SUPPLEMENTARY FOR

Bongard-HOI: Benchmarking Few-Shot Visual Reasoning for Human-Object Interactions

Huaizu Jiang[1*], Xiaojian Ma[2*], Weili Nie[3] , Zhiding Yu[3] , Yuke Zhu[3,4], Anima Anandkumar[3,5]
[1]Northeastern University [2]UCLA [3]NVIDIA [4]UT Austin [5]Caltech
h.jiang@northeastern.edu, xiaojian.ma@ucla.edu, {wnie,zhidingy}@nvidia.com,
yukez@cs.utexas.edu, anima@caltech.edu
nvlabs.github.io/Bongard-HOI

A. Limitation Statement

We re-use the images collected by the HAKE [1] creators, including the ones for HICO [1], V-COCO [2], OpenImages [4], HCVRD [7], and PIC [6], which were crawled from the web. Except the images, in this paper, no identity related information were collected nor used when constructing the dataset and benchmarking other approaches. It is possible, however, that some person may be identified via facial recognition techniques. We will provide contact information of the benchmark maintainer and commit to processing request of removing some certain images from the dataset. In addition, similar to other human-centric dataset, the images we use are from just a small portion of the population, which may contain biases toward some certain races, gender, ethnic groups, etc. We are unable to measure the bias as we do not have any identity-related data. We encourage researchers to investigate such issues.

B. More details on the Bongard-HOI Benchmark

B.1. Constructing Bongard Problems

Given positive images $\mathcal{I}_c$ that depict a certain relationship $c = \langle s, a, o \rangle$ and and negative images $\mathcal{I}_{\bar{c}}$ that does not, we need to sample few-shot instances from them. We randomly sample images to form $\mathcal{P}$, $\mathcal{N}$, and a query image I_q. Two parameters control the sampling process: M, the number of images in $\mathcal{P}$ and $\mathcal{N}$ ($M = 6$ in Bongard-HOI), and the overlap threshold τ, indicating the maximum number of overlapped images between two few-shot instances. We want to sample as many few-shot instances as possible, but we also need to avoid significant image overlap between few-shot instances, which limits the diversity of the data. The sampling process in summarized in Algorithm 1. We set $\tau = 3$ and $\tau = 2$ for training and test sets, respectively.

Algorithm 1: Sample few-shot instances for a visual concept c

Input: Positive images $\mathcal{I}_c$, negative images $\mathcal{I}_{\bar{c}}$, number of images in a few-shot instance M, overlap threshold τ.
Output: Sampled few-shot instances $\mathcal{Q}$.

```
Q = ∅;
while True do
    P^i, N^i, I_q^i = sample_instance(I_c, I_c̄, M);
    if sample fails then
        break;
    t = overlap(P^i, N^i, I_q^i, Q);
    if t < τ then
        Q = Q ∪ (P^i, N^i, I_q^i);
```

B.2. Data Curation

Although the HAKE dataset [5] has provided high-quality annotations, we found that curations are still needed to construct the Bongard problems (few-shot instances) for our Bongard-HOI benchmark. Recall, to sample negative images, we assume a particular action is not depicted in them. In HAKE, an image region may have multiple action labels. Naively relying on the provided annotations is problematic as the action labels are either not manually exclusive or not exhaustively annotated. We show different cases of data curations in Fig. 1 and discuss them in details as follows.

Similar actions. Although some action labels may convey different semantic meanings, for some certain object categories, they look visually similar and indistinguishable. As shown in Fig. 1(a), `scratch cat` and `pet cat` are hard to differentiate visually. If we simply use images of

*First two authors contributed equally.

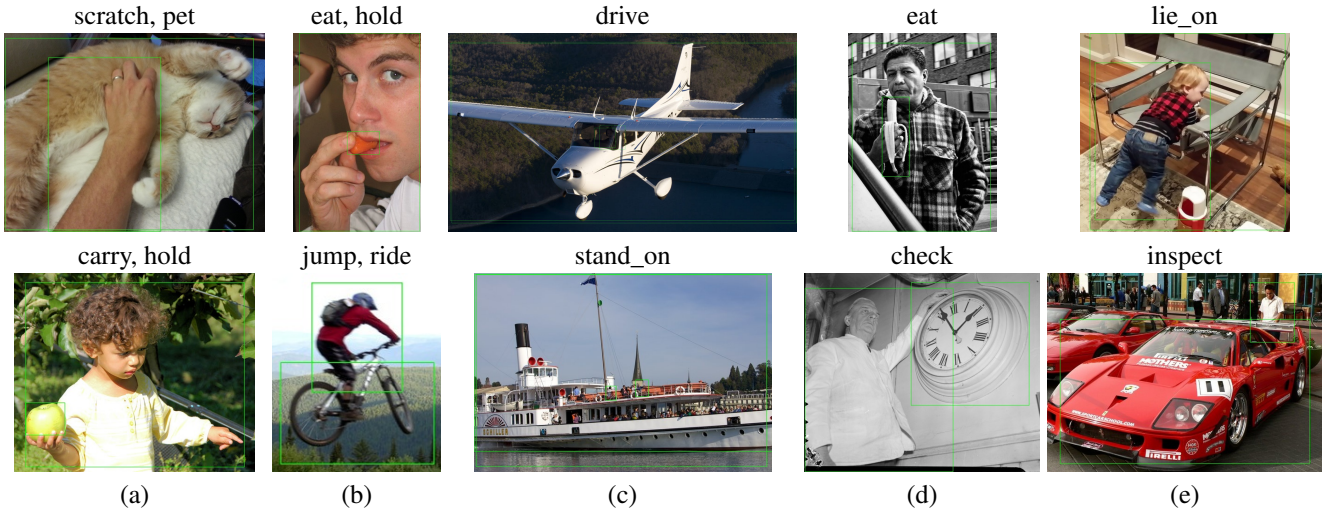


Figure 1. **Samples of annotations where curations are needed.** For each image region, its annotated action labels are show on its top and bounding boxes corresponding to the person and object are shown for visualization purpose. From left to right: (a) similar actions, (b) hierarchical annotations, (c) hard-to-see objects, (d) extrapolating annotations, and (e) inaccurate or confusing annotations.

`scratch cat` as negatives to construct few-shot instances for `pet cat`, such few-shot instances are ambiguous, as it violates the basic assumption that the visual concept depicted in the Set $\mathcal{A}$ is not available in the Set $\mathcal{B}$. We therefore simply merge such similar action labels to reduce the visual ambiguity.

Hierarchical actions. Action labels are inherently hierarchical. For example, as shown in Fig. 1(b), `eat carrot` very likely also means `hold carrot` visually. There are two problems to construct few-shot instances with multiple hierarchical action labels associated with the same image region. First of all, as we previously explained, using images of `eat carrot` as negatives for `hold carrot` may cause ambiguity. More importantly, there is the *visual specificity* issue. People tend to focus on capturing the most salient actions in an image, which are usually the parent actions (`eat carrot` in this case). In our preliminary experiments, images of `eat carrot` were used as positives for `hold carrot` to construct few-shot instances. We found that it caused a lot of confusion for human testers. To this end, we merge such hierarchical action labels for the same region, keeping the parent action labels only.

Hard-to-see objects. In some cases, the person or the objects in image regions are hard to see. For example, in Fig. 1(c), the person with the action label `stand_on boat` is hard to see clearly. On the one hand, it causes significant challenges for a visual perception system (*e.g.*, [3]) to accurately localize the meaningful objects. At the same time, it also imposes difficulty for annotators to accurately annotate the image region. We simply discard all image regions with hard-to-see objects.

Extrapolating actions. Actions are continuous. As a result, annotators tend to *extrapolate* the action label given a single image, instead of describing the current state the action. For example, as we can see in the top row of Fig. 1(d), the `eat` action is about to happen. Yet, the action is different from a normal `hold banana` without any indication of `eat`. To distinguish different scenarios, we introduce `hold_not_about_to_eat banana`, `hold_and_about_to_eat banana`, and `eat banana`. In this way, all the actions are mutually exclusive. We can sample image regions for form few-shot instances without worrying about causing ambiguity.

Inaccurate or confusing actions. In some rare cases, the annotations in HAKE are inaccurate or confusing, as shown in Fig. 1(e). We modify the action labels if such a image region depicts a clear action label. Otherwise we discard such regions to avoid introducing ambiguity to sampled few-shot instances.

MTurk data curation. After performing the aforementioned data curations, each image region is assigned to a single action label, describing the most salient content. Such action labels are mutually exclusive so that we can significantly reduce the ambiguity when constructing few-shot instances. Finally, we hire high-quality testers on the Amazon Mechanical Turk (MTurk) platform, who maintain a good job approval record, to curate the testing set to further remove the ambiguous few-shot instances. Every single BP is assigned to three independent testers. We compare their responses with the ground-truth labels and disard about 2.5% few-shot instances where none of the three testers correctly classifies the query images. We provide more details of the MTurk curations in Section D.

Figure 2. **Illustration of the context-dependent reasoning property of the Bongard problems (few-shot instances) in our Bongard-HOI benchmark.** Two instances are shown here with their underlying visual concepts (relationships) displayed on top with red color. The same query image receives two different labels (negative in the top and positive in the bottom) among different context (*i.e.*, positive and negative examples).

	seen object	unseen object
seen action	99 / 5008	36 / 5002
unseen action	20 / 3402	12 / 3775

(a) validation set

	seen object	unseen object
seen action	102 / 4476	27 / 4562
unseen action	21 / 3291	16 / 1612

(b) test set

Table 1. **Number of concepts and few-shot instances in the validation and test sets.** Depending on whether an action and object is seen during the training, we divide the validation and test sets into four categories, where we can study the systematic generalization of machine learning models. For each category, we show number of concepts (combinations of action and object) and number of few-shot instances.

B.3. Dataset statistics

Our Bongard-HOI benchmark provides disjoint training, validation, and testing sets. In specific, there are 118 concepts (visual relationships) and 21,956 few-shot instances in the training set. There are 17,184 and 13,941 few-shot instances in the validation and testing set, respectively, corresponding to 167 and 166 visual concepts. Detailed distribution of concepts and few-shot instances among different generalization types are provided in Table 1.

B.4. Illustration about the Context-Dependent Reasoning Property

Two Bongard problems (few-shot instancess) are shown in Fig. 2. For the same query image, among different context (*i.e.*, positive and negative examples), it receives different classification labels. This context-dependent reasoning property distinguishes our Bongard-HOI benchmark from other few-shot learning ones, where an image always has a fixed label.

C. More details on the oracle model

We first review how our oracle model works. Denoting the HOI detections in the $\mathcal{P}$ and $\mathcal{N}$ as $\mathcal{D}^P$ and $\mathcal{D}^N$, respectively. $\mathcal{D}^P$ contains the detections from all of the images in the $\mathcal{P}$, defined as $\mathcal{D}^P = \{c_i^P\}_{i=1}^{N_P}$, where c_i^P is a HOI triplet. N_P is the total number of detections. Note that there may be multiple or no detections for a single image. Similarly, $\mathcal{D}^P$ is defined as $\mathcal{D}^P = \{c_i^P\}_{i=1}^{N_P}$. According to the property of Bongard-HOI, the visual concept c_P should only appear in the $\mathcal{P}$, not in the $\mathcal{N}$. We, therefore, compute c_P as

$$c_P = \texttt{majority\_vote}(\mathcal{D}^P - \mathcal{D}^N),$$

where $-$ is the set operator for set subtraction. Given the detections $\mathcal{D}^q = \{c_i^q\}_{i=1}^{N_q}$ for the query image I_q, our prediction y becomes

$$y = \begin{cases} 1, & \text{if } c_P \in \mathcal{D}^q, \\ 0, & \text{otherwise.} \end{cases}$$

We now discuss some possible corner cases where the main paper does not cover.

What if `majority_vote` return multiple concepts? In this case, we simply enumerate each of them when making predictions for y. The predicted y will be 1 as long as at least one returned concepts present in $\mathcal{D}^q$; otherwise it will be 0.

What if $\mathcal{D}^P$, $\mathcal{D}^N$ or $\mathcal{D}^q$ is empty? In case when $\mathcal{D}^P$ is empty, we view this example as an failure case for our oracle model, as it does not induce the right concept as expected. On the contrary, it's totally fine that $\mathcal{D}^N$, meaning that no detection need to be removed from $\mathcal{D}^P$. Finally, how we handle the case when $\mathcal{D}_q$ is empty depends on the true label $y^\star$. If $y^\star$ is 1, then we view this example as an failure case. But we will make the prediction an automatic success if $y^\star$ is 0, since our oracle model finds there is no ground truth concept presenting in the query, which should be the right prediction.

We show successful cases of our oracle model in Fig. 3, Fig. 4, Fig. 5, Fig. 6. A failure case is shown in Fig. 7.

D. More Details on MTurk Data Curation

User interface. The user interface of data curation on the Amazon Mechanical Turkp (MTurk) platform is shown in Fig. 8. In the top part, we show images depicting a common visual relationship between human and objects in the left (*i.e.*, positive examples $\mathcal{P}$ in our Bongard problem). In the right, images that do not contain the visual relationship are shown (*i.e.*, negative examples $\mathcal{N}$). In the bottom part, given a query image, a tester needs to decide whether it depicts the particular visual relationship or not. Each MTurk job contains two few-shot instances, where a tester can freely switch between two pages. They can only submit the job once both two tasks are finished.

We do not tell the testers what objects to focus on to induce the common visual relationship. It is intended to be similar to what a machine learning model does, which needs to do object detection first.

Simple examples given to testers. To ensure testers who see the form of few-shot instances for the first time can successfully finish the job, we provide some examples of different visual relationships and encourage them to take a look at these examples before starting working on a job. Such examples are shown in Fig. 9.

MTurk job setting. We provided more details about the job setting below.

- **Region.** We restrict the regions of testers to be in the US and Germany.

Figure 3. **Illustration of our oracle model.** The concept in $\mathcal{P}$ is `wash car`.

Figure 4. **Illustration of our oracle model.** The concept in $\mathcal{P}$ is `ride car`.

Figure 5. **Illustration of our oracle model.** The concept in $\mathcal{P}$ is `teach person`.

- **Approval rate.** Each MTurker tester maintains a job approval rate based on their performance on previous jobs. We invite only MTurk testers whose job approval rate is equal to or greater than 98%.
- **Number of approved jobs.** Setting a qualification for the job approval rate only is not sufficient to hire high-quality testers since newly registered novel testers have a job approval rate of 100%. Therefore, we also set a qualification such that only testers who have more than 500 jobs approved previously are invited.
- **Invited annotators.** Through a couple of small-scale preliminary studies, we identified 35 reliable annotators on MTurk. For the large-scale data curation, we invited them to participate only.
- **Reward setting.** We provide \$0.15 for each job with an additional \$0.15 bonus if consistently high-quality annotations are made. According to our experiences of finishing the job, it roughly corresponds to about \$30 per hour.
- **Number of testers for each job.** We hire three independent testers for each job and aggregate their annotations. In specific, we only keep the few-shot instances where at least one of the three testers correctly classified the query image according to the ground-truth annotations. Otherwise, it suggests that a BP is either ambiguous or too difficult. We discard 2.5% of the few-shot instances that we submitted to MTurk.
- **Job life time.** A job will not be available after 7 days if it is not claimed by any tester. But we found that all of the jobs were finished within such a limit.

References

[1] Yu-Wei Chao, Zhan Wang, Yugeng He, Jiaxuan Wang, and Jia Deng. HICO: A benchmark for recognizing human-object interactions in images. In *ICCV*, 2015. 1

[2] Saurabh Gupta and Jitendra Malik. Visual semantic role labeling. *arXiv preprint arXiv:1505.04474*, 2015. 1

[3] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. *IEEE TPAMI*, 42(2):386–397, 2020. 2

[4] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov,

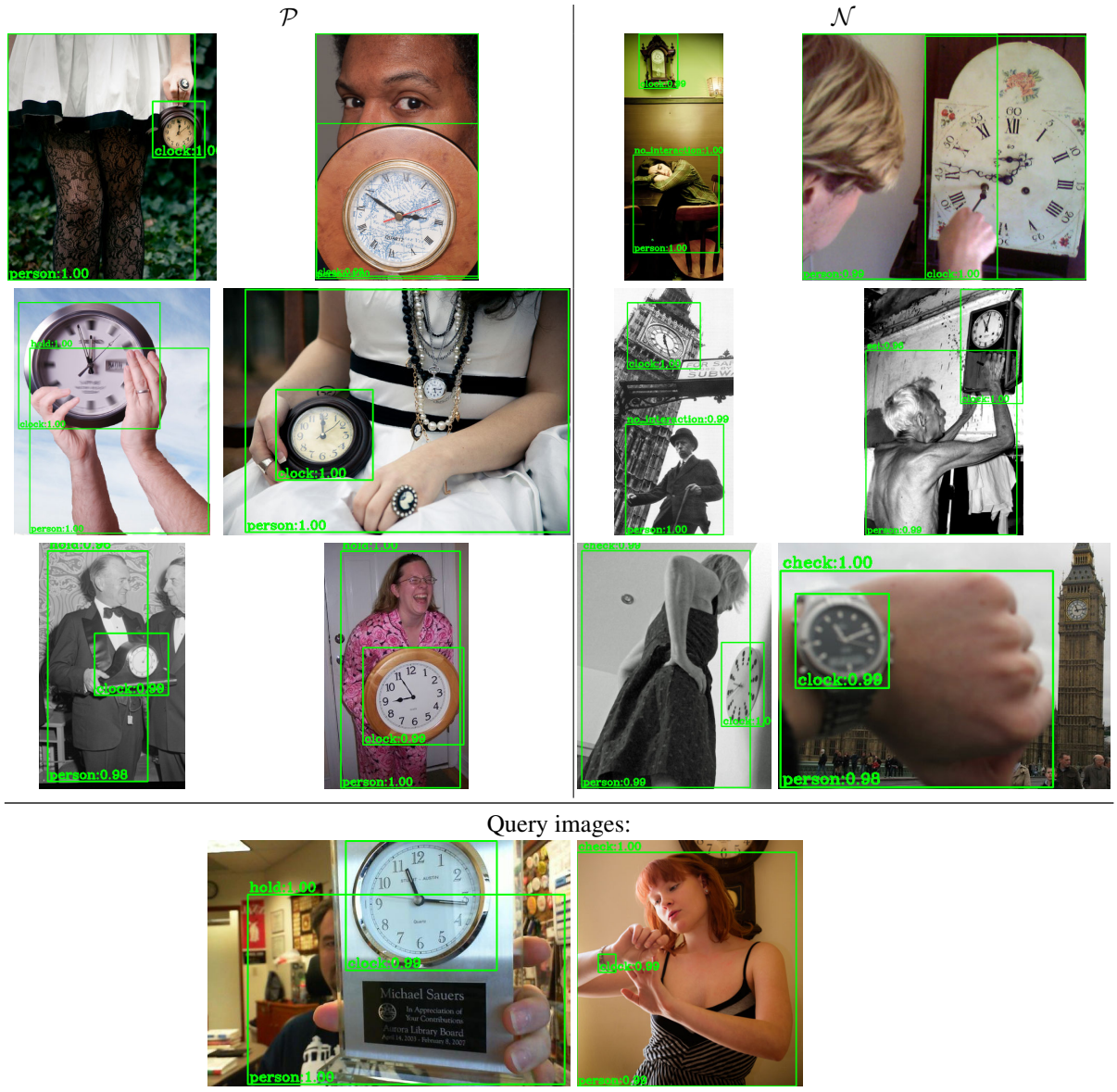


Figure 6. **Illustration of our oracle model.** The concept in $\mathcal{P}$ is `hold clock`.

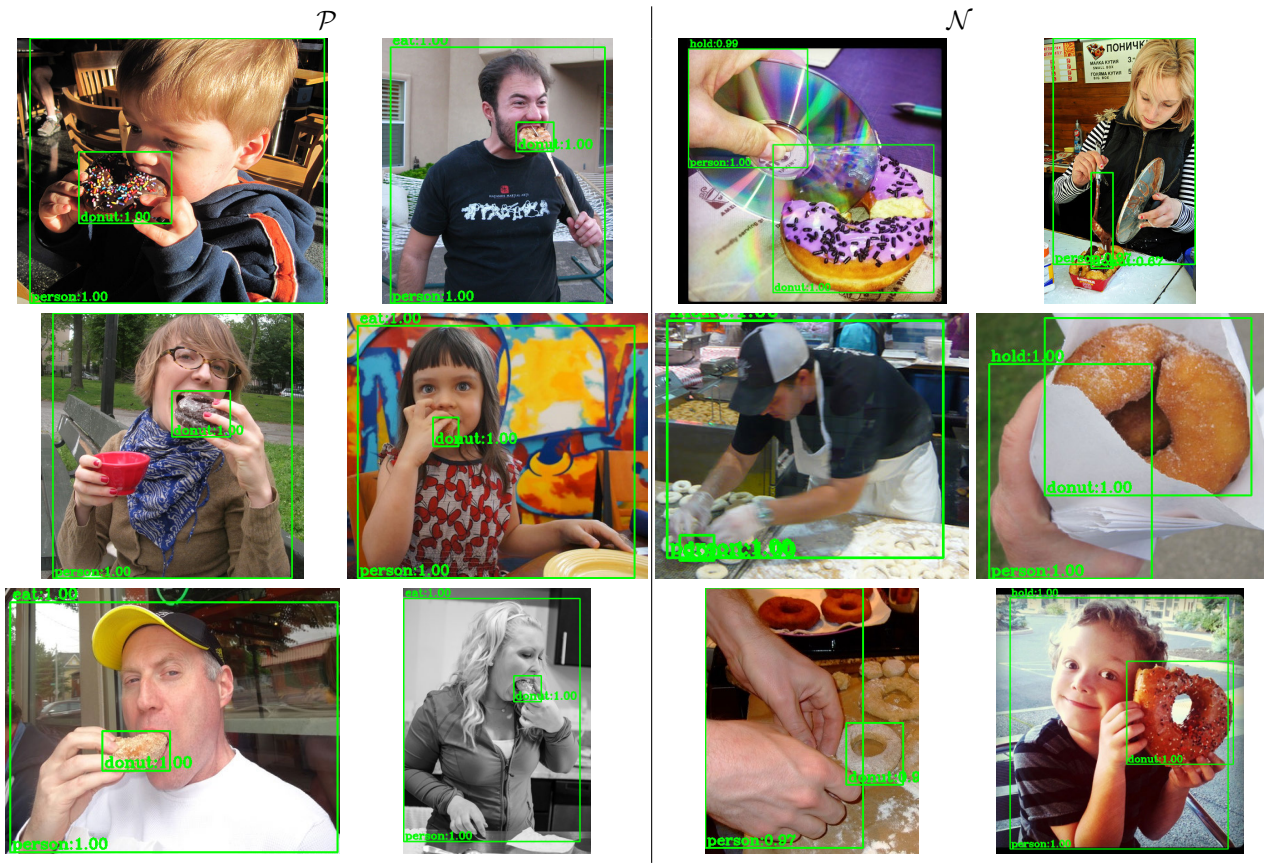
Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *IJCV*, 2020. 1

[5] Yonglu Li, Liang Xu, Xijie Huang, Xinpeng Liu, Ze Ma, Mingyang Chen, Shiyi Wang, Haoshu Fang, and Cewu Lu. HAKE: human activity knowledge engine. *CoRR*, abs/1904.06539, 2019. 1

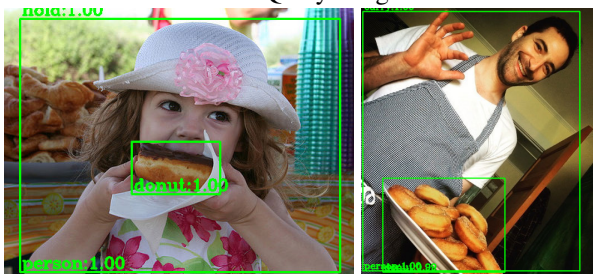
[6] Yue Liao, Si Liu, Fei Wang, Yanjie Chen, Chen Qian, and Jiashi Feng. PPDM: parallel point detection and matching for real-time human-object interaction detection. In *CVPR*, 2020. 1

[7] Bohan Zhuang, Qi Wu, Chunhua Shen, Ian D. Reid, and Anton van den Hengel. HCVRD: A benchmark for large-scale human-centered visual relationship detection. In Sheila A. McIlraith and Kilian Q. Weinberger, editors, *AAAI*, 2018. 1

[8] Cheng Zou, Bohan Wang, Yue Hu, Junqi Liu, Qian Wu, Yu Zhao, Boxun Li, Chenguang Zhang, Chi Zhang, Yichen Wei, et al. End-to-end human object interaction detection with hoi transformer. In *CVPR*, 2021. 8



Query images:



Predictions: **negative (wrong)** **negative**

Figure 7. **A failure of our oracle model.** The concept in $\mathcal{P}$ is `eat cake`. The *HOITrans* model [8] incorrectly recognizes the first query image as `hold cake` (which should be `eat cake`). As a result, it makes a wrong prediction for the first query image.

Figure 8. **The user interface (UI) of MTurk data curation.**

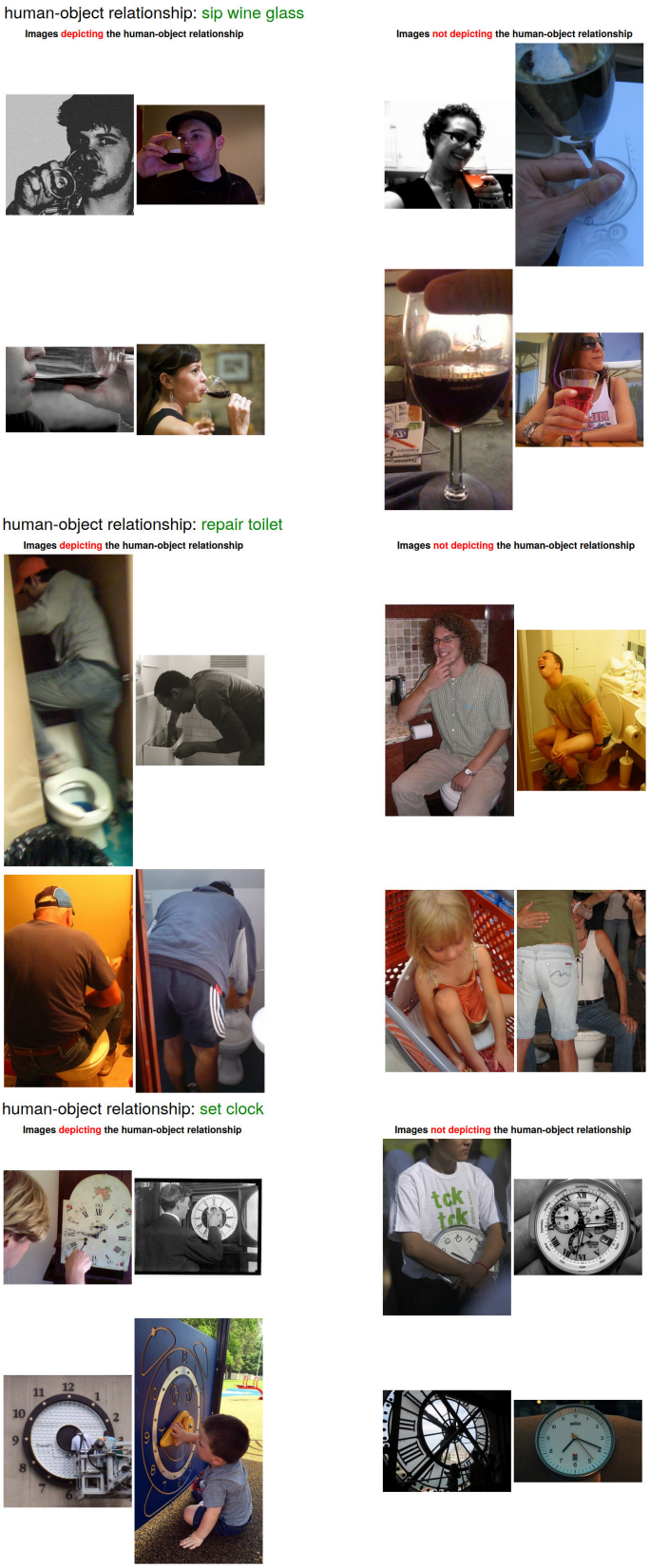


Figure 9. **Examples of different visual relationships given to MTurk testers.** For each example, we tell what the visual relationship is so that the testers can better understand the scope of the job.