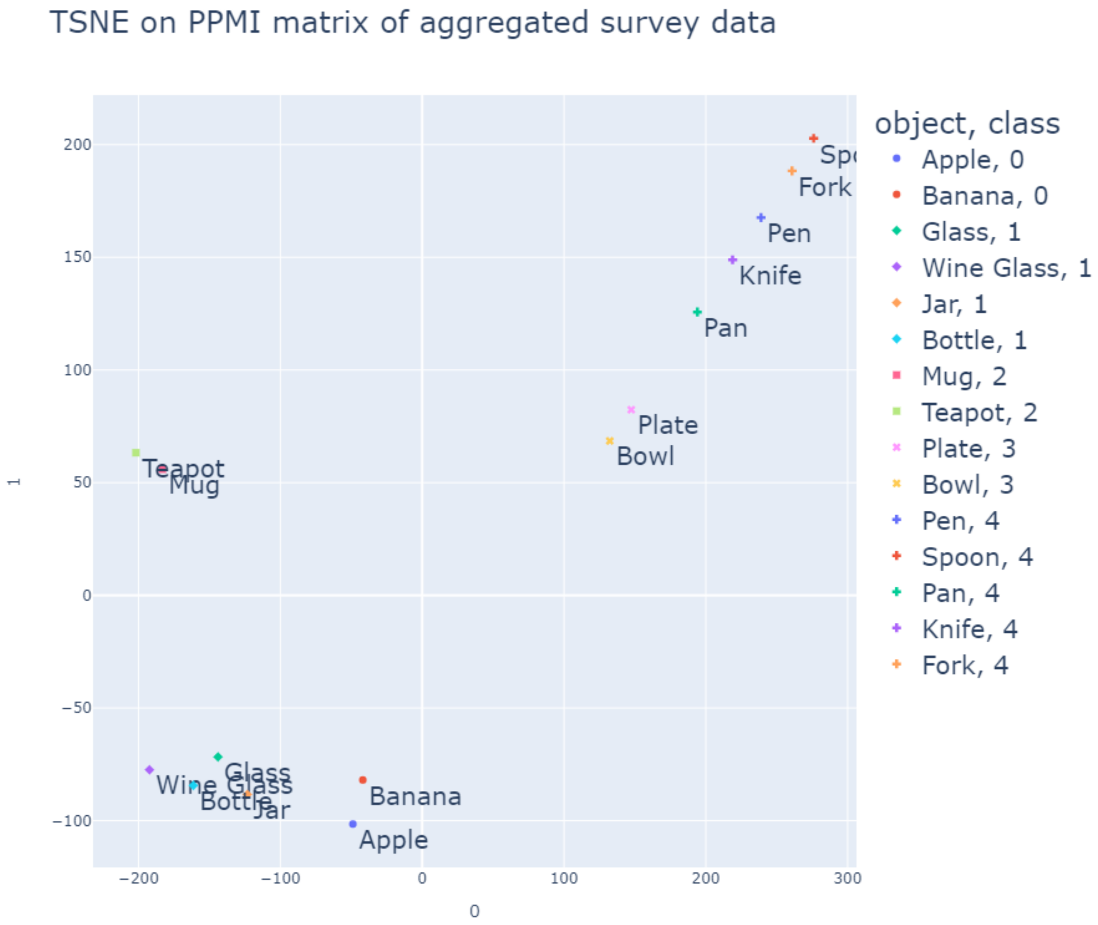
- Sum 28 15D vectors for each object, derived from the human annotation, into cooccurrence matrix
- For each object (i.e., row), compute PPMI with each other object (i.e., column), given by

$$PPMI(a,b) = max\left(ln\left(\frac{P(a,b)}{P(a)P(b)} \right), 0 \right)$$

- Normalize cooccurrence matrix by PPMI values, then use Euclidean distance between object vectors as a similarity metric to derive grasp classes

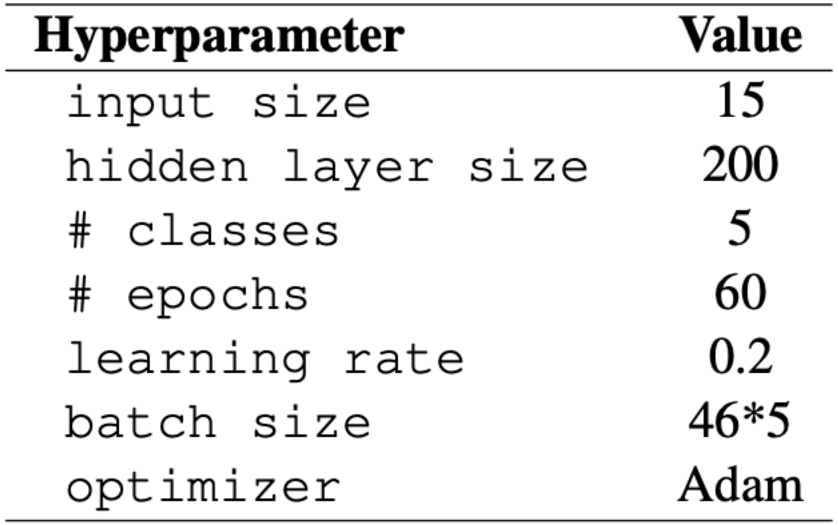
Grasp Class	Objects
spherical	apple, banana
cylindrical	bottle, wine glass, glass, jar
hook	mug, teapot
palmar pinch	bowl, plate
tripod	spoon, fork, knife, pen, pan

Deriving Grasp Classes

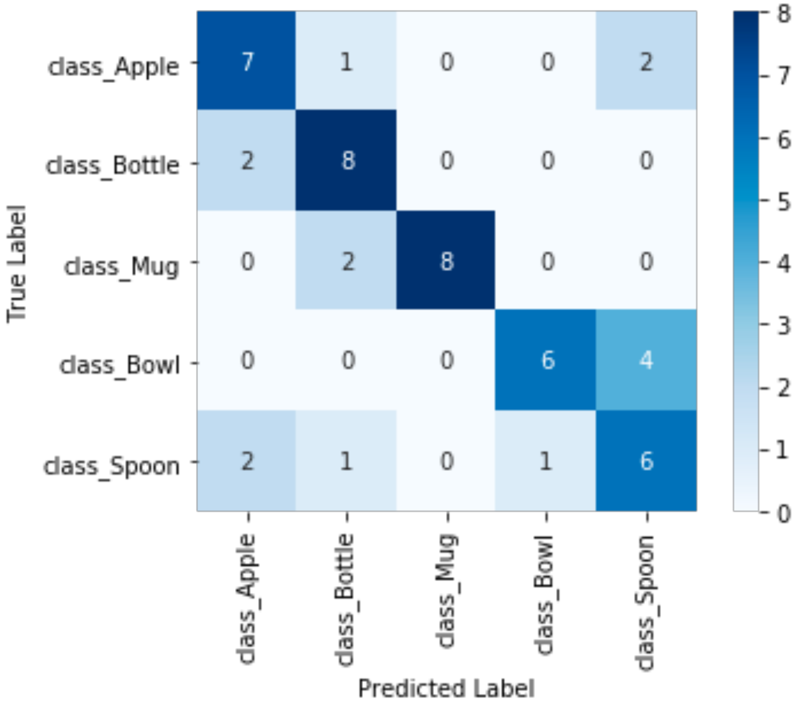


Classifying Human Annotations

- Trained MLP classifier to predict grasp class from “human data” PPMI vectors
- 46 samples per class (train), 10 samples per class (test)
- **Overall 72% test accuracy, 70.6 test F1**



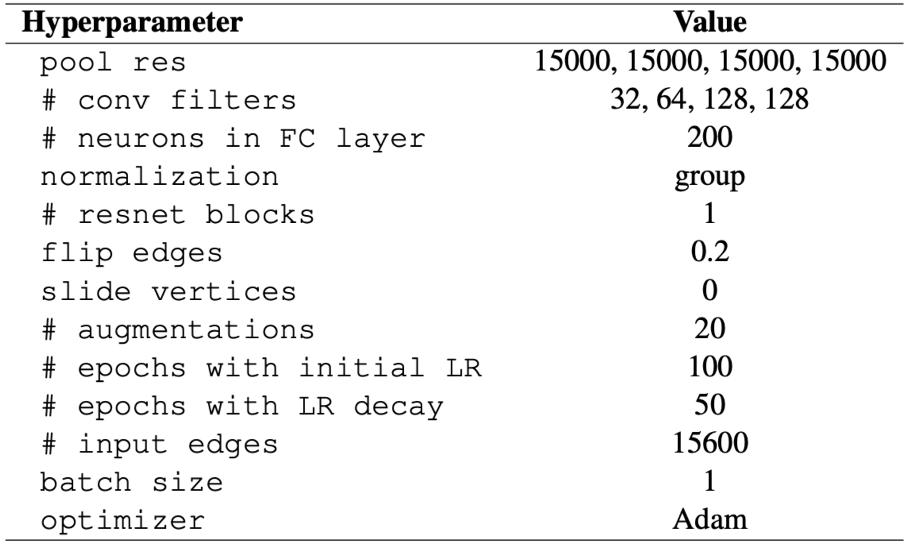
Hyperparameter	Value
input size	15
hidden layer size	200
# classes	5
# epochs	60
learning rate	0.2
batch size	46*5
optimizer	Adam



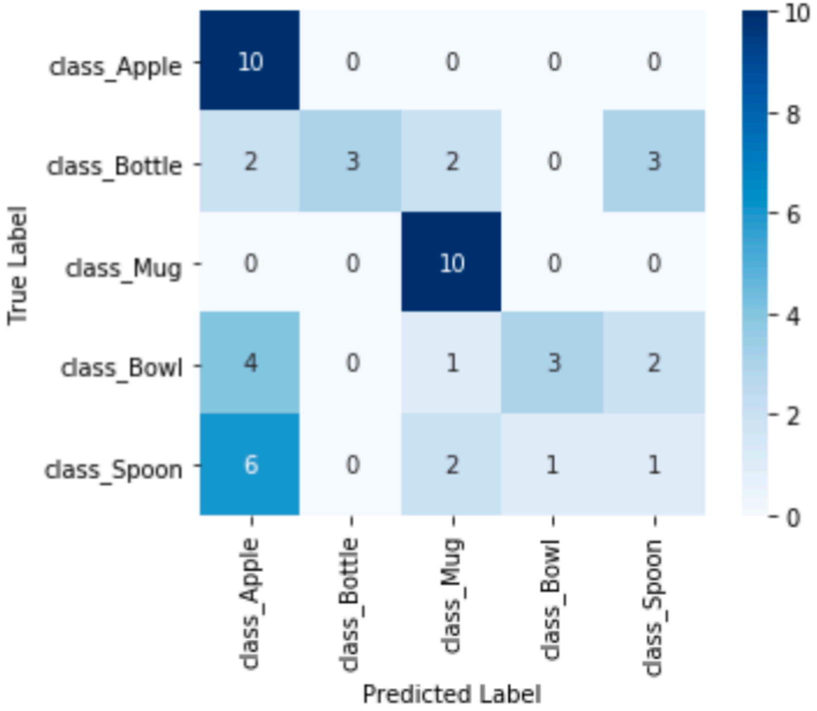
	class_Apple	class_Bottle	class_Mug	class_Bowl	class_Spoon
class_Apple	7	1	0	0	2
class_Bottle	2	8	0	0	0
class_Mug	0	2	8	0	0
class_Bowl	0	0	0	6	4
class_Spoon	2	1	0	1	6

Classifying Meshes

- Trained MeshCNN instance to predict grasp class mesh data
- 70 samples per class (train), 10 samples per class (test)
- **Overall 54% test accuracy, 46.9 test F1**



Hyperparameter	**Value**
pool res	15000, 15000, 15000, 15000
# conv filters	32, 64, 128, 128
# neurons in FC layer	200
normalization	group
# resnet blocks	1
flip edges	0.2
slide vertices	0
# augmentations	20
# epochs with initial LR	100
# epochs with LR decay	50
# input edges	15600
batch size	1
optimizer	Adam



Correpondences Between Classifier Embedding Spaces

- Poor MeshCNN test performance is curious!
- MeshCNN has advertised effectiveness on related tasks, such as classifying if an object has a handle (Hanocka et al., 2019)
- What makes the grasp classification task difficult?

Correpondences Between Classifier Embedding Spaces

- Poor MeshCNN test performance is curious!
- MeshCNN has advertised effectiveness on related tasks, such as classifying if an object has a handle (Hanocka et al., 2019)
- What makes the grasp classification task difficult?

Correpondences Between Classifier Embedding Spaces

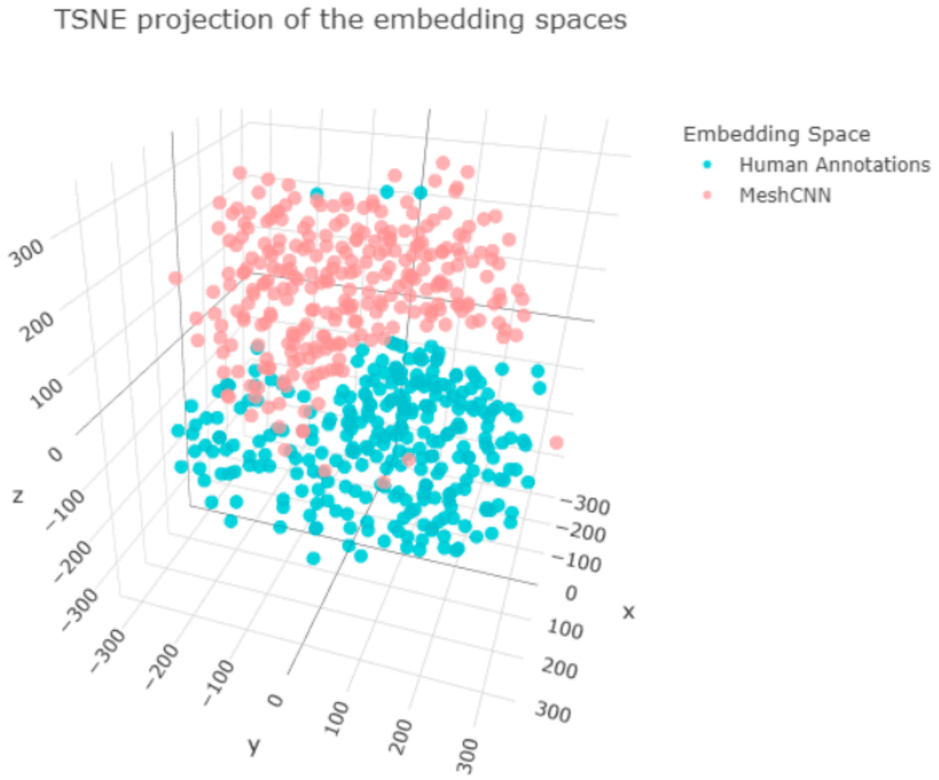
- Previous research from the computer vision community (e.g., McNeely-White et al., 2020) has demonstrated that:
  - In closed-set tasks with fixed label sets, interchangeability between network embedding spaces is expected
  - Up to a matrix $M_{A \to B} \in \mathbb{R}^{d_A} \times \mathbb{R}^{d_B}$ that transforms inputs in $\mathbb{R}^{d_A}$ to outputs in $\mathbb{R}^{d_B}$ if the inputs and outputs correspond to the same label
  - Asking, given two sets of objects (vectors) A and B, are they the "same" under an affine transformation?
- If our grasp classes can be represented as equivalent subspaces in both the MeshCNN and MLP classifiers…
  - Hypothesis: if MeshCNN embeddings can be transformed to MLP embedding space with sufficiently high R^2, poor performance likely due to overfitting (as the trained model spaces are roughly equivalent)
  - If not, the underlying representations are fundamentally different

Correpondences Between Classifier Embedding Spaces

- **Process:**
- Retrieve 200D embeddings for all inputs (train and test) from each classifier
- MeshCNN embeddings as inputs, MLP embeddings as outputs
- Pair embeddings according to grasp class: map MeshCNN embeddings into MLP embedding space
- Use MLP regressor to minimize distance between paired embeddings
- Resulting matrix attempts to align the two embedding spaces as closely as possible according to information in embedding pairings
- **We are transforming the MeshCNN embeddings into MLP embedding space to make them directly comparable**

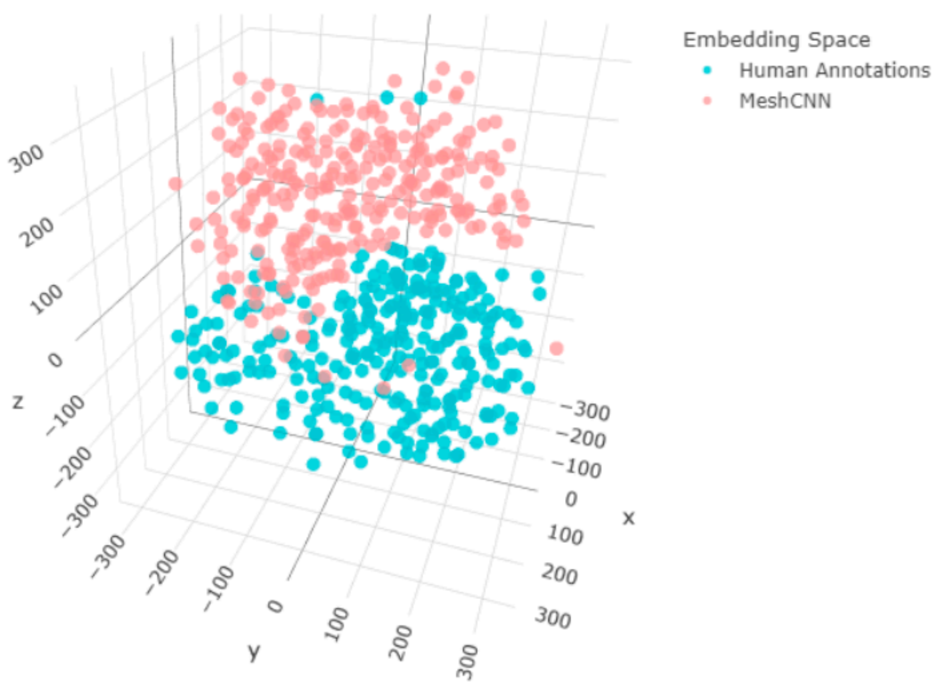
Correpondences Between Classifier Embedding Spaces

- **Results:**
- Train embeddings R^2: 0.06
- Test embeddings R^2: 0.02
- After mapping, classifier embedding spaces remain largely non-overlapping
  - Two classifiers learned fundamentally different representations
  - Poor MeshCNN performance likely not due to overfitting on training data
  - Mesh information doesn’t include effective indicators of grasp-relevant groupings (except in certain cases—more later!)



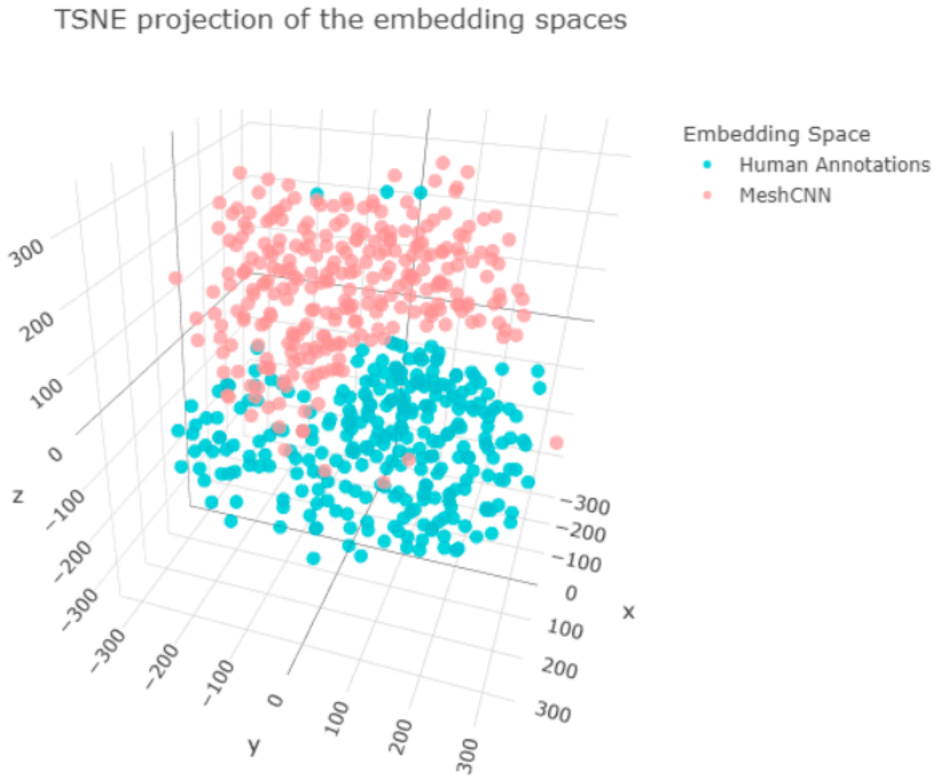
Gibsonian vs. Telic Information

- Two almost completely separate regions
- If MeshCNN had overfitted to training data, MeshCNN training embeddings (pink) should map much more closely to equivalent MLP embeddings (blue)
- Instead, convex hull of MLP embeddings encloses mapped MeshCNN embeddings
- Most embeddings from one space neatly separated from embeddings from the other space
- **Why?**


TSNE projection of the embedding spaces

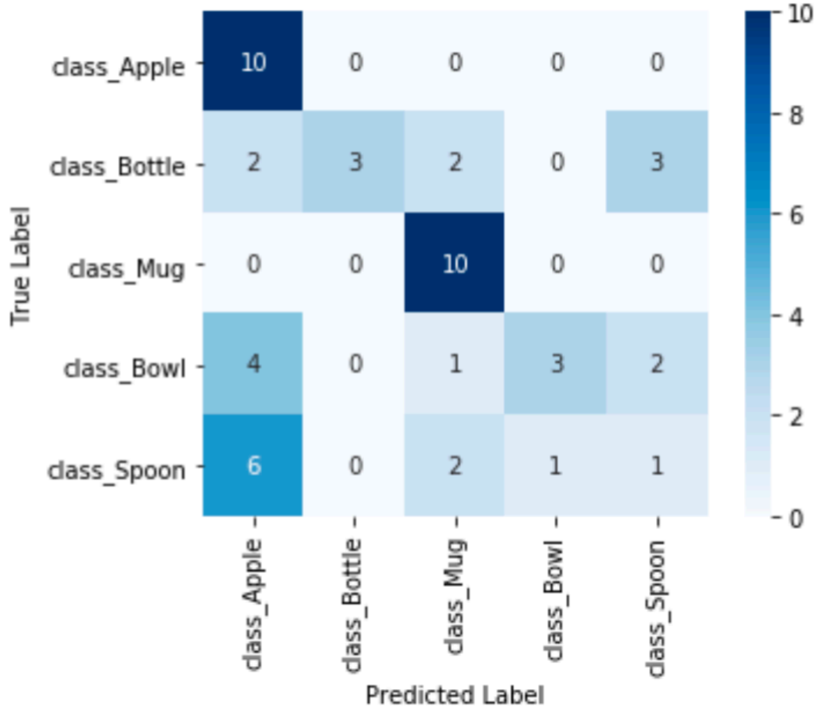
Gibsonian vs. Telic Information

- Recall the prompt for human annotators:
    - *“Consider how your hand is posed while grasping each object for typical use. Then, for each object, select all other objects which are grasped using a similar hand pose.”*
    - **This phrasing, esp. “for typical use” selects for telic affordances**
- 3D meshes alone do not capture this information



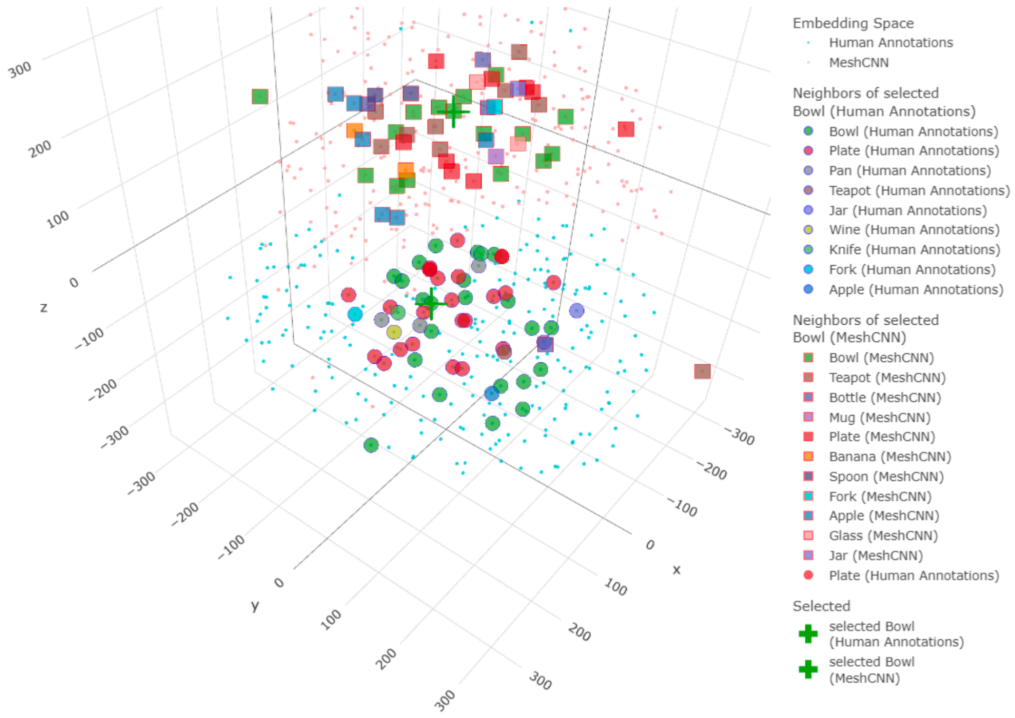
Gibsonian vs. Telic Information

- There may be cases where geometry correspondences to use
- e.g., handles
  - MeshCNN performed well on the “hook” and “spherical” grasp classes
- Results suggest that MeshCNN geometry-level representations correlated with Gibsonian affordances, but class labels came from telic affordances
  - Information not present in the mesh itself to learn Gibsonian-telic correspondences
- MLP trained over human annotation of telic affordances learned different information
- **Gibsonian and telic affordances represent related but fundamentally different ways of interpreting the same objects**



True Label \ Predicted Label	class_Apple	class_Bottle	class_Mug	class_Bowl	class_Spoon
class_Apple	10	0	0	0	0
class_Bottle	2	3	2	0	3
class_Mug	0	0	10	0	0
class_Bowl	4	0	1	3	2
class_Spoon	6	0	2	1	1

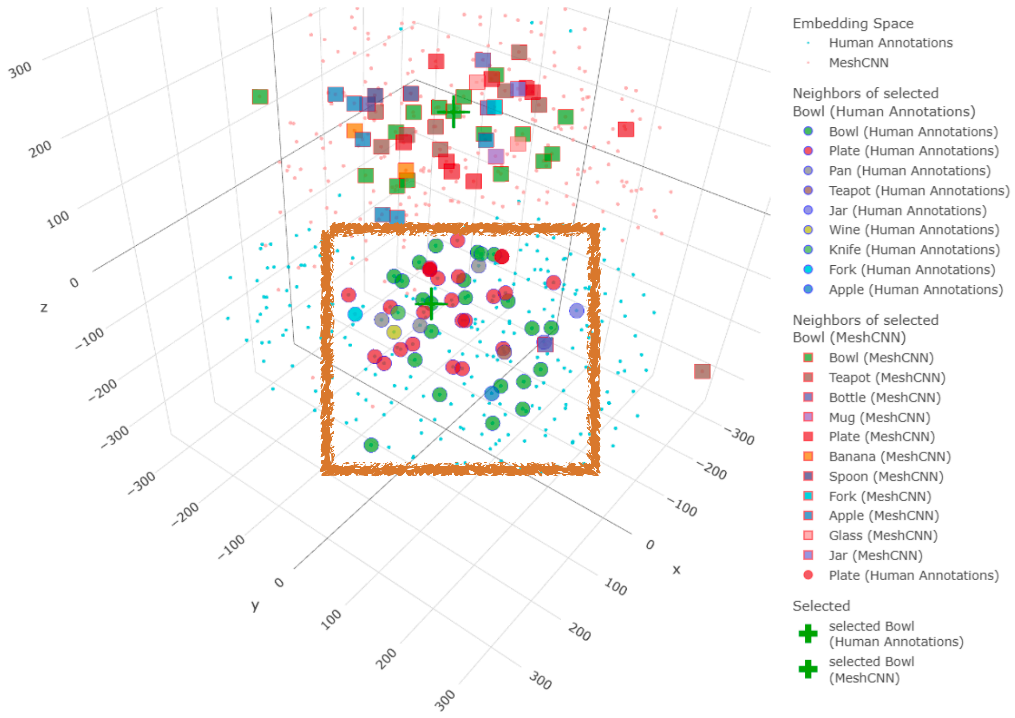
Specific Nearest Neighbors



Neighborhood of a representative "Bowl" object in each of the two original embedding spaces

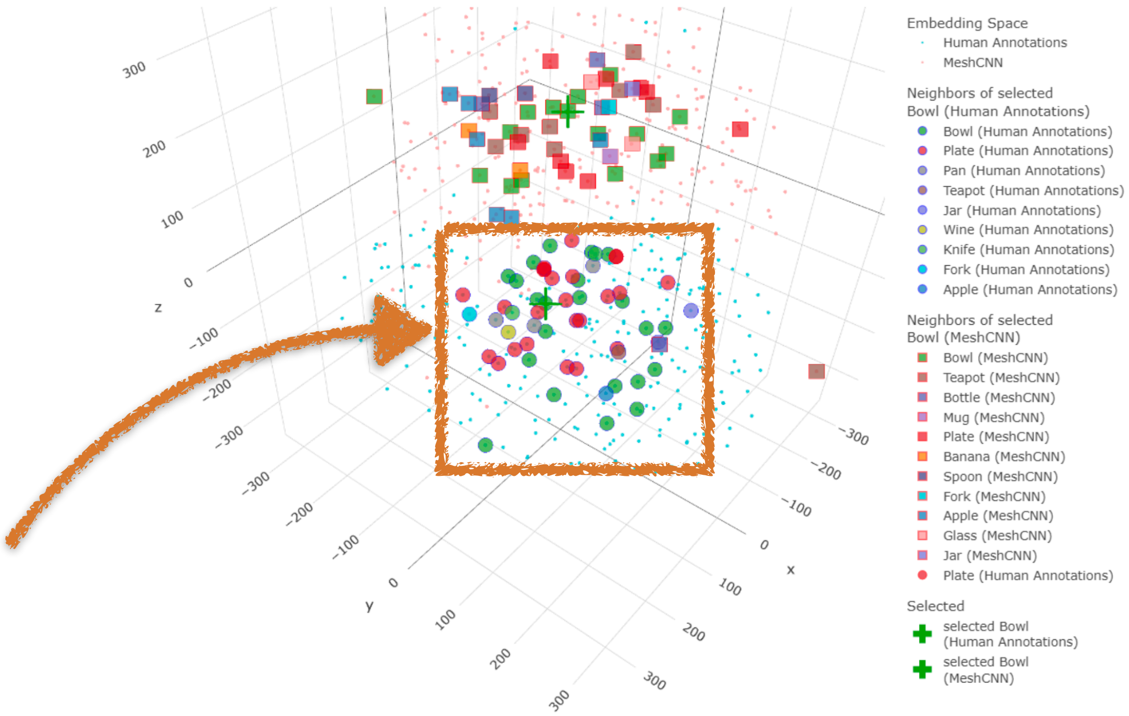
Specific Nearest Neighbors

Neighborhood of a representative "Bowl" object
in each of the two original embedding spaces

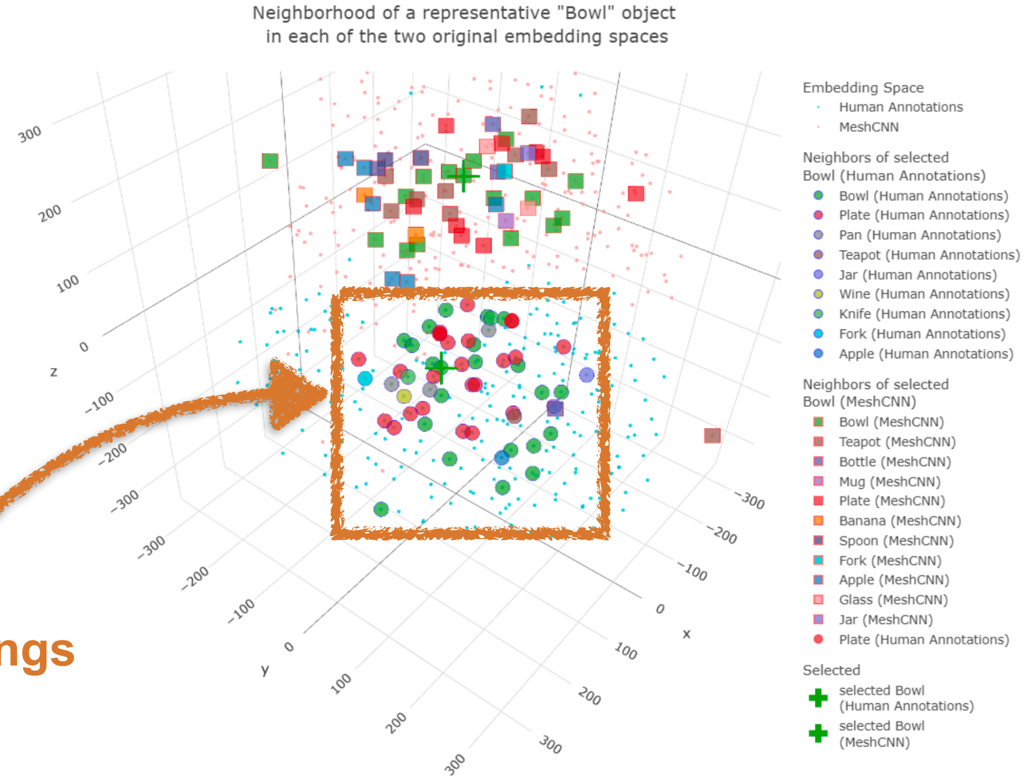


Specific Nearest Neighbors

Neighborhood of a representative "Bowl" object
in each of the two original embedding spaces

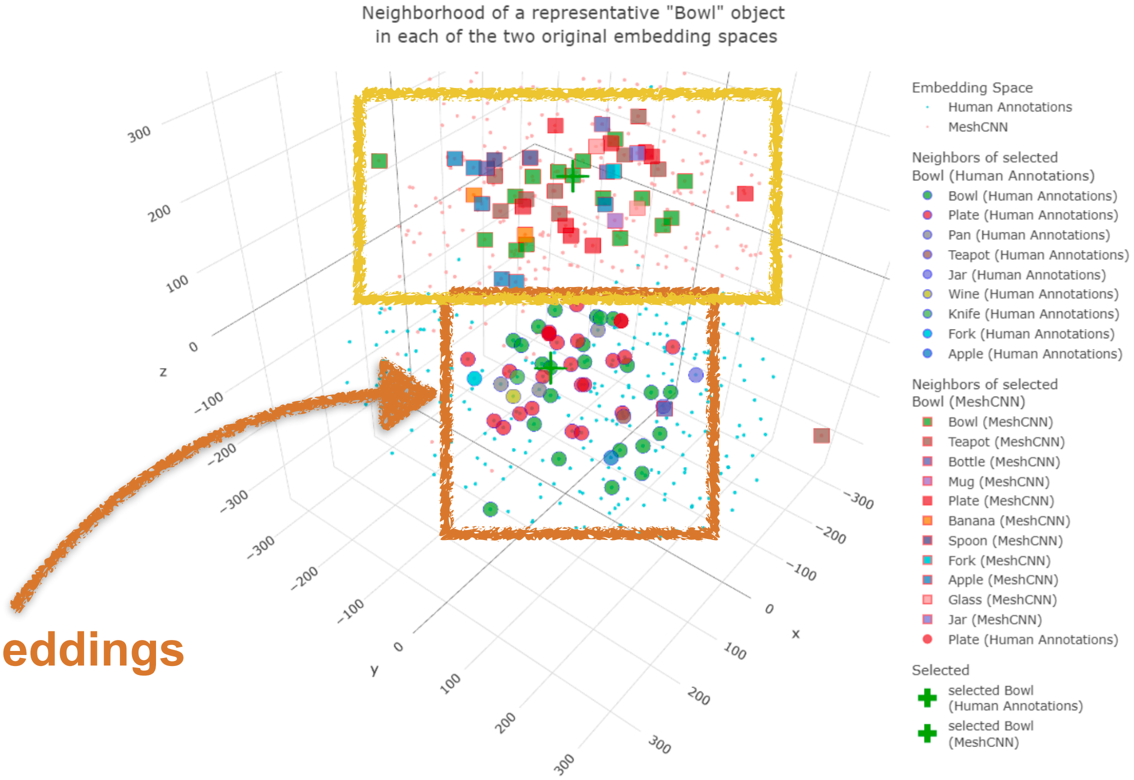


Specific Nearest Neighbors



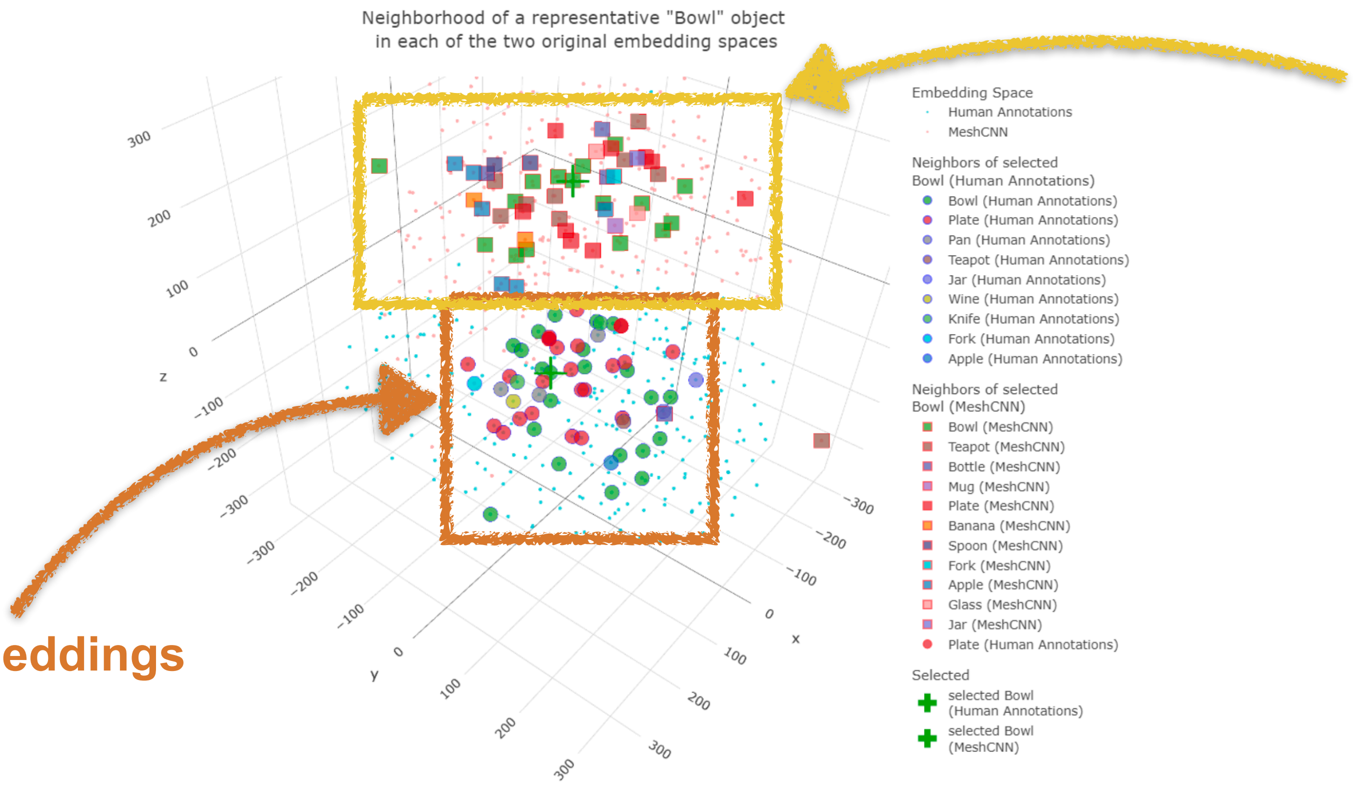
MLP Embeddings

Specific Nearest Neighbors


Neighborhood of a representative "Bowl" object in each of the two original embedding spaces

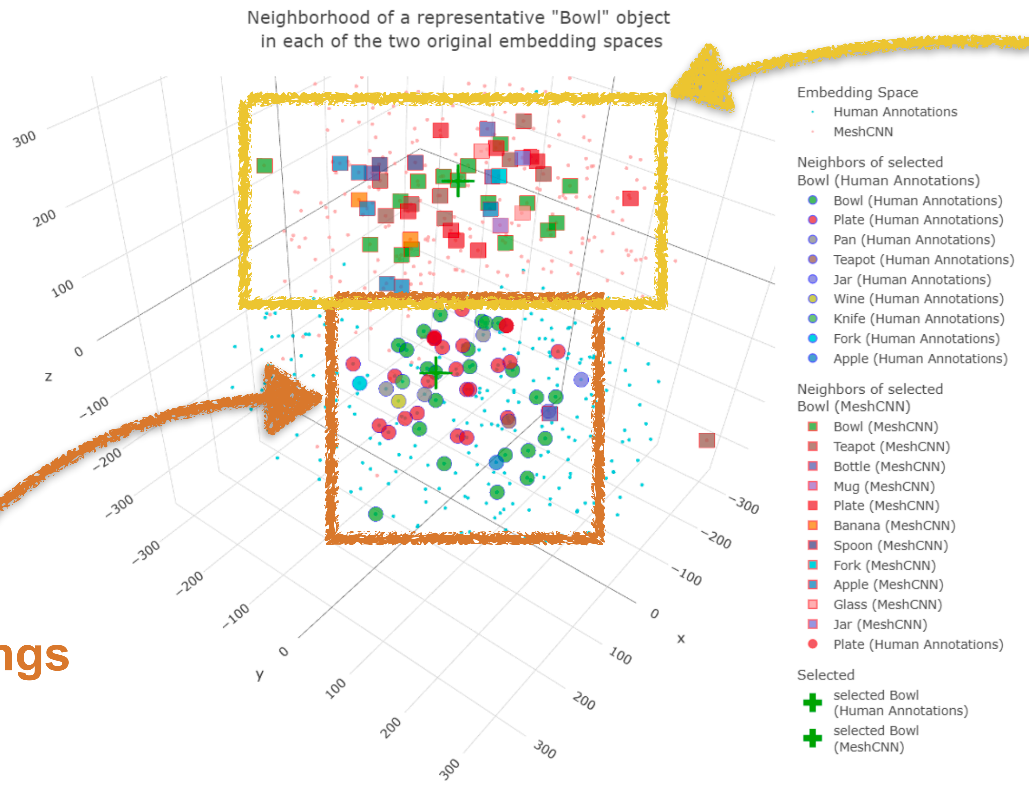
MLP Embeddings

Specific Nearest Neighbors



MLP Embeddings

Specific Nearest Neighbors

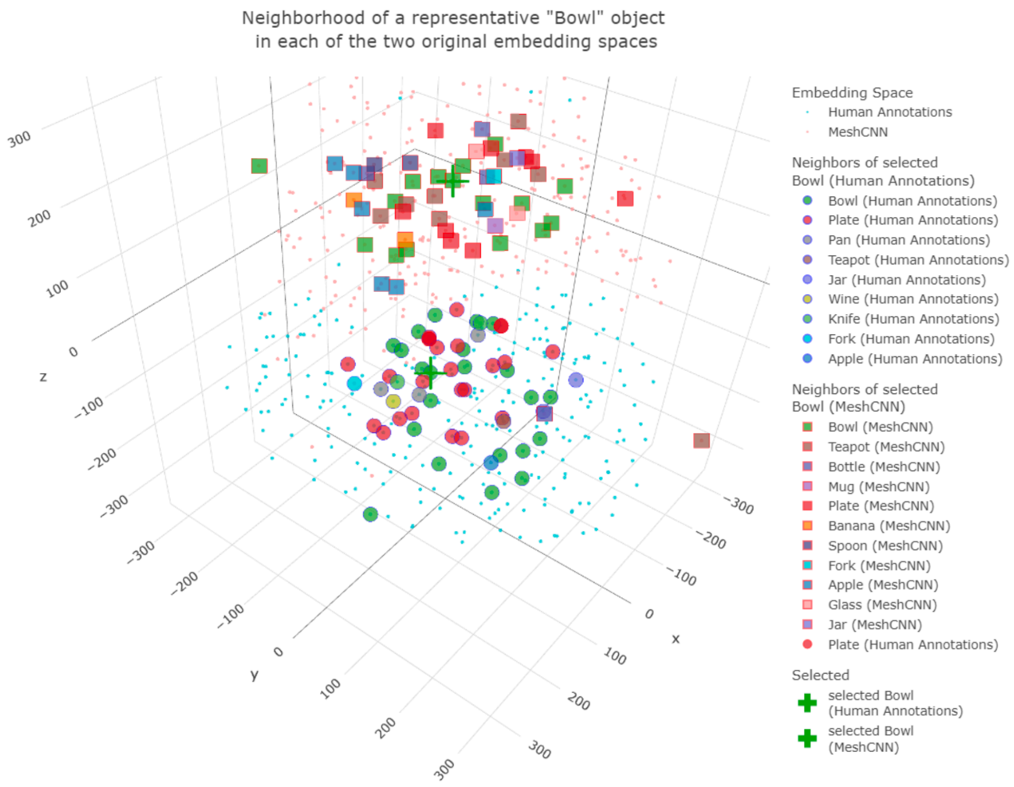


MeshCNN Embeddings

MLP Embeddings

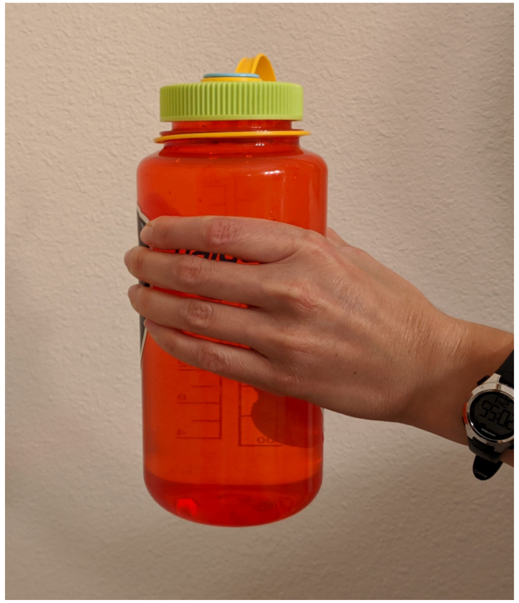
Specific Nearest Neighbors

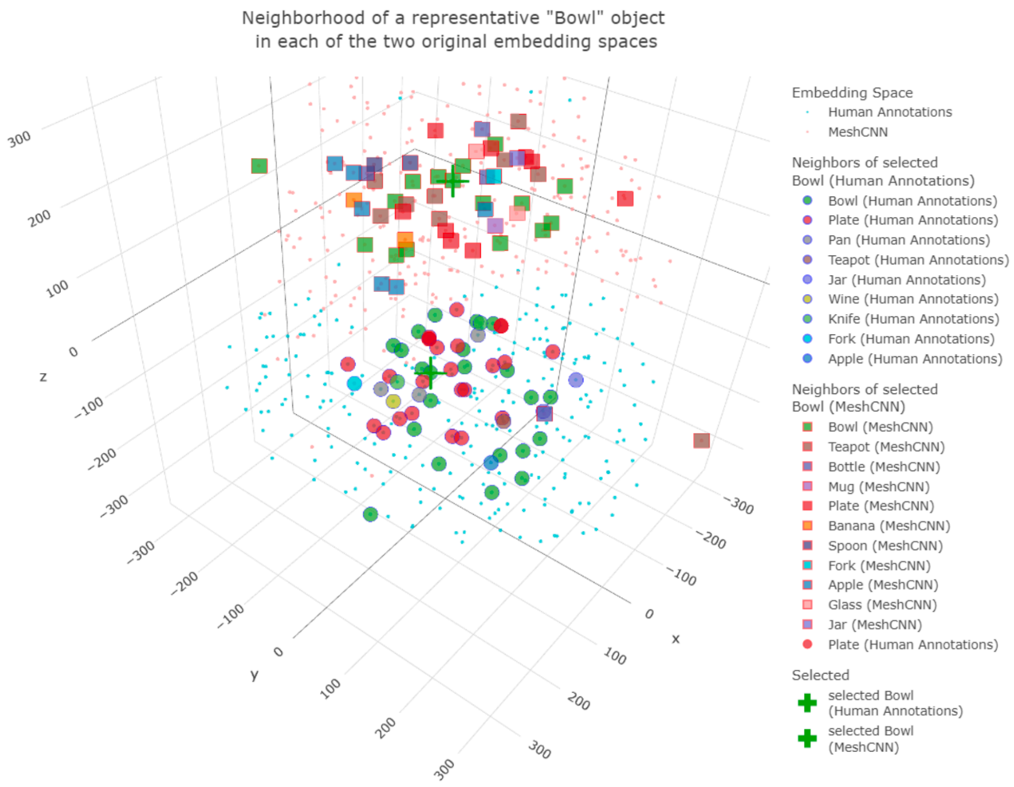
- Examining representative “bowl” embedding in each embedding space
- MLP nearest neighbors: mostly other bowls, or plates (other “palmar pinch” grasp)
  - Some others, due to disagreement in human annotation data
- MeshCNN nearest neighbors: diverse—bowls, plates, bottles, jars, teapots
  - Possible topological correspondence?
  - (This would exclude teapot handle)



Specific Nearest Neighbors

- When grasped for typical use or purpose, hand pose is different
  - (e.g., drinking from a bottle vs. eating from or filling a bowl)



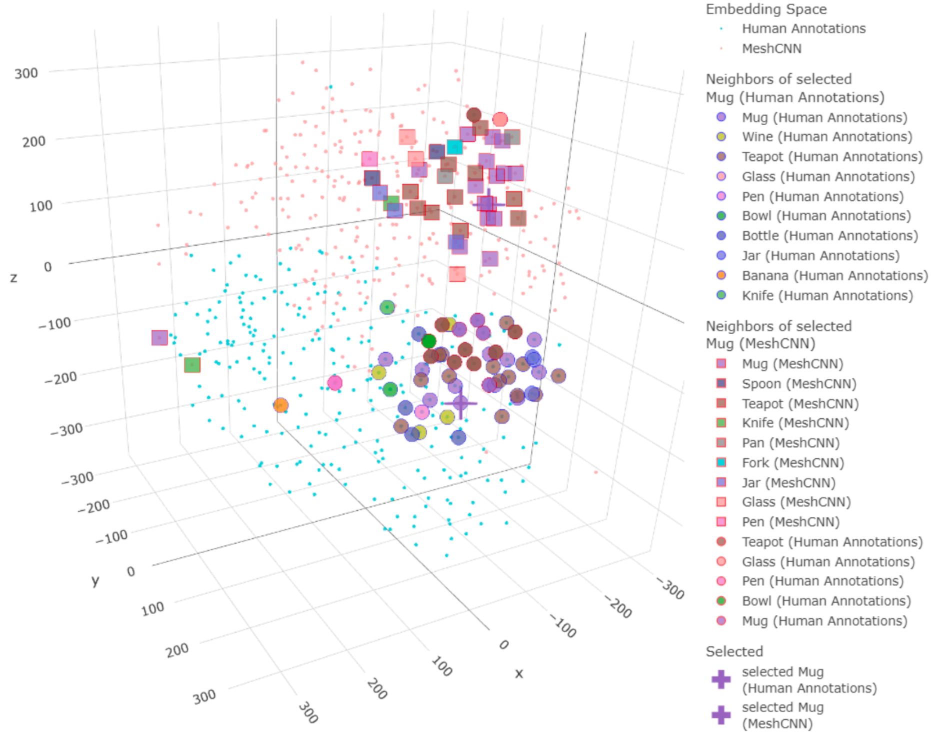


Neighborhood of a representative "Bowl" object in each of the two original embedding spaces

Specific Nearest Neighbors

- Examining representative “mug” embedding in each embedding space
- *Mug* belongs to “hook” grasp class (handles)
- Both MeshCNN and MLP performed well on this class
- Most nearest neighbors, among both embedding types, are mugs and teapots (“hook” class members)
- Nearest neighbors of MeshCNN mug embedding include *MLP* embeddings of teapot
  - Indicates that for this grasp class, the two models did converge somewhat

Neighborhood of a representative "Mug" object
in each of the two original embedding spaces



Conclusion

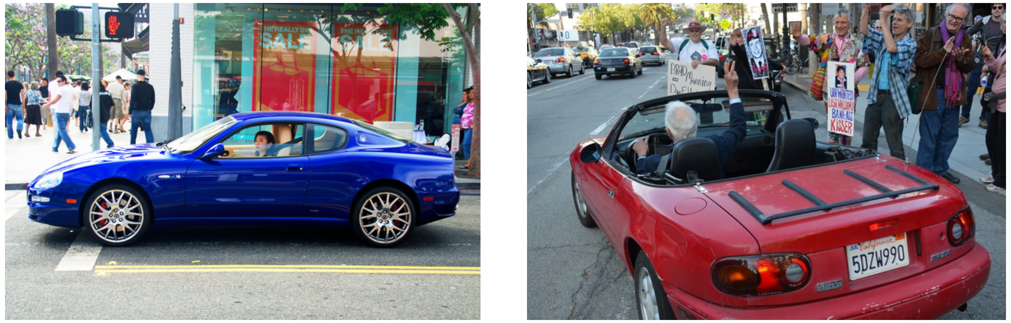
- **One of the first works to use ML to explore the Gibsonian/telic distinction**
- Grasping is typically a Gibsonian affordance
    - Based on object structure, as would be encoded in a geometric mesh representation
- Grasping for a particular use or purpose implies a *telic* affordance
- Encoding information this way, as human annotations did, results in different representation from the geometric representation learned by MeshCNN
- Even with the same output labels, models trained on this differing data do not appear to be learning equivalent representations that can be correlated to relationships between Gibsonian and telic affordances

Conclusion

- An embodied task like affordance classification is still difficult for even specialized models like MeshCNN that operate directly over 3D data
- Problem becomes more difficult if affordances are characterized by a telic/Gibsonian distinction
    - Even a single afforded behavior such as grasping, when annotated in telic terms, carries different information from the same affordance viewed Gibsonianly
- **Broader implication**: Gibsonian and telic affordances may carry fundamentally different information about an object
- **Shape underspecifies use**

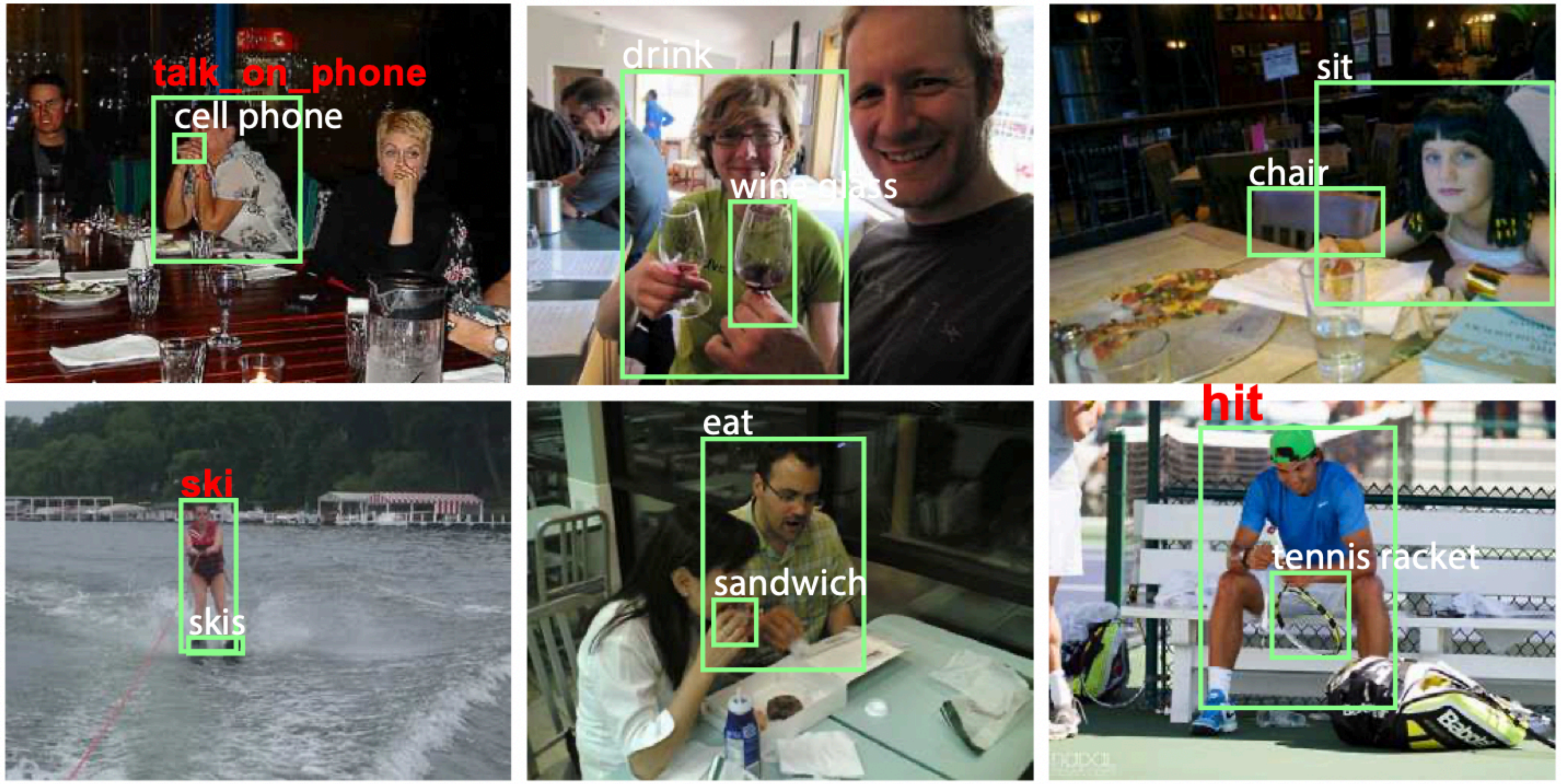
Implications for Action Recognition

- An affordance is not just any action taken with an object
  - Not every HOI exploits the object’s affordances
- An affordance is a distinct action possibility that an object allows an agent to take
  - More particular to that object than would arise by chance
- Simple object detection is not enough
  - False positives in HOI tasks often result from detecting the presence of an object when it is *not* being held for typical use or purpose



2 “ride” images from HICO-Det dataset

Implications for Action Recognition



False positives from Gkioxari et al. (2018)

Future Work

- **The hand is missing!**
  - Imagine about a cup on the table
  - Can potentially afford anything (drinking, pushing, etc.)
  - The final action cannot be predicted, unless coupled with a grasp pose
  - <object>+*grasp* can predict hand pose
  - Without hand pose, <object> cannot predict action!
- Explore methods for getting 3D hand pose information from a mesh (Grady et al., 2021; Hampali et al., 2021)

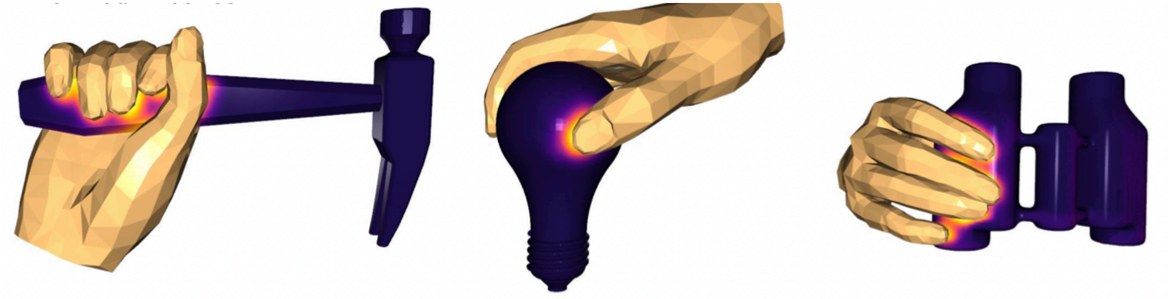


Image from Grady et al. (2021)

References

- Yu-Wei Chao, Yunfan Liu, Xieyang Liu, Huayi Zeng, and Jia Deng. 2018. Learning to detect human-object interactions. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 381–389. IEEE.
- Yu-Wei Chao, Zhan Wang, Yugeng He, Jiaxuan Wang, and Jia Deng. 2015. Hico: A benchmark for recognizing human-object interactions in images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1017–1025.
- Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. 2018. Scaling egocentric vision: The epic-kitchens dataset. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 720–736.
- Kuan Fang, Te-Lin Wu, Daniel Yang, Silvio Savarese, and Joseph J. Lim. 2018. Demo2vec: Reasoning object affordances from online videos. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2139–2147.
- James J Gibson. 1977. The theory of affordances. *Hilldale, USA*, 1(2):67–82.
- Georgia Gkioxari, Ross Girshick, Piotr Dollár, and Kaiming He. 2018. Detecting and recognizing human-object interactions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8359–8367.

References

- Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. 2017. The "something something" video database for learning and evaluating visual common sense. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5842–5850.
- Patrick Grady, Chengcheng Tang, Christopher D Twigg, Minh Vo, Samarth Brahmbhatt, and Charles C Kemp. 2021. Contactopt: Optimizing contact to improve grasps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1471–1481.
- Shreyas Hampali, Sayan Deb Sarkar, Mahdi Rad, and Vincent Lepetit. 2021. Handsformer: Key- point transformer for monocular 3d pose estimation ofhands and object in interaction. *arXiv preprint arXiv:2104.14639*.
- Rana Hanocka, Amir Hertz, Noa Fish, Raja Giryes, Shachar Fleishman, and Daniel Cohen-Or. 2019. MeshCNN: a network with an edge. *ACM Transactions on Graphics (TOG)*, 38(4):1–12.
- Rana Hanocka, Gal Metzer, Raja Giryes, and Daniel Cohen-Or. 2020a. Point2mesh. *ACM Transactions on Graphics (TOG)*, 39:126:1 – 126:12.

References

- Ajay Jain, Matthew Tancik, and Pieter Abbeel. 2021. Putting NERF on a diet: Semantically consistent few- shot view synthesis, pages 5865–5874.
- Helena Chmura Kraemer. 1980. Extension of the kappa coefficient. *Biometrics*, pages 207–216.
- J Richard Landis and Gary G Koch. 1977. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174.
- Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. 2021. Barf: Bundle-adjusting neural radiance fields. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5721–5731.
- David McNeely-White, Benjamin Sattelberg, Nathaniel Blanchard, and Ross Beveridge. 2020. Exploring the interchangeability of CNN embedding spaces. *arXiv preprint arXiv:2010.02323*.
- Gal Metzer, Rana Hanocka, Raja Giryes, and Daniel Cohen-Or. 2021. Self-sampling for neural point cloud consolidation. *ACM Transactions on Graphics*, 40:1–14.
- Oscar Jarillo Michel, Roi Bar-On, Richard Liu, Sagie Benaim, and Rana Hanocka. 2021. Text2mesh: Text-driven neural stylization for meshes. *ArXiv*, abs/2112.03221.

References

- Tushar Nagarajan, Christoph Feichtenhofer, and Kristen Grauman. 2019. Grounded human-object interaction hotspots from video (extended abstract). *ArXiv*, abs/1906.01963.
- James Pustejovsky. 2013. Dynamic event structure and habitat theory. In Proceedings of the 6th International Conference on Generative Approaches to the Lexicon (GL2013), pages 1–10.
- Tete Xiao, Quanfu Fan, Dan Gutfreund, Mathew Monfort, Aude Oliva, and Bolei Zhou. 2019. Reasoning about human-object interactions through dual attention networks. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3918–3927.
- Yufei Ye, Shubham Tulsiani, and Abhinav Gupta. 2021. Shelf-supervised mesh prediction in the wild.
- Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. 2020. PixelNERF: Neural radiance fields from one or few images.