Appendix for "UNOPose: Unseen Object Pose Estimation with an Unposed RGB-D Reference Image"

Xingyu Liu[1,*], Gu Wang[1,*], Ruida Zhang[1], Chenyangguang Zhang[1], Federico Tombari[2,3], and Xiangyang Ji[1]
[1]Tsinghua University, [2]Technical University of Munich, [3]Google
{liuxy21@mails., wanggu1@, zhangrd23@mails., zcyg22@mails., xyji@}tsinghua.edu.cn, tombari@in.tum.de

Contents

A. Additional Details of Method

A.1. UNO Object Segmentation

UNOSeg is constructed with three steps, *i.e.*, mask proposal generation, global and local matching, and mask proposal assignment.

Mask Proposal Generation. Given the query image I_q and uniformly sampled pixel positions P as prompt, the segment anything model (SAM) [8, 19] Θ predicts N_m mask proposals $\mathcal{M}$ with confidence scores $\mathcal{C}$, denoted by

$$\mathcal{M}, \mathcal{C} = \Theta(I_q, P). \tag{A-1}$$

We discard low-confidence predictions and apply Non-Maximum Suppression to filter duplicate proposals.

*Equal contributions.

Global and Local Matching. In the matching stage, the network assigns each mask proposal a similarity score w.r.t. the reference view. Specifically, given the reference image, we first remove the background using M_p, then crop the region of interest and resize it to $\hat{I}_p \in \mathbb{R}^{224\times224\times3}$. Concurrently, we crop and resize the target image with all mask proposals to a consistent size $\{\hat{I}_q^j \in \mathbb{R}^{224\times224\times3}\}_{j=1}^{N_m}$. $\hat{I}_q$ and $\hat{I}_p$ are fed into a pre-trained DINOv2 model [5, 13] to generate image-level global descriptors $\hat{\mathcal{G}}_q, \hat{\mathcal{G}}_p$ and N_l patch-level local descriptors $\{\hat{\mathcal{L}}_q^k\}_{k=1}^{N_l}, \{\hat{\mathcal{L}}_p^k\}_{k=1}^{N_l}$. By evaluating the cosine similarity of descriptors, the matching score ξ can be obtained as

$$\xi = (\xi_{\mathcal{G}} + \xi_{\mathcal{L}})/2, \tag{A-2}$$

where $\xi_{\mathcal{G}}$ and $\xi_{\mathcal{L}}$ are global and local descriptor similarities calculated by

$$\begin{aligned} \xi_{\mathcal{G}} &= \frac{\hat{\mathcal{G}}_q^\top \hat{\mathcal{G}}_p}{\|\hat{\mathcal{G}}_q\|_2 \cdot \|\hat{\mathcal{G}}_p\|_2}, \\ \xi_{\mathcal{L}} &= \frac{1}{N_l}\sum_{k=1}^{N_l} \max_{i=1,\dots,N_l} \frac{(\hat{\mathcal{L}}_q^k)^\top \hat{\mathcal{L}}_p^i}{\|\hat{\mathcal{L}}_q^k\|_2 \cdot \|\hat{\mathcal{L}}_p^i\|_2}. \end{aligned} \tag{A-3}$$

Leveraging both global and local matching scores, the network effectively distinguishes the mask proposal most similar to M_q relative to the reference object.

Mask Proposal Assignment. A single query image may contain multiple distinct query objects in the test scenario. In this scenario, we generate mask proposals using Eq. (A-1) once. For each individual mask proposal, we calculate its similarity score against every candidate reference and select the highest score to determine the object class of this mask proposal. Noteworthy, while the whole image may have multiple reference images, each distinct target object only has a single reference image.

A.2. Details of Eq. (5) in GRF Construction

Here we provide a detailed definition of $\mathbf{r}_{Gx}$ in the construction of GRF (*c.f.* Eq. (5) in the main text). Considering the vector from the center $\mathbf{c}_Q$ to a point $\mathbf{q} \in \mathbf{Q}_{cam}$, given the normal vector $\mathbf{r}_{Gz}$ at $\mathbf{c}_Q$, its projected vector $\mathbf{v}_\mathbf{q}$ on the tangent plane of $\mathbf{c}_Q$ can be computed as

$$\mathbf{v}_\mathbf{q} = (\mathbf{q} - \mathbf{c}_Q) - \mathbf{r}_{Gz}^\top(\mathbf{q} - \mathbf{c}_Q)\mathbf{r}_{Gz}. \tag{A-4}$$

Then, the projected vectors of all points are aggregated by summing them, with each vector being weighted according to the distance between $\mathbf{q}$ and $\mathbf{c}_Q$. We decrease the weight of points that are farther from the center, as they are more likely to be outliers. Specifically, $\mathbf{r}_{Gx}$ is calculated as

$$\begin{aligned} \mathbf{r}_{Gx} &= \frac{\sum_{\mathbf{q}\in\mathbf{Q}_{cam}} w_{q,1}w_{q,2}\mathbf{v}_\mathbf{q}}{\|\sum_{\mathbf{q}\in\mathbf{Q}_{cam}} w_{q,1}w_{q,2}\mathbf{v}_\mathbf{q}\|_2}, \\ w_{q,1} &= (s_Q - \|\mathbf{q} - \mathbf{c}_Q\|_2)^2, \\ w_{q,2} &= (\mathbf{r}_{Gz}^\top(\mathbf{q} - \mathbf{c}_Q))^2, \\ s_Q &= \max_{\mathbf{q}\in\mathbf{Q}_{cam}} \|\mathbf{q} - \mathbf{c}_Q\|_2. \end{aligned} \tag{A-5}$$

The weighting factor w_q in Eq. (5) can thus be expressed as

$$w_q = \frac{w_{q,1}w_{q,2}}{\|\sum_{\mathbf{q}\in\mathbf{Q}_{cam}} w_{q,1}w_{q,2}\mathbf{v}_\mathbf{q}\|_2}. \tag{A-6}$$

A.3. Details of Training Objectives

We leverage the InfoNCE [12] loss to constrain the learning of the correlation matrix and the weighted binary cross-entropy (WBCE) loss for supervising overlap prediction. In specific, given the correlation matrix $\mathbf{X}^c \in \mathbb{R}^{(N^c+1)\times(N^c+1)}$ denoting the predicted correspondence between the query point cloud $\mathbf{Q}_G^c$ and the reference point cloud $\mathbf{P}_G^c$, it is supervised with

$$\mathcal{L}_X^c = \text{CE}(\mathbf{X}^c[1:,:], \bar{y}_q) + \text{CE}(\mathbf{X}^c[:,1:]^\top, \bar{y}_p). \tag{A-7}$$

Here $\bar{y}_q \in \mathbb{R}^{N^c}$ and $\bar{y}_p \in \mathbb{R}^{N^c}$ are the ground-truth correspondence for $\mathbf{Q}_G^c$ and $\mathbf{P}_G^c$.

The overlap predictions $\hat{O}_Q^c \in \mathbb{R}^{(N^c+1)\times 1}$, and $\hat{O}_P^c \in \mathbb{R}^{(N^c+1)\times 1}$ are supervised with

$$\mathcal{L}_O^c = \text{WBCE}(\hat{O}_Q^c, \bar{O}_Q^c) + \text{WBCE}(\hat{O}_P^c, \bar{O}_P^c), \tag{A-8}$$

where $\bar{O}_Q^c$, and $\bar{O}_P^c$ are overlap labels for the query and reference point clouds, respectively. For each point, its ground-truth overlap $\bar{o}_i^c, i \in \{1, ..., N^c + 1\}$ is calculated by

$$\bar{o}_i^c = \begin{cases} 0, & \text{if } i = 1 \text{ or } d_{i,\min} > \delta, \\ 1, & \text{otherwise.} \end{cases} \tag{A-9}$$

For the background token ($i = 1$), the ground-truth overlap score is constant at 0. Moreover, $d_{i,\min}$ is the distance of the i-th point to the closest point in the counterpart point cloud under ground-truth transformation, and δ is a hyper-parameter which we set to 0.15.

Similarly, we calculated $\mathcal{L}_X^f$ and $\mathcal{L}_O^f$ for supervising the fine point matching procedure.

To sum up, the overall learning objective can be written as

$$\mathcal{L} = \sum_{t\in\{1,2,3\}} (\mathcal{L}_X^{c,t} + \mathcal{L}_O^{c,t}) + \sum_{t\in\{1,2,3\}} (\mathcal{L}_X^{f,t} + \mathcal{L}_O^{f,t}). \tag{A-10}$$

Here t denotes the block index of the geometric transformer decoder.

A.4. Details of Hyper-parameters

We empirically set the number of coarse point samples N^c to 196, and the number of fine point samples N^f to 2048. We sample $N_H = 300$ pose hypotheses in coarse pose prediction. In building the local reference frame, the number of neighborhoods N_D is set to 64. The dimensions for geometric embedding, color embedding, LRF encoding, and positional encoding are set to 256.

A.5. Discussion on Handling Symmetric Objects

Currently, UNOPose does not explicitly handle symmetry. However, the use of DINOv2 [13] visual features can handle geometrically symmetric but visually distinguishable objects. Explicitly and effectively handling symmetry holds promise as a valuable direction for future exploration.

B. More Experimental Results

B.1. Comparison with the Open-vocabulary Setting on TYO-L

Oryon [4], which proposes an open-vocabulary setting for object pose estimation, shares similar settings with UNOPose but differs in object segmentation and the pipeline focus. Specifically, Oryon focuses on segmentation given potentially vague textual prompts and employs an off-the-shelf method PointDSC [1] for pose estimation. In contrast, UNOPose leverages the GT reference mask to directly identify the target object and focuses on developing a novel method and a BOP-based standard benchmark for unseen one-reference-based pose estimation.

Method	Segmentation	VSD	MSSD	MSPD	AR_{BOP}
Oryon	Pred	12.1	37.5	41.4	30.3
Ours	UNOSeg	**69.6**	**79.1**	**81.7**	**76.8**
Oryon	GT	13.9	42.9	45.5	34.1
Ours	GT	**73.5**	**81.7**	**84.4**	**79.9**

Table B-1. **Comparison with Oryon on TYO-L.**

Figure B-1. **Qualitative results in real-world phone-captured scenarios.** The first image is the reference for this row. Ground-truth and predicted poses are colored with yellow and blue accordingly.

Method	Modality	In-dataset fine-tuning	Angular error (°)↓	Acc_{30} (%) ↑	Acc_{15} (%)↑
RelPose [16]	RGB	✓	58.3	26.1	7.0
RelPose++ [9]	RGB	✓	46.6	42.5	15.8
3DAHV [17]	RGB	✓	41.7	61.5	29.9
DVMNet [18]	RGB	✓	36.8	-	-
UNOPose*	RGB	✗	49.1	50.0	19.1
UNOPose	RGB-D	✗	**23.9**	**84.2**	**81.1**

Table B-2. **Comparison with RGB-based relative pose estimation methods on LM.** The depth of UNOPose* is predicted by ZoeDepth [2].

To compare UNOPose with Oryon, we test on Toyota Light [7] (TYO-L) with Oryon's image pairs and present results in Tab. B-1. This BOP dataset includes 21 objects captured under diverse lighting conditions and backgrounds. Tab. B-1 demonstrates the clear superiority of UNOPose with both GT and predicted segmentations.

B.2. Comparison with RGB-based Relative Pose Estimation Methods on LM

Although the setting of UNOPose is different from that of RGB-based relative pose estimation methods [9, 16–18], we include a comparison of their test split of the LM dataset [6] in Tab. B-2. We report the rotation error and the accuracy with thresholds of 30° and 15°. For comparing our UNOPose with them under RGB modality, we use ZoeDepth [2] to predict metric depth from monocular data. It is shown that our method achieves comparable results with relative 3DoF pose estimators even under RGB modality. Moreover, when using ground-truth depth, our approach largely surpasses in terms of rotation metrics. While leveraging more advanced monocular metric depth estimation techniques could potentially yield better results, it is beyond the scope of this paper, and we leave it for future work. Additionally, we can estimate translation besides rotation. Note that, for evaluating on LM, they need to first train on the synthetic data and then perform in-dataset fine-tuning on the real LM data by excluding the test objects. However, UNOPose is only trained on the synthetic dataset and then tested on several real-world datasets.

B.3. Results on Real-world Scenarios

To verify the practicality of UNOPose, we adopt it in real-world phone-captured scenarios and present qualitative results in Fig. B-1. Specifically, we run UNOPose on three sequences of daily objects (banana, detergent, and kettle) provided by the WildRGB-D dataset [14]. This dataset is

Method	VSD	MSSD	MSPD	AR_{BOP}
Ref. AlignCenter (baseline)	31.7	43.4	19.5	31.5
UNOPose	69.2	74.8	63.9	69.3

Table B-3. **Reference from different scenes on YCB-V.**

Image ID	VSD	MSSD	MSPD	AR_{BOP}
751	50.4	74.0	69.5	64.6
3895	51.0	74.7	70.5	65.4
4329	56.1	75.2	72.4	67.9
5274	56.3	76.1	73.5	68.6
7268	54.7	76.7	73.2	68.2
10741	47.1	73.2	69.4	63.2
11430	45.8	67.3	63.4	58.8
9734	36.3	57.3	52.1	48.5

Table B-4. **Ablation of the reference selection on "One Reference per Category" experiments on the "Dragon" object of TUD-L.** 10741 is the image ID we chose in the paper.

photographed in several in-the-wild scenarios by the front camera of an iPhone. For each sequence, one frame is chosen as the reference, and the rest frames are treated as queries. The SAM-based masks provided by the dataset are directly employed. For visualization, we use the 3D bounding box derived from the reconstructed object and the provided absolute camera poses. Note that our UNOPose requires no information from the 3D model or the camera poses. Qualitative results show that UNOPose can adjust well to in-the-wild scenarios, daily objects, varying lighting conditions, low-quality depth, and occasional occlusions.

B.4. More Ablation studies

Reference from Different Scenes. Compared to previous methods, UNOPose cuts objects out of the background, eliminating reliance on scene context. Our approach allows reference images to originate from diverse scenes and be captured by different cameras. We additionally present the evaluation results for all references from different scenes on YCB-V [15] and show the results in Tab. B-3. Compared to the baseline, UNOPose achieves a notable 37.8% improvement in accuracy, demonstrating its robust adaptability across changing scenes.

Selection of Single Reference. In the main text, we conduct the "One Reference for a Category" experiment. To explore the effect of reference selection on the results, we further conduct ablations on the dragon object from the TUD-L dataset [7]. Specifically, we sampled 8 different viewpoints from TUD-L, ensuring a rotation difference of at least 40 degrees between any two viewpoints. This sampling strategy allowed the references to cover the object's full range

Num. Ref.	VSD	MSSD	MSPD	AR_{BOP}
1	54.4	63.7	64.1	60.7
8	53.4	65.9	65.6	61.6
12	61.3	74.2	74.0	69.8
16	62.5	74.3	75.4	70.7
32	**62.8**	**76.2**	**75.7**	**71.6**
42	60.7	75.0	74.8	70.0
Random	61.9	75.2	75.9	71.0

Table B-5. **Ablation on the total number of references in a dataset (TUD-L).** We randomly select a certain number of references for each category on the TUD-L training set (denoted as Num. Ref.). "Random" is the setting in main experiments, indicating randomly selecting a reference for each query object. The reference selection policy is consistent with the main text.

Method	SAM-Real	SAM-Ren	FastSAM-Real	FastSAM-Ren
AP_{Avg}	54.2	51.7	51.6	49.5

Table B-6. **Ablation on segmentation with rendered reference images.** We report the average mAP on LM-O, YCB-V, and TUD-L.

of perspectives. The further "one reference per category" experimental results are shown in Fig. B-2 and Tab. B-4. It is shown that image IDs 751, 3895, 4329, 5274, and 7268 achieve better results than the viewpoint we chose in the paper (image ID 10741, AR_{BOP} 63.2). However, when using 11430 or 9734 as the only reference image of the category, they gain sub-optimal results. Qualitative and quantitative results show that the reference with brighter illumination and larger visible regions can lead to better results.

Impact of Reference Number. We further dig into the influence of references' numbers on the whole dataset and present the results in Tab. B-5. The number of references (Num. Ref.) that equals 1 corresponds to the experiment in Tab. 3 of the main text, and "random" indicates randomly selecting a reference for each query object. Note that *UNOPose still runs with only one reference for each query*. We find that the best performance is achieved with 32 references and competitive results can be observed after the total reference number of each object increases to 12. Impressively, even with just one reference for each object in the entire dataset, the results remain excellent (comparable to using 8 references). It indicates that UNOPose does not rely on multiple references at the dataset level.

Segmentation with Rendered References. In the main experiments, we employ real-world reference images for segmentation comparison, which exhibits a smaller domain gap than compared methods [3, 10, 11]. However, we

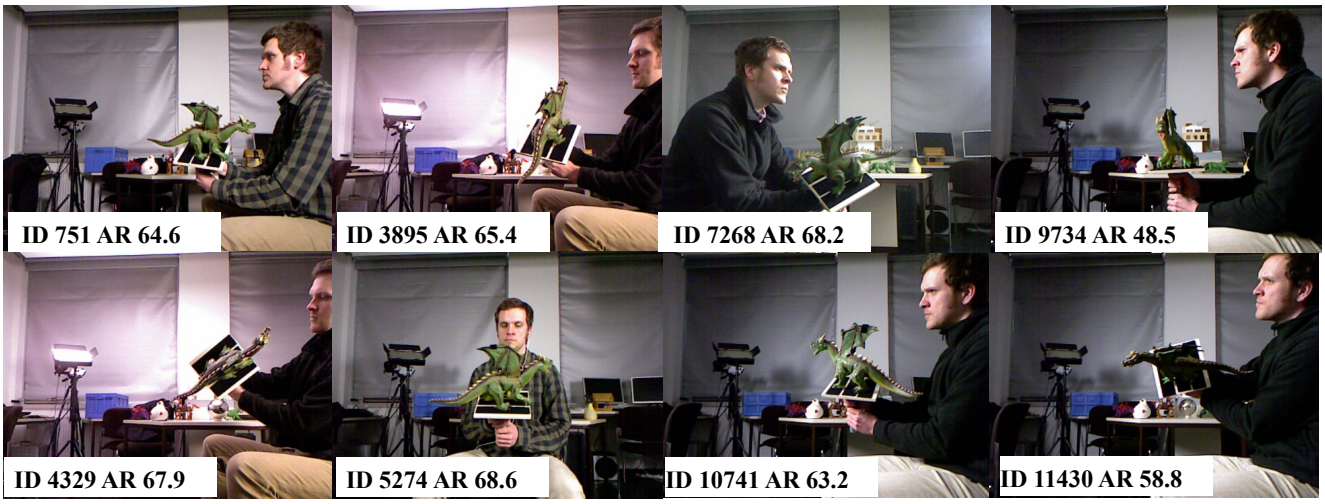

Figure B-2. **Different reference selection experiment on TUD-L.** Each image is chosen as a reference for all test data in the dragon category. The results w.r.t. AR_{BOP} metric are shown in the picture.

aim to show that our one-reference UNOSeg is competitive with model-based multi-reference methods, not to set a new SOTA. We opt to use a real reference image instead of a rendered one, as the latter would require a 3D model. For fair pose estimation comparison, we use consistent segmentation results from UNOSeg. Replacing the real reference with the rendered version slightly lowers UNOSeg's performance (See Tab. B-6), but the absence of template retrieval makes our results reasonable compared to CAD-based methods.

B.5. Qualitative Results

We show some qualitative results on YCB-V, LM-O, and TUD-L datasets in Fig. B-3. Despite occlusion, sensor noise, and varying testing scenarios, our generic approach achieves robust unseen object pose estimation results with one single reference image.

C. Potential Positive and Negative Societal Impacts

We developed UNOPose for estimating unseen object poses given a single RGB-D reference image. By avoiding the need for retraining and reducing the cost of creating references for each new object, UNOPose can not only reduce the environmental burden but also offer more application possibilities. This technology has broad applications in industrial manufacturing and robotic manipulation. While it may lead to job displacement due to increased automation, we aim for this work to have a positive societal impact by enhancing efficiency and safety.

References

[1] Xuyang Bai, Zixin Luo, Lei Zhou, Hongkai Chen, Lei Li, Zeyu Hu, Hongbo Fu, and Chiew-Lan Tai. Pointdsc: Robust point cloud registration using deep spatial consistency. In *CVPR*, pages 15859–15869, 2021. 2

[2] Shariq Farooq Bhat, Reiner Birkl, Diana Wofk, Peter Wonka, and Matthias Müller. Zoedepth: Zero-shot transfer by combining relative and metric depth. *arXiv preprint arXiv:2302.12288*, 2023. 3

[3] Jianqiu Chen, Zikun Zhou, Mingshan Sun, Rui Zhao, Liwei Wu, Tianpeng Bao, and Zhenyu He. Zeropose: Cad-prompted zero-shot object 6d pose estimation in cluttered scenes. *IEEE TCSVT*, 2024. 4

[4] Jaime Corsetti, Davide Boscaini, Changjae Oh, Andrea Cavallaro, and Fabio Poiesi. Open-vocabulary object 6d pose estimation. 2024. 2

[5] Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *ICLR*, 2024. 1

[6] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, , and N. Navab. Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In *ACCV*, 2012. 3

[7] Tomas Hodan, Frank Michel, Eric Brachmann, Wadim Kehl, Anders GlentBuch, Dirk Kraft, Bertram Drost, Joel Vidal, Stephan Ihrke, Xenophon Zabulis, et al. BOP: Benchmark for 6D Object Pose Estimation. In *ECCV*, pages 19–34, 2018. 3, 4

[8] 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 *ICCV*, pages 4015–4026, 2023. 1

[9] Amy Lin, Jason Y Zhang, Deva Ramanan, and Shubham Tulsiani. Relpose++: Recovering 6d poses from sparse-view observations. In *3DV*, 2024. 3

Figure B-3. Qualitative Results on YCB-V, LM-O and TUD-L datasets. We illustrate the ground-truth and predicted poses of query objects with 3D bounding boxes.

[10] Jiehong Lin, Lihua Liu, Dekun Lu, and Kui Jia. Sam-6d: Segment anything model meets zero-shot 6d object pose estimation. In *CVPR*, 2024. 4

[11] Van Nguyen Nguyen, Thibault Groueix, Georgy Ponimatkin, Vincent Lepetit, and Tomas Hodan. Cnos: A strong baseline for cad-based novel object segmentation. In *ICCV*, pages 2134–2140, 2023. 4

[12] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 2

[13] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *TMLR*, 2024. 1, 2

[14] Hongchi Xia, Yang Fu, Sifei Liu, and Xiaolong Wang. Rgbd

objects in the wild: Scaling real-world 3d object learning from rgb-d videos. In *CVPR*, pages 22378–22389, 2024. 3

[15] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. PoseCNN: A convolutional neural network for 6D object pose estimation in cluttered scenes. *RSS*, 2018. 4

[16] Jason Y Zhang, Deva Ramanan, and Shubham Tulsiani. Relpose: Predicting probabilistic relative rotation for single objects in the wild. In *ECCV*, pages 592–611. Springer, 2022. 3

[17] Chen Zhao, Tong Zhang, and Mathieu Salzmann. 3d-aware hypothesis & verification for generalizable relative object pose estimation. In *ICLR*, 2023. 3

[18] Chen Zhao, Tong Zhang, Zheng Dang, and Mathieu Salzmann. Dvmnet: Computing relative pose for unseen objects beyond hypotheses. In *CVPR*, 2024. 3

[19] Xu Zhao, Wenchao Ding, Yongqi An, Yinglong Du, Tao Yu, Min Li, Ming Tang, and Jinqiao Wang. Fast segment anything. *arXiv preprint arXiv:2306.12156*, 2023. 1