Weakly-Supervised Learning of Dense Functional Correspondences

Stefan Stojanov* Linan Zhao* Yunzhi Zhang Daniel L. K. Yamins Jiajun Wu
Stanford University

Abstract

Establishing dense correspondences across image pairs is essential for tasks such as shape reconstruction and robot manipulation. In the challenging setting of matching across different categories, the function of an object, i.e.*, the effect that an object can cause on other objects, can guide how correspondences should be established. This is because object parts that enable specific functions often share similarities in shape and appearance. We derive the definition of* dense functional correspondence *based on this observation and propose a weakly-supervised learning paradigm to tackle the prediction task. The main insight behind our approach is that we can leverage vision-language models to pseudo-label multi-view images to obtain functional parts. We then integrate this with dense contrastive learning from pixel correspondences to distill both functional and spatial knowledge into a new model that can establish dense functional correspondence. Further, we curate synthetic and real evaluation datasets as task benchmarks. Our results demonstrate the advantages of our approach over baseline solutions consisting of off-the-shelf self-supervised image representations and grounded vision language models.*[1]

1. Introduction

Finding pixel correspondence across image pairs is fundamental for object understanding and is critical for applications like shape reconstruction [39, 43, 46, 72], editing [19], and object manipulation in robotics [17, 28, 29, 55]. This task requires reasoning beyond visual similarity in local appearance, geometry, and texture across images. It also involves structural similarity, *e.g.*, the part-whole relationships of objects and their part components, and semantic similarity, *e.g.*, the functional properties of parts of objects.

These aspects of similarity are essential for learning efficient generalizable systems for downstream applications. For example, in imitation learning in robotics, human demonstrations are a scarce and valuable data source. Given a demonstration with an object, such as pouring with a kettle, establishing dense functional correspondence with another object that supports this function, *e.g.*, a bottle, enables the efficient transfer of the demonstration.

It becomes harder to find dense correspondence when the input images shift from being two views of the same object to different objects from the same category, and finally to objects from distinct categories, as the visual similarity becomes less apparent. This work focuses on the most challenging scenario with objects from different categories. We aim to establish dense pixel-level correspondence between pairs of images containing objects with parts whose shape enables the execution of similar functions. Specifically, by "function", we refer to the effect one object can have on another object or substance, *e.g.*, the function "cut-with" for a knife and a spatula or "hang-onto" for objects with hooks.

Practically, training and evaluation for this task are challenging due to the lack of labeled data. Supervised training at scale is infeasible because manual dense correspondence labeling is intractable, emphasizing the need for a self- or weakly-supervised approach. For evaluation, while datasets exist for dense within-category correspondence [34, 68, 77] and sparse functional keypoint correspondence across categories [37], there is still no established task or dataset for dense correspondence across categories. In this work, we make progress toward addressing both the challenges of training and evaluation.

The key insight behind our training approach is that the capabilities of self-supervised image representations like DINOv2 [54] or Stable Diffusion [62] and vision language models (VLMs) [24, 78] are complementary but individually insufficient for solving this task. On the one hand, surprisingly accurate dense correspondences can be established using image features from pre-trained self-supervised models. This works well when the input images contain visually similar object instances from the same category, *e.g.*, two cats or two cars [87]. However, the accuracy decreases for the more generic scenario when objects come from distinct categories. On the other hand, VLMs can detect the bounding boxes of object parts with similar functions in a zero-shot manner [24, 78] but cannot perform fine-grained reasoning about correspondences across objects.

[*] Equal contribution.

[1] Project website: https://dense-functional-correspondence.github.io/

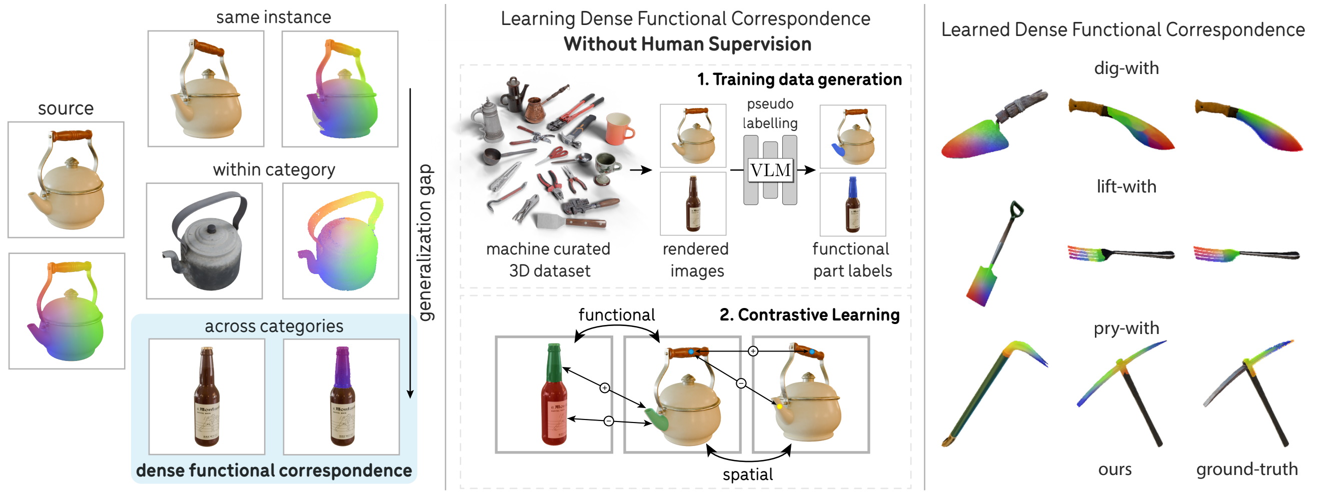

Figure 1. **Dense Functional Correspondence** refers to establishing dense correspondences across object instances based on function similarity (*e.g*., "pour-with"). This task is especially challenging when objects have visually different but functionally similar parts, requiring both semantic understanding, *i.e*., identifying which parts can perform the same function, and structural understanding, *i.e*., establishing dense correspondence across the parts at a surface-point-level based on functionally equivalent alignment. We propose a method to learn such correspondences with little human supervision, leveraging automated data curation and annotation, and dense contrastive learning.

We distill the strengths of each approach into a new model using a scalable technique that requires minimal human supervision. Specifically, we first obtain multi-view-consistent pseudo labels of functionally relevant regions of 3D object assets [7] using an off-the-shelf grounded VLM [78]. We then combine these labels with multi-view correspondences [17, 64] in a contrastive learning framework building on pre-trained DINOv2 [54] feature extractor. For evaluation, we define the dense functional 2D correspondence task and develop an annotation procedure based on aligning 3D object pairs in functionally equivalent poses.

In sum, we define the task of dense functional correspondence as a means for investigating cross-category dense correspondence. We then curate synthetic and real-world evaluation datasets for this task. We further propose a scalable, weakly-supervised method leveraging vision foundation models, which empirically outperforms baselines.

2. Related Work

Learning representations to establish dense functional correspondence requires fine-grained structural and semantic visual reasoning about objects. The most relevant prior works come from the object-level correspondences and affordance learning research domains. We also review recent work on vision foundation models and VLMs, focusing on works relating to fine-grained object understanding.

Learning Correspondences. For this work, it is relevant to categorize correspondence learning methods based on their degrees of generalization. For generalizing across geometric scene transforms, works on multi-view correspondences aim to match different views of the same scene [27, 63, 65], whereas optical flow techniques match consecutive video frames [25, 26, 69]. For generalizing within categories, NOCS-style representations [34, 76, 77] enable dense matching across instances of a category, whereas learning sparse keypoints [47, 66] enables sparse matching based on pre-defined semantic keypoint taxonomies. For generalizing across categories, Lai et al. [37] propose matching based on object function by learning five keypoints per object function category. The main drawback of keypoint-based correspondences is the requirement for a keypoint taxonomy, which by definition limits such techniques' capability to capture nuanced similarities across highly dissimilar objects (*e.g*., a bottle and a kettle). Through our dense functional correspondence formulation, we overcome the limitation of keypoint definitions and enable higher precision in downstream applications.

Learning Affordances. In his seminal work [18], James J. Gibson defines affordances as objects' "opportunities for interaction." Various object affordance formalisms have been developed in computer vision and robotics, such as estimating grasps [2, 16, 48, 49], and localizing affordance regions in 2D [5, 14, 44, 50–52] and 3D [10, 20, 83] through bounding boxes and segments [14, 50–52], heatmaps [10, 44, 50] or keypoints [58, 73, 81]. Early works adopt a fully supervised learning paradigm [2, 14, 50], while more recent works aim to use less supervision by learning from human object interaction videos [51], egocentric videos [41] or unlabeled exocentric images [40, 44]. Our work has two key distinctions: First, affordance heatmaps or segments identify object regions or parts in individual images. They do not allow for fine-grained spatial correspondence across object parts in different images (*e.g*., can identify the blades of two knives but cannot find correspondences for pixels between the tips or edges of the blades). Second, our focus is on object function – the effect an object can cause on something else, rather than the broader concept of affordance, which emphasizes potential interactions with a specific object instance (*e.g.*, striking with a hammer vs. holding). Last, our goal is to learn dense functional correspondence in a weakly-supervised manner, without relying on human annotations of ground-truth correspondences.

Vision Foundation Models. Recent developments in large-scale language [11, 60, 71] and image [59, 86] pretraining have led to the development of vision-language models (VLMs) capable of strong zero-shot performance through vision-question answering [42, 78], which have been adapted to reasoning about functional affordances and grasping in robotics [13, 24, 57, 85]. Powerful correspondence representations have been found to emerge [1, 67] in DINO [3, 54] and Stable Diffusion [62], which have led to direct applications in low-shot affordances [29] and object manipulation [12, 29, 35, 55, 56]. In this work, we leverage the complementary characteristics of VLMs and self-supervised image models to go beyond their individual capabilities for dense functional correspondence.

3. Dense Functional Correspondence

Distinct object categories with similar functionality, *e.g.*, a "kettle" and a "bottle" which can both pour liquid, may have different visual shapes and appearances as well as distinct part organizations. However, individual parts that serve the specific functionality of interest, *e.g.*, the spout of a kettle and the mouth of a bottle in this example, have a higher resemblance with each other than at the overall object level. Such consistency is a consequence of how form follows function – object parts that fulfill a specific function tend to remain consistent across objects, even if other parts vary greatly in shapes and arrangements. The part-level consistency provides a crucial ground from which we can derive the definition of functional correspondence (Section 3.1) and develop a corresponding evaluation data curation pipeline to benchmark this task (Section 3.2).

3.1. Problem Definition

We refer to the effect that an object causes on other objects or substances as an "object function." This concept has been widely studied in model generalization in visual computing [36, 37, 70, 89] and the development of categorization in humans [31, 38, 79]. Examples are shown in Figure 1, *e.g.*, "pour with." When executing a function with an object, such as pouring with a kettle, the functional part (the spout) follows a specific 3D trajectory. To replicate this function with a different object, *e.g.*, a bottle, the neck of the bottle would be aligned with the spout and follow the same trajectory. This illustrates how the *same* object function is fulfilled with *different* objects via aligning functionally equivalent parts, which is central to robotic applications with imitation learning approaches [23, 55, 90].

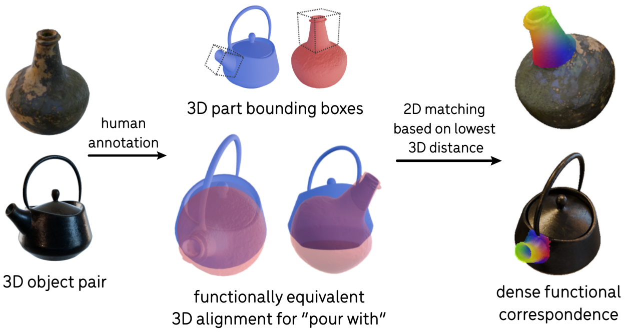
human annotation
3D part bounding boxes
2D matching based on lowest 3D distance
3D object pair
functionally equivalent 3D alignment for "pour with"
dense functional correspondence

Figure 2. **Annotation Pipeline (Evaluation Only).** Given a 3D object pair (left) and a function ("pour-with"), we annotate the functional alignment of two objects by aligning the functional parts in 3D (middle). Afterward, we derive dense 2D correspondences (right) based on 3D distances of corresponding object surface points, with matching pixels shown in the same color.

The above observations lead us to define dense functional correspondence through *3D object alignment based on functionally equivalent parts.* Specifically, given two objects (*e.g.*, a kettle and a bottle) and an object function (*e.g.*, "pour with"), the objects are aligned if and only if the parts that fulfill this function (*e.g.*, the kettle spout and the bottle neck) are spatially close to each other. The alignment induces an image-space distance: for any pair of pixels on the functional parts of two objects, the pixels are in *functional correspondence* if their respective surface points are close in 3D when the objects are aligned. Since this distance is defined at the pixel level, it is inherently *dense*.

Formally, the input consists of an object function $\mathcal{F}$ and an image pair (I_1, I_2), where each image is a view of a 3D object O_1 and O_2. Let $\pi^{-1} : I \to O$ represent the backprojection function that maps an image pixel to the corresponding 3D object surface point. We define $M(O; \mathcal{F})$ as the functional part of object O responsible for executing $\mathcal{F}$, and let $M(I; \mathcal{F})$ be its projected 2D mask in the image. In our setup, the functional parts of both objects, $M(O_1; \mathcal{F})$ and $M(O_2; \mathcal{F})$, are assumed to be aligned in 3D such that they follow the same trajectory when performing $\mathcal{F}$. We therefore define *dense functional correspondence* as a mapping $f(I_1, I_2; \mathcal{F}) : M(I_1; \mathcal{F}) \to M(I_2; \mathcal{F})$ that minimizes $\sum_{p \in M(I_1;\mathcal{F})} ||\pi^{-1}(p) - \pi^{-1}(f(p))||_2$. This ensures that pixel pairs in functional correspondence are from spatially close locations in 3D when the objects are aligned.

3.2. Evaluation Dataset Curation

The problem definition in Section 3.1 provides a guiding principle to obtain ground truth annotations for dense functional correspondence in image pairs by *aligning objects in 3D*. We introduce the annotation procedure and use it to construct both synthetic and real-world evaluation datasets for quantitative evaluation in Section 5.

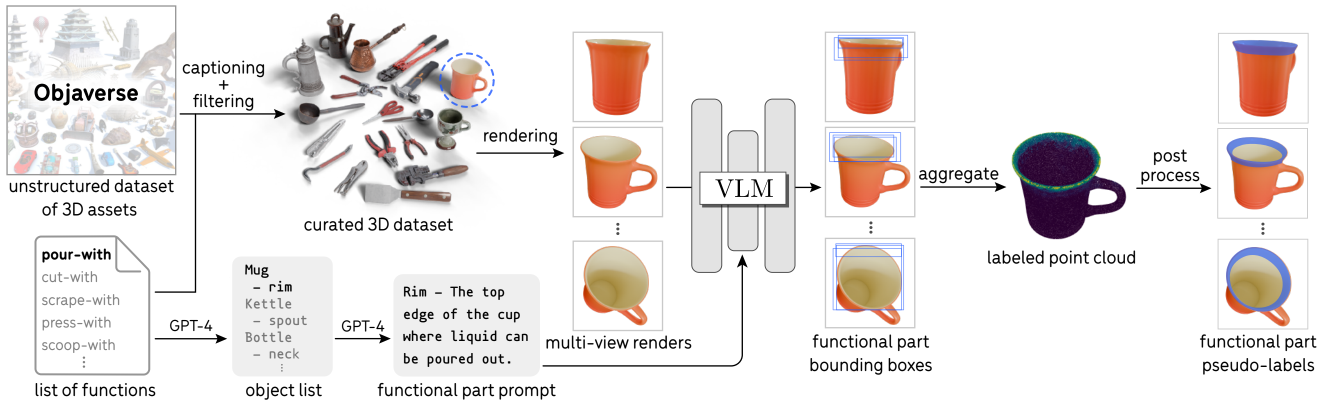

Figure 3. **Training Data Curation via VLM Pseudo Labeling.** Given a large unstructured dataset like Objaverse [7], we leverage off-the-shelf VLMs to curate and label the functional parts. Specifically, GPT-4 [53] generates category-specific functional part prompts, and CogVLM [78] produces bounding box proposals for multi-view image renderings, which are aggregated onto a 3D point cloud. The point cloud is post-processed to produce pixel-level functional part labels for training.

Annotation Procedure. To obtain ground-truth functional correspondence for an image pair, we assume each object is rendered from a known 3D asset. By aligning the two assets in 3D, we derive dense pixel correspondences between the images. This procedure eliminates the need for manual dense 2D labeling, enabling large-scale evaluation. An overview is shown in Figure 2.

Specifically, given two 3D meshes of objects supporting the same function, we first align them based on their functional parts and annotate a 3D bounding box around each functional part. Then, for a pair of rendered images, we unproject pixels from the functional parts onto the object surfaces and compute 3D distances between these points to perform minimum-cost matching. Pixels corresponding to visible surface points that are spatially close in 3D are matched. A detailed description of the annotation procedure is provided in the Supplement.

Synthetic Evaluation Dataset. We use the 3D assets from Objaverse [7], a large collection of diverse 3D models, to obtain a synthetic evaluation dataset. We hand-label 3D annotations for 950 pairs of assets from Objaverse spanning 24 functions, selected for quality and diversity. See Section 4.1 for how the assets and functions are selected. As such, 85% of the ground-truth pairs contain across-category correspondences. From these 3D annotations, we derive 1,800+ unique 2D image pairs rendered from the 3D assets, with ground truth dense functional correspondences.

Real Evaluation Dataset. Setting up a real-world benchmark is crucial for measuring model performance on real images. Thus, we utilize the HANDAL dataset [20], which contains images and 3D reconstructions of real-world objects. After manually fixing the geometry of the 3D scanned assets, *e.g.*, the missing concavities of mugs, pots and pans, we label 190 asset pairs spanning 13 functions. This results in a real evaluation dataset of 500+ unique 2D real image pairs with ground-truth functional correspondence labels.

4. Approach

Our goal is to develop a scalable learning framework for dense functional correspondences without relying on human-labeled ground truth. Since this task requires both semantic and structural knowledge, we distill from off-the-shelf VLMs to obtain pseudo-labeled training data (Sec. 4.1), which is further combined with dense spatial correspondences from synthetic data in a contrastive learning framework (Sec. 4.2). This approach enables the model to generalize to real-world data, as we will show in Sec. 5.

4.1. Dataset of Pseudo-labeled Functional Parts

A dataset for learning dense functional correspondences at scale requires a diverse source of object images, a diverse taxonomy of functions and associated functional parts, and a low-cost, reliable means for part labeling.

Image Data. Our approach requires a large and diverse multi-view image dataset where functional parts are visible. Existing multi-view object datasets [61, 80, 84, 88] are suboptimal because they have few desired objects like tools and utensils, the objects are in canonical poses that may not reveal functional parts, or are placed in cluttered contexts where occlusions often occur. To overcome this, we render high-quality images from the Objaverse [7] dataset using ray-tracing and HDRI environments [21] in Blender [6], obtaining arbitrary amounts of diverse multi-view data.

Object and Function Taxonomy. To curate relevant object assets for our training dataset, we prompt GPT-4 [53] for common functions and refer to object functions studied in [37, 49]. Then, we prompt GPT-4 to generate a comprehensive list of object categories for each function. After deduplication and manual filtering, our taxonomy has 24 functions and 160 object categories.

Object Asset Selection. To retrieve relevant assets from the noisy-labeled Objaverse dataset based on the list of object categories, we utilize asset captions from Caption3D [45]. We use Llama 3.1 [15] to summarize the captions into category names and use Llama word embeddings to match the summaries to our category list. Finally, we prompt Llama to verify these matches. To ensure diversity, we cap each category at 200 assets. To ensure quality, we manually filter the retrieved assets to obtain 8,285 assets in total, 80% of which are used for training. Details about prompting, filtering, and the taxonomy are included in the Supplement.

Functional Part Pseudo-Labeling. Labeling data at scale using large pre-trained models has been shown as an effective approach for achieving high performance with minimal human effort [75, 82]. The key elements for success are a sufficiently accurate pre-trained model and a low-cost and reliable procedure for rejecting low-quality labels. Grounded VLMs [8, 42, 78] have shown remarkable capabilities for zero-shot prompt-based object detection. We, therefore, use the 17B grounded CogVLM [78] model, which has state-of-the-art referring expression detection performance. For an overview of the pseudo-labeling pipeline, see Figure 3. Given our list of object categories and functions, we prompt GPT-4 to obtain the names and appearance descriptions of functional parts to serve as prompts for CogVLM, which we then manually filtered and deduplicated. Because functional part names can be different across categories (*e.g.*, the spout of a kettle vs. the neck of a bottle), we generate these functional part lists separately for each category. We empirically found that prompting CogVLM with part names and appearance descriptions significantly improves the bounding box predictions.

Given a set of rendered views for an object and a functional part text prompt, we generate bounding box predictions with CogVLM [78], which vary due to sampling in VLM inference. The accuracy of the bounding boxes also depends on viewpoint because of part pose and visibility. To aggregate these possibly noisy labels and obtain a final part label, we sample a dense point cloud on the surface of the object, and accumulate the 2D labels across views onto the 3D points. We post-process these labeled point clouds to generate 2D masks for views rendered for training.

This dataset curation and pseudo-labeling procedure allows us to generate a large dataset of functional part segmentation labels with relatively little human effort, which was mostly necessary for prompt engineering and quality control. In this work, we apply this approach on the $\approx$ 600K labeled meshes from Caption3D, but it is straightforward to scale up to the millions of meshes in Objaverse-XL [9].

4.2. Learning Dense Functional Correspondence

To learn dense functional correspondence, we train a feature embedding that captures both the high-level function semantics and the structural similarity between functional parts. For instance, given a bottle and a kettle for the function "pour-with," the features for the neck of the bottle and the spout of the kettle should be similar. Moreover, the mouth of the bottle and the tip of the kettle spout should be in correspondence, as well as the bottom of the bottle's neck and the bottom of the kettle's spout. To achieve this, we train a function-conditioned network on top of frozen DINOv2 [54] and CLIP [59] (illustrated in Figure 4), that is applied at the local feature level. Because of significant developments in object segmentation [30, 33] and our focus on object-level understanding, we assume that the input images consist of segmented objects.

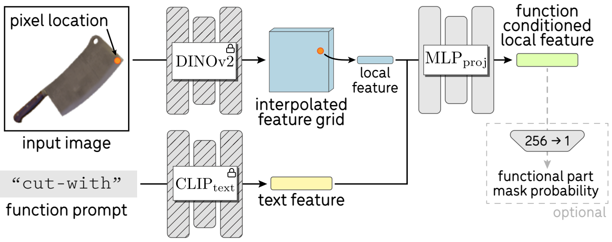

Figure 4. **Local Functional Feature Extraction.** To obtain dense functionally conditioned features, we apply an MLP on top of a function text embedding and the spatial DINO features. The MLP is trained with both functional and spatial contrastive losses.

Function-Conditioned MLP. Given an image and a function, we first extract the image features from the last three blocks of DINOv2 and the function conditioning from CLIP text embeddings. We average the DINOv2 features from each block using learned weights into a single feature grid, and use bilinear interpolation to obtain a feature vector for each pixel location. Then, we concatenate the image feature with the CLIP embedding of the function and pass it through a 3-layer MLP, which produces the final feature at each pixel location. This network can be thought of as a function-conditioned version of the final projection layer used in contrastive learning [4, 22]. We parameterize our model as $g_\theta(p|I, \mathcal{F})$, which outputs the normalized feature of pixel p on image I conditioned on the function $\mathcal{F}$.

We also investigate the option of adding an extra fully connected layer that maps the output feature vector to a prediction for the functional part mask. This allows us to obtain a binary functional part mask at inference time.

Functional Part Contrastive Learning. To distill the knowledge of functional part semantics from the VLM, we use contrastive learning based on the pseudo-labeled functional part masks. The parts from two objects that can be used to perform the same function should share a more similar embedding space. Specifically, given two images, I_1 and I_2 of objects that can perform the same function $\mathcal{F}$, let the functional part segments be P_1^+ and P_2^+. Then, define the rest of the objects' pixels as P_1^- and P_2^-. Learning correspondence requires the pixels in P_1^+ to be similar to the ones in P_2^+ but different from the ones in P_2^-. In addition, to encourage the model to focus on the functionally relevant regions of objects, we add a term that pushes the features of P_1^- away from that of P_2^-.

Let $\text{sim}(x, y|I_1, I_2, \mathcal{F}) = g(x|I_1, \mathcal{F}) \cdot g(y|I_2, \mathcal{F})$ represent the feature similarity between pixel x on image I_1 and pixel y on image I_2 when conditioned on function $\mathcal{F}$. For brevity, we short-hand it to $\text{sim}(x, y)$ below. The infoNCE loss [74] for the function-part contrastive learning given $(p_1^+, p_1^-, p_2^+, p_2^-) \in (P_1^+, P_1^-, P_2^+, P_2^-)$ is thus

$$\mathcal{L}_{\text{func}} = -\log \frac{e^{\text{sim}(p_1^+,p_2^+)/\tau}}{e^{\text{sim}(p_1^+,p_2^+)/\tau} + e^{\text{sim}(p_1^+,p_2^-)/\tau} + e^{\text{sim}(p_1^-,p_2^-)/\tau}} \tag{1}$$

for temperature τ.

When the model predicts functional part masks, we add a binary cross-entropy loss $\mathcal{L}_{\text{mask}}$ to compare the predicted mask with the pseudo-labeled functional part segment.

Part Structure via Multi-view Contrastive Learning. If we train the embedding with only the functional part contrastive loss, we inevitably run into mode collapse issues. That is, the whole spout of the kettle would have the same features regardless of the pixel's spatial location. To preserve the structural information, we apply dense contrastive learning from multi-view correspondences.

Given two views of an object, we can find corresponding pixels that project to the same location in 3D space. We require a view-invariant feature embedding – a pixel should have high similarity with its corresponding pixel on the other image but remain different from all the other pixels. This encourages the model to learn the structural information of the object, to not collapse the embedding space, and to encode the object part consistently across different views. This multi-view contrastive objective only applies to two images of the same asset. However, because the underlying DINOv2 embedding space enables generalization for visually similar regions, the trained feature embedding can retain information about the structural similarities between functional parts *across* categories.

Formally, let q be a pixel in the first view I, q'_+ be a pixel in the second view I' that corresponds to the same location in 3D as q, and any other pixel on I' be denoted as q'_-. The multiview contrastive objective is

$$\mathcal{L}_{\text{spatial}} = -\log \frac{e^{\text{sim}(q,q'_+)/\tau}}{e^{\text{sim}(q,q'_+)/\tau} + e^{\text{sim}(q,q'_-)/\tau}}. \tag{2}$$

Combining the terms, we obtain the final loss

$$\mathcal{L} = \mathcal{L}_{\text{func}} + \lambda_{\text{spatial}}\mathcal{L}_{\text{spatial}} + \lambda_{\text{mask}}\mathcal{L}_{\text{mask}}. \tag{3}$$

4.3. Implementation Details

We use DINOv2-B as the backbone and an image size of 224. The MLP projector has 3 layers with 1024 hidden dimensions each. We use the Adam [32] optimizer with default hyperparameters, a batch size of 50 image pairs, 128 positive and negative sampled points on each image, and a learning rate of 1×10^{-4}. In addition, we use a weight of $\lambda_{\text{spatial}} = 10$ for the spatial loss and a weight of $\lambda_{\text{mask}} = 1$ for the mask loss. We use random-color background augmentation during training following [17]. A sensitivity analysis of loss weights and a breakdown of computational costs are provided in the Supplement.

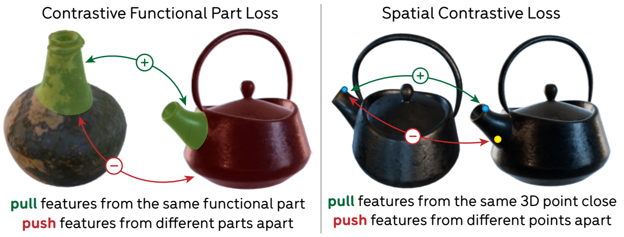
Contrastive Functional Part Loss
Spatial Contrastive Loss
+
−
+
−
pull features from the same functional part
push features from different parts apart
pull features from the same 3D point close
push features from different points apart

Figure 5. **Training Objectives.** To ensure functional part similarity in the learned feature space, we use a part-level contrastive objective to distill *functional part semantics* from VLMs (left). The *spatial* contrastive loss (right) serves a complementary role and prevents the model from collapsing predictions for different regions of a part, *e.g.*, the top and bottom of a kettle spout.

5. Experiments

In this section, we benchmark our approach in Sec. 4 and several baseline solutions on the dense functional correspondence task. Since our problem formulation in Sec. 3.1 requires a function as input and focuses on matches within functional parts, it differs significantly from existing benchmarks on semantic correspondence [47, 66]. As such, we leverage the evaluation datasets from Sec. 3.2.

5.1. Metrics

We evaluate dense functional correspondence from two different aspects: *correspondence label transfer*, which assesses the precision with which the model can transfer one functional part to another, and *correspondence discovery*, which assesses the model's ability to identify relevant functional correspondences without any reference input labels.

Correspondence Label Transfer. To evaluate the precision of the correspondences that can be found using the learned features, we use normalized pixel distance (Normalized Dist) and percentage of correct keypoints (PCK).

Specifically, let the ground-truth correspondences between images I_1, I_2 given the function $\mathcal{F}$ be $\{p_1^1, p_1^2, \cdots, p_1^k\}, \{p_2^1, p_2^2, \cdots, p_2^k\}$. For each pixel p_1^i on image I_1, we can find its most similar match $p_2^{j(i)}$ on I_2 using feature similarity. The normalized distance metric is simply the mean of $||p_2^{j(i)} - p_2^i||_2$ normalized by the image size, and PCK@k pixels is the mean of $\mathbb{1}_{||p_2^{j(i)} - p_2^i||_2 < k}$.

Correspondence Discovery. In addition to label transfer, models should discover the relevant set of functional correspondences on its own, without assuming a priori that the

Model	**Correspondence Label Transfer**			**Correspondence Discovery**			
	Normalized Dist (↓)	PCK@23p (↑)	PCK@10p (↑)	Best F1@23p (↑)	Best F1@10p (↑)	AP@23p (↑)	AP@10p (↑)
Synthetic Evaluation Dataset							
Chance	0.310	0.165	0.046	0.416	0.176	0.256	0.093
DINO [54]	0.212	0.381	0.148	0.578	0.281	0.381	0.130
SD [87]	0.268	0.298	0.126	0.479	0.231	0.267	0.097
SD-DINO [87]	0.227	0.376	0.161	0.563	0.301	0.341	0.144
CogVLM [78] + DINO	0.180	0.416	0.158	0.678	0.333	0.556	0.188
ManipVQA-P [24] + DINO	0.223	0.346	0.130	0.575	0.269	0.418	0.134
ManipVQA-F [24] + DINO	0.272	0.259	0.093	0.528	0.244	0.320	0.097
Ours (functional only)	0.228	0.287	0.094	0.575	0.233	0.441	0.112
Ours (spatial only)	0.204	0.470	0.227	0.610	0.369	0.412	0.211
Ours (full without mask loss)	**0.170**	**0.486**	**0.227**	0.768	0.470	**0.685**	**0.338**
Ours (full with mask loss)	0.172	0.480	0.223	**0.774**	**0.471**	0.684	0.330
Real Evaluation Dataset							
Chance	0.313	0.170	0.045	0.417	0.167	0.248	0.087
DINO [54]	0.206	0.408	0.159	0.589	0.294	0.382	0.138
SD [87]	0.259	0.309	0.127	0.503	0.238	0.285	0.101
SD-DINO [87]	0.220	0.385	0.163	0.577	0.301	0.343	0.142
CogVLM [78] + DINO	0.172	0.440	0.169	0.695	0.350	0.561	0.198
ManipVQA-P [24] + DINO	0.204	0.398	0.153	0.600	0.295	0.420	0.148
ManipVQA-F [24] + DINO	0.256	0.309	0.114	0.575	0.281	0.368	0.126
Ours (functional only)	0.200	0.336	0.115	0.652	0.283	0.532	0.148
Ours (spatial only)	0.203	0.472	0.228	0.708	0.353	0.382	0.182
Ours (full without mask loss)	**0.152**	**0.516**	**0.249**	0.775	0.476	0.691	0.344
Ours (full with mask loss)	0.153	0.501	0.235	**0.808**	**0.502**	**0.730**	**0.360**

Table 1. **Quantitative Evaluation** on the synthetic and real evaluation datasets. The simplest baselines, self-supervised features from Stable Diffusion and DINOv2, perform relatively poorly. Adding semantic knowledge from predicted functional part labels from VLMs can offer slight improvement. Our approach, combining the strengths of both self-supervised features and VLMs, achieves the best performance.

relevant pixels on one image have been given. This capability is essential for potential downstream applications such as object alignment in robot object manipulation.

First, since we assume that the input images are segmented, let M_1, M_2 be the object masks for images I_1, I_2. For every pixel $p_1^i \in M_1$, we find its most similar match $p_2^{j(i)}$ on I_2 and find the backward match of $p_2^{j(i)}$ on I_1, denoted as q_1^i. As such, $||p_1^i - q_1^i||_2$ captures the level of cycle-consistency of the match. We therefore construct a score $s = (1 - ||p_1^i - q_1^i||_2) \cdot \text{sim}(p_1^i, p_2^{j(i)})$ to rank each pair of $(p_1^i, p_2^{j(i)})$, using both similarity and cycle consistency.

Then, we consider the top $t\%$ of all pairs as "discovered" and compare them with the ground-truth. A discovered pair (x_1, x_2) is equivalent to a ground-truth pair (y_1, y_2) if both end points are within k pixels of the ground truth. Increasing t results in higher recall but potentially lower precision: the number of discovered ground-truth correspondences monotonically increases while the percentage of correct correspondence tends to decrease. Sweeping t produces a precision-recall curve, from which we can calculate the best F1 score (at k pixels) and the average precision (AP) (at k pixels). Formally, Best F1 $= \max_t \frac{2\times\text{Precision}_t \times \text{Recall}_t}{\text{Precision}_t + \text{Recall}_t}$ and AP $= \sum_t (\text{Recall}_t - \text{Recall}_{t-1})\text{Precision}_t$.

5.2. Baselines

We describe several baseline methods below.

Self-Supervised Features. Powerful correspondences emerge in the feature space of large pre-trained vision foundation models, as reviewed in Sec. 2. We use features extracted from DINOv2 [54], Stable Diffusion [62, 87], and fused features of the two [87] as baselines. We use feature-level similarity between pixel pairs to find correspondences.

Self-Supervised Features and VLM Grounding. Since our task requires both semantic and structural reasoning based on the function prompt, these baselines chain a VLM that grounds functional parts with a pre-trained model that provides structural priors. Given an image pair, we use functional part bounding boxes generated by the VLM for each image, and then use self-supervised features to find correspondences within these part labels. This approach can benefit both label transfer and discovery because the functional part prediction adds a constraint on the space of possible matches, making it easier to find accurate matches. We consider two VLMs as the functional part grounding modules to be combined with off-the-shelf DINOv2 features:

- CogVLM [78], which outputs bounding boxes based on prompts of the functional part.

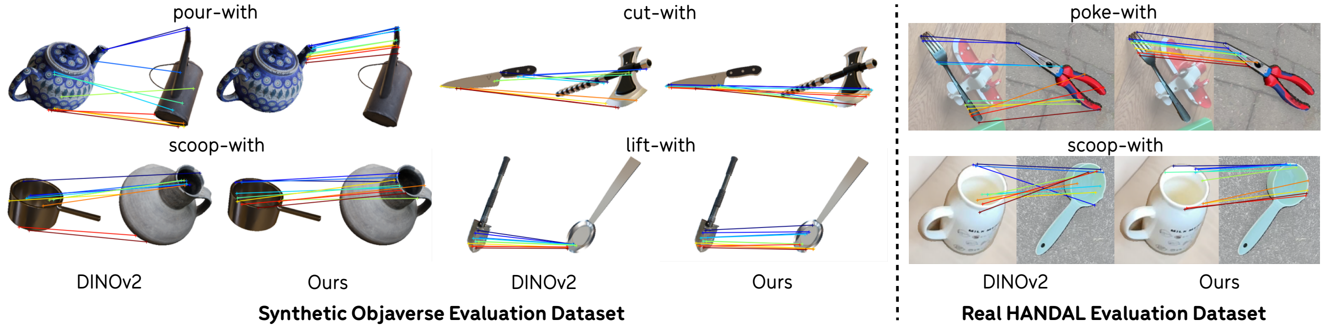

Figure 6. **Correspondence Discovery Comparisons.** We observe that our approach more reliably retrieves the functionally relevant correspondences than off-the-shelf DINOv2. The top 10 highest-ranked matches are shown.

- ManipVQA [24], an affordance-grounding model that outputs bounding boxes conditioned on actions. We use the 7B model in our experiments. We also prompt ManipVQA in two ways, one with the functional part name and the other with the function itself because the model is finetuned for robotic tasks. We refer to these as ManipVQA-P and ManipVQA-F, respectively.

5.3. Quantitative Comparisons

Results in Table 1 evaluate the performance of our method and baseline solutions on the synthetic and real evaluation datasets introduced in Sec. 3.2. Results show that our model trained on fully synthetic data can generalize to real images.

Compared to baseline solutions that solely use self-supervised features, our full model – trained with both functional and spatial contrastive loss – consistently outperforms. These metrics demonstrate that the pseudo-label quality is sufficient for learning meaningful functional correspondences. Additionally, given that the evaluation dataset predominantly includes cross-category pairs, Table 1 illustrates that self-supervised features struggle with cross-category generalization. Further evidence is provided in the Supplement, where we present a detailed breakdown of metrics for both within- and across-category pairs.

Compared to baselines using VLM grounding, even with CogVLM bounding boxes as additional functional part information, off-the-shelf DINOv2 features underperform relative to our full model. The margin is generally smaller, which highlights the importance of understanding the context of the function. On the other hand, ManipVQA outputs less accurate bounding boxes, which is reflected in the metrics. In particular, prompting with the part instead of the function is significantly better, which shows the difficulty of zero-shot affordance grounding given a function name. Note also that running CogVLM inference is roughly 50 times slower than our model and running ManipVQA inference is roughly 1000 times slower than our model.

Ablations. We ablate the role of the functional and spatial contrastive loss in Table 1. The model trained solely with functional loss performs poorly in both label transfer and correspondence discovery. The model trained solely with spatial loss is better but still falls short compared to the full model due to its lack of functional information. Finally, models with and without mask loss share similar performances. The model with mask loss does outperform the model without it in all metrics for correspondence discovery on the real evaluation dataset, which represents the least constrained and most realistic case. This optional mask prediction module can learn functional part masks with minimal additional cost.

5.4. Qualitative Results

We present results for correspondence label transfer in Figure 1 and correspondence discovery in Figure 6. Our model predictions not only capture object parts specific to the input function, but also preserve the structural relation among parts. Figure 6 shows top 10 matches according to the score from Section 5.1 separated by 5 pixels each. DINOv2 features are not function-aware and result in inaccurate matching, especially in cross-category image pairs. In comparison, our model produces dense matches between functional parts from different object categories with high spatial precision, *e.g*., matching the rim of a saucepan with the rim of a jug. Overall, our model demonstrates a deep understanding of functional and structural information of objects, which produces high-quality dense functional correspondences.

6. Conclusion

We have introduced the problem of dense functional correspondence, where input images contain objects with similar functionality but possibly come from distinct object categories. We have proposed a principled approach to obtain dense 2D functional correspondences from 3D object alignments and curated datasets for comprehensive evaluations. To tackle the task, we have presented a weakly-supervised framework that distills semantic information from vision-language models, while learning structural information through tuning self-supervised features with a multi-view contrastive loss. Our model outperforms a set of baselines in both synthetic and real-world benchmarks.

Acknowledgments. This work is in part supported by NSF RI #2211258 and #2338203, NSF CAREER #1844724, NSF NCS-FR #2123963, ONR YIP N00014-24-1-2117, ONR MURI N00014-22-1-2740, ONR N00014-20-1-2589, ONR MURI N00014-21-1-2801, ONR MURI N00014-24-1-2748, and Simons Foundation grant SFI-AN-NC-GB-Culmination-00002986-05. We also thank the Stanford HAI for their support with computing resources.

References

[1] Shir Amir, Yossi Gandelsman, Shai Bagon, and Tali Dekel. Deep vit features as dense visual descriptors. *arXiv preprint arXiv:2112.05814*, 2021. 3

[2] Samarth Brahmbhatt, Cusuh Ham, Charles C Kemp, and James Hays. Contactdb: Analyzing and predicting grasp contact via thermal imaging. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8709–8719, 2019. 2

[3] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 3

[4] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020. 5

[5] Ching-Yao Chuang, Jiaman Li, Antonio Torralba, and Sanja Fidler. Learning to act properly: Predicting and explaining affordances from images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 975–983, 2018. 2

[6] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018. 4

[7] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13142–13153, 2023. 2, 4

[8] Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. Molmo and pixmo: Open weights and open data for state-of-the-art multimodal models. *arXiv preprint arXiv:2409.17146*, 2024. 5

[9] Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, et al. Objaverse-xl: A universe of 10m+ 3d objects. *Advances in Neural Information Processing Systems*, 36, 2024. 5

[10] Shengheng Deng, Xun Xu, Chaozheng Wu, Ke Chen, and Kui Jia. 3d affordancenet: A benchmark for visual object affordance understanding. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1778–1787, 2021. 2

[11] Jacob Devlin. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 3

[12] Norman Di Palo and Edward Johns. On the effectiveness of retrieval, alignment, and replay in manipulation. *IEEE Robotics and Automation Letters*, 2024. 3

[13] Kairui Ding, Boyuan Chen, Ruihai Wu, Yuyang Li, Zongzheng Zhang, Huan-ang Gao, Siqi Li, Guyue Zhou, Yixin Zhu, Hao Dong, et al. Preafford: Universal affordance-based pre-grasping for diverse objects and environments. *arXiv preprint arXiv:2404.03634*, 2024. 3

[14] Thanh-Toan Do, Anh Nguyen, and Ian Reid. Affordancenet: An end-to-end deep learning approach for object affordance detection. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 5882–5889. IEEE, 2018. 2

[15] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. 5

[16] Hao-Shu Fang, Chenxi Wang, Minghao Gou, and Cewu Lu. Graspnet-1billion: A large-scale benchmark for general object grasping. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11444–11453, 2020. 2

[17] Peter R Florence, Lucas Manuelli, and Russ Tedrake. Dense object nets: Learning dense visual object descriptors by and for robotic manipulation. In *Conference on Robot Learning*, pages 373–385. PMLR, 2018. 1, 2, 6

[18] James J Gibson. The ecological approach to visual perception: Classic edition. *Taylor & Francis*, 1979. 2

[19] Yuchao Gu, Yipin Zhou, Bichen Wu, Licheng Yu, Jia-Wei Liu, Rui Zhao, Jay Zhangjie Wu, David Junhao Zhang, Mike Zheng Shou, and Kevin Tang. Videoswap: Customized video subject swapping with interactive semantic point correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7621–7630, 2024. 1

[20] Andrew Guo, Bowen Wen, Jianhe Yuan, Jonathan Tremblay, Stephen Tyree, Jeffrey Smith, and Stan Birchfield. Handal: A dataset of real-world manipulable object categories with pose annotations, affordances, and reconstructions. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 11428–11435. IEEE, 2023. 2, 4

[21] HDRI Haven. Hdri haven. `https://hdri-haven.com`, 2024. Accessed: 2024-9. 4

[22] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020. 5

[23] Nick Heppert, Max Argus, Tim Welschehold, Thomas Brox, and Abhinav Valada. Ditto: Demonstration imitation by trajectory transformation. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024. 3

[24] Siyuan Huang, Iaroslav Ponomarenko, Zhengkai Jiang, Xiaoqi Li, Xiaobin Hu, Peng Gao, Hongsheng Li, and Hao Dong. Manipvqa: Injecting robotic affordance and physically grounded information into multi-modal large language models. *arXiv preprint arXiv:2403.11289*, 2024. 1, 3, 7, 8

[25] Zhaoyang Huang, Xiaoyu Shi, Chao Zhang, Qiang Wang, Ka Chun Cheung, Hongwei Qin, Jifeng Dai, and Hongsheng Li. Flowformer: A transformer architecture for optical flow. In *European conference on computer vision*, pages 668–685. Springer, 2022. 2

[26] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2462–2470, 2017. 2

[27] Wei Jiang, Eduard Trulls, Jan Hosang, Andrea Tagliasacchi, and Kwang Moo Yi. Cotr: Correspondence transformer for matching across images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6207–6217, 2021. 2

[28] Zhenyu Jiang, Hanwen Jiang, and Yuke Zhu. Doduo: Learning dense visual correspondence from unsupervised semantic-aware flow. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 12420–12427. IEEE, 2024. 1

[29] Yuanchen Ju, Kaizhe Hu, Guowei Zhang, Gu Zhang, Mingrun Jiang, and Huazhe Xu. Robo-abc: Affordance generalization beyond categories via semantic correspondence for robot manipulation. *arXiv preprint arXiv:2401.07487*, 2024. 1, 3

[30] Lei Ke, Mingqiao Ye, Martin Danelljan, Yu-Wing Tai, Chi-Keung Tang, Fisher Yu, et al. Segment anything in high quality. *Advances in Neural Information Processing Systems*, 36, 2024. 5

[31] Deborah G Kemler Nelson, Rachel Russell, Nell Duke, and Kate Jones. Two-year-olds will name artifacts by their functions. *Child development*, 71(5):1271–1288, 2000. 3

[32] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. 6

[33] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023. 5

[34] Akshay Krishnan, Abhijit Kundu, Kevis-Kokitsi Maninis, James Hays, and Matthew Brown. Omninocs: A unified nocs dataset and model for 3d lifting of 2d objects. *arXiv preprint arXiv:2407.08711*, 2024. 1, 2

[35] Yuxuan Kuang, Junjie Ye, Haoran Geng, Jiageng Mao, Congyue Deng, Leonidas Guibas, He Wang, and Yue Wang. Ram: Retrieval-based affordance transfer for generalizable zero-shot robotic manipulation. *arXiv preprint arXiv:2407.04689*, 2024. 3

[36] Hamid Laga, Michela Mortara, and Michela Spagnuolo. Geometry and context for semantic correspondences and functionality recognition in man-made 3d shapes. *ACM Transactions on Graphics (TOG)*, 32(5):1–16, 2013. 3

[37] Zihang Lai, Senthil Purushwalkam, and Abhinav Gupta. The functional correspondence problem. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15772–15781, 2021. 1, 2, 3, 4

[38] Barbara Landau, Linda Smith, and Susan Jones. Object shape, object function, and object name. *Journal of memory and language*, 38(1):1–27, 1998. 3

[39] Yixing Lao, Xiaogang Xu, Xihui Liu, Hengshuang Zhao, et al. Corresnerf: Image correspondence priors for neural radiance fields. *Advances in Neural Information Processing Systems*, 36:40504–40520, 2023. 1

[40] Gen Li, Varun Jampani, Deqing Sun, and Laura Sevilla-Lara. Locate: Localize and transfer object parts for weakly supervised affordance grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10922–10931, 2023. 2

[41] Gen Li, Nikolaos Tsagkas, Jifei Song, Ruaridh Mon-Williams, Sethu Vijayakumar, Kun Shao, and Laura Sevilla-Lara. Learning precise affordances from egocentric videos for robotic manipulation. *arXiv preprint arXiv:2408.10123*, 2024. 2

[42] Ziyi Lin, Chris Liu, Renrui Zhang, Peng Gao, Longtian Qiu, Han Xiao, Han Qiu, Chen Lin, Wenqi Shao, Keqin Chen, et al. Sphinx: The joint mixing of weights, tasks, and visual embeddings for multi-modal large language models. *arXiv preprint arXiv:2311.07575*, 2023. 3, 5

[43] Philipp Lindenberger, Paul-Edouard Sarlin, Viktor Larsson, and Marc Pollefeys. Pixel-perfect structure-from-motion with featuremetric refinement. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5987–5997, 2021. 1

[44] Hongchen Luo, Wei Zhai, Jing Zhang, Yang Cao, and Dacheng Tao. Learning affordance grounding from exocentric images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2252–2261, 2022. 2

[45] Tiange Luo, Chris Rockwell, Honglak Lee, and Justin Johnson. Scalable 3d captioning with pretrained models. *Advances in Neural Information Processing Systems*, 36, 2024. 5

[46] Jinjie Mai, Wenxuan Zhu, Sara Rojas, Jesus Zarzar, Abdullah Hamdi, Guocheng Qian, Bing Li, Silvio Giancola, and Bernard Ghanem. Tracknerf: Bundle adjusting nerf from sparse and noisy views via feature tracks. *arXiv preprint arXiv:2408.10739*, 2024. 1

[47] Juhong Min, Jongmin Lee, Jean Ponce, and Minsu Cho. Spair-71k: A large-scale benchmark for semantic correspondence. *arXiv preprint arXiv:1908.10543*, 2019. 2, 6

[48] Arsalan Mousavian, Clemens Eppner, and Dieter Fox. 6-dof graspnet: Variational grasp generation for object manipulation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2901–2910, 2019. 2

[49] Adithyavairavan Murali, Weiyu Liu, Kenneth Marino, Sonia Chernova, and Abhinav Gupta. Same object, different grasps: Data and semantic knowledge for task-oriented grasping. In *Conference on robot learning*, pages 1540–1557. PMLR, 2021. 2, 4

[50] Austin Myers, Ching L Teo, Cornelia Fermüller, and Yiannis Aloimonos. Affordance detection of tool parts from geometric features. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1374–1381. IEEE, 2015. 2

[51] Tushar Nagarajan, Christoph Feichtenhofer, and Kristen Grauman. Grounded human-object interaction hotspots from video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8688–8697, 2019. 2

[52] Anh Nguyen, Dimitrios Kanoulas, Darwin G. Caldwell, and Nikos G. Tsagarakis. Object-based affordances detection with convolutional neural networks and dense conditional random fields. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5908–5915, 2017. 2

[53] OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. Gpt-4 technical report, 2024. 4

[54] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 1, 2, 3, 5, 7

[55] Norman Di Palo and Edward Johns. Dinobot: Robot manipulation via retrieval and alignment with vision foundation models. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2024. 1, 3

[56] Georgios Papagiannis and Edward Johns. Miles: Making imitation learning easy with self-supervision. In *Proceedings of the Conference on Robot Learning (CoRL)*, 2024. 3

[57] Shengyi Qian, Weifeng Chen, Min Bai, Xiong Zhou, Zhuowen Tu, and Li Erran Li. Affordancellm: Grounding affordance from vision language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7587–7597, 2024. 3

[58] Zengyi Qin, Kuan Fang, Yuke Zhu, Li Fei-Fei, and Silvio Savarese. Keto: Learning keypoint representations for tool manipulation. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7278–7285. IEEE, 2020. 2

[59] 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. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 3, 5

[60] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020. 3

[61] Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10901–10911, 2021. 4

[62] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 1, 3, 7

[63] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4938–4947, 2020. 2

[64] Stefan Stojanov, Anh Thai, Zixuan Huang, and James M Rehg. Learning dense object descriptors from multiple views for low-shot category generalization. *Advances in Neural Information Processing Systems*, 35:12566–12580, 2022. 2

[65] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. Loftr: Detector-free local feature matching with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8922–8931, 2021. 2

[66] Yixuan Sun, Yiwen Huang, Haijing Guo, Yuzhou Zhao, Runmin Wu, Yizhou Yu, Weifeng Ge, and Wenqiang Zhang. Misc210k: A large-scale dataset for multi-instance semantic correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7121–7130, 2023. 2, 6

[67] Luming Tang, Menglin Jia, Qianqian Wang, Cheng Perng Phoo, and Bharath Hariharan. Emergent correspondence from image diffusion. *Advances in Neural Information Processing Systems*, 36:1363–1389, 2023. 3

[68] Tatsunori Taniai, Sudipta N Sinha, and Yoichi Sato. Joint recovery of dense correspondence and cosegmentation in two images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4246–4255, 2016. 1

[69] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 402–419. Springer, 2020. 2

[70] Skye Thompson, Leslie Pack Kaelbling, and Tomas Lozano-Perez. Shape-based transfer of generic skills. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5996–6002. IEEE, 2021. 3

[71] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 3

[72] Prune Truong, Marie-Julie Rakotosaona, Fabian Manhardt, and Federico Tombari. Sparf: Neural radiance fields from sparse and noisy poses. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4190–4200, 2023. 1

[73] Dylan Turpin, Liquan Wang, Stavros Tsogkas, Sven Dickinson, and Animesh Garg. Gift: Generalizable interaction-aware functional tool affordances without labels. *arXiv preprint arXiv:2106.14973*, 2021. 2

[74] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding, 2019. 6

[75] Raviteja Vemulapalli, Hadi Pouransari, Fartash Faghri, Sachin Mehta, Mehrdad Farajtabar, Mohammad Rastegari, and Oncel Tuzel. Knowledge transfer from vision foundation models for efficient training of small task-specific models. In *Forty-first International Conference on Machine Learning*, 2024. 5

[76] Boyan Wan, Yifei Shi, and Kai Xu. Socs: Semantically-aware object coordinate space for category-level 6d object pose estimation under large shape variations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14065–14074, 2023. 2

[77] He Wang, Srinath Sridhar, Jingwei Huang, Julien Valentin, Shuran Song, and Leonidas J Guibas. Normalized object coordinate space for category-level 6d object pose and size estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2642–2651, 2019. 1, 2

[78] Weihan Wang, Qingsong Lv, Wenmeng Yu, Wenyi Hong, Ji Qi, Yan Wang, Junhui Ji, Zhuoyi Yang, Lei Zhao, Xixuan Song, et al. Cogvlm: Visual expert for pretrained language models. *arXiv preprint arXiv:2311.03079*, 2023. 1, 2, 3, 4, 5, 7

[79] Elizabeth A Ware and Amy E Booth. Form follows function: Learning about function helps children learn about shape. *Cognitive Development*, 25(2):124–137, 2010. 3

[80] Tong Wu, Jiarui Zhang, Xiao Fu, Yuxin Wang, Liang Pan Jiawei Ren, Wayne Wu, Lei Yang, Jiaqi Wang, Chen Qian, Dahua Lin, and Ziwei Liu. Omniobject3d: Large-vocabulary 3d object dataset for realistic perception, reconstruction and generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 4

[81] Ruinian Xu, Fu-Jen Chu, Chao Tang, Weiyu Liu, and Patricio A Vela. An affordance keypoint detection network for robot manipulation. *IEEE Robotics and Automation Letters*, 6(2):2870–2877, 2021. 2

[82] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10371–10381, 2024. 5

[83] Yuhang Yang, Wei Zhai, Hongchen Luo, Yang Cao, Jiebo Luo, and Zheng-Jun Zha. Grounding 3d object affordance

from 2d interactions in images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10905–10915, 2023. 2

[84] Xianggang Yu, Mutian Xu, Yidan Zhang, Haolin Liu, Chongjie Ye, Yushuang Wu, Zizheng Yan, Chenming Zhu, Zhangyang Xiong, Tianyou Liang, et al. Mvimgnet: A large-scale dataset of multi-view images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9150–9161, 2023. 4

[85] Wentao Yuan, Jiafei Duan, Valts Blukis, Wilbert Pumacay, Ranjay Krishna, Adithyavairavan Murali, Arsalan Mousavian, and Dieter Fox. Robopoint: A vision-language model for spatial affordance prediction for robotics. *arXiv preprint arXiv:2406.10721*, 2024. 3

[86] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11975–11986, 2023. 3

[87] Junyi Zhang, Charles Herrmann, Junhwa Hur, Luisa Polania Cabrera, Varun Jampani, Deqing Sun, and Ming-Hsuan Yang. A tale of two features: Stable diffusion complements dino for zero-shot semantic correspondence, 2023. 1, 7

[88] Chenchen Zhu, Fanyi Xiao, Andrés Alvarado, Yasmine Babaei, Jiabo Hu, Hichem El-Mohri, Sean Culatana, Roshan Sumbaly, and Zhicheng Yan. Egoobjects: A large-scale egocentric dataset for fine-grained object understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20110–20120, 2023. 4

[89] Yixin Zhu, Yibiao Zhao, and Song Chun Zhu. Understanding tools: Task-oriented object modeling, learning and recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2855–2864, 2015. 3

[90] Yifeng Zhu, Arisrei Lim, Peter Stone, and Yuke Zhu. Vision-based manipulation from single human video with open-world object graphs. *arXiv preprint arXiv:2405.20321*, 2024. 3