Supplementary for *YouRefIt*: Embodied Reference Understanding with Language and Gesture

Yixin Chen[1], Qing Li[1], Deqian Kong[1], Yik Lun Kei[1],
Song-Chun Zhu[2,3,4], Tao Gao[1], Yixin Zhu[2,3], Siyuan Huang[1]
[1] University of California, Los Angeles [2] Beijing Institute for General Artificial Intelligence
[3] Peking University [4] Tsinghua University
https://yixchen.github.io/YouRefIt

1. Data Post-process and Annotation Details

In this section, we provide additional details of the annotation process. The annotation process takes two stages: (i) the annotation of temporal segments, canonical frames, and referent bounding boxes, and (ii) the annotation of sentence parsing. Fig. 1 visualizes the annotation process.

Quality consistency of MTurk videos We manually check the uploaded videos during the post-processing and discard low-quality ones (low resolution, small bounding boxes, poor lighting, *etc*.) to ensure high quality. Some videos are collected with hand-held cameras; we remove the ones with severe motion blurs during post-processing.

Bounding boxes and attributes We use Vatic [9] to annotate the bounding boxes of the referred object according to the tapping phase after each reference. The object color and material are also annotated. The taxonomy of typical object color and material is adopted from Visual Genome dataset [3]; we choose the ones that are identifiable and non-ambiguous. Specifically, we include "Black, Blue, Golden, Green, Pink, Purple, Red, Silver, White, Yellow" for color annotation and "Glass, Leather, Metal, Wooden, Plastic" for material annotation.

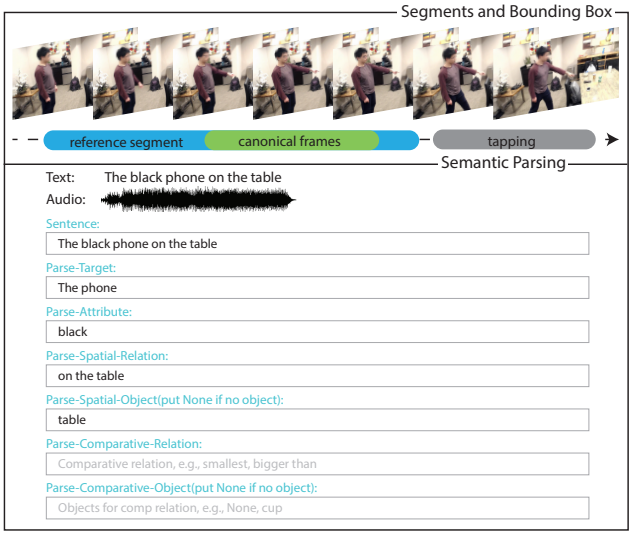


Figure 1: A visualization of dataset annotation process, which contains segments, bounding box annotation, sentence verification, and parsing.

2. Experiment Details

2.1. Object Size Distribution

In experiment, we report model performance on subsets with various object sizes, *i.e*., *small*, *medium* and *large*. Object size is estimated using the ratio between the area of the ground-truth object bounding box and the area of the image. The size thresholds are 0.48% and 1.76% based on the size distribution in the dataset. The size ratio distribution for ***YouRefIt*** is shown is Fig. 2.

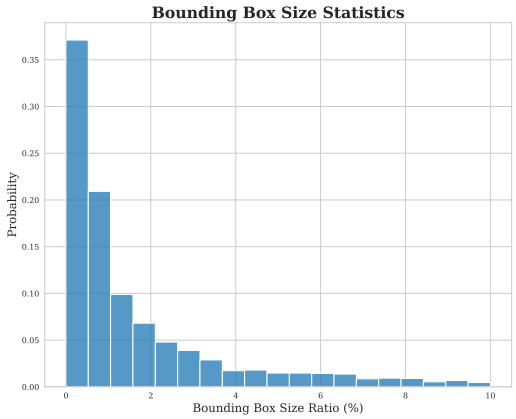


Figure 2: The distribution of bounding box size ratio.

2.2. Pointing Heatmap

Pointing is a gesture specifying a direction from a person's body, usually indicating a location or object. It is typically formed by extending the arm, the hand, and the index finger. Taking these evidence into consideration, we treat *Hand* pointing direction as the **primary pointing direction**

Figure 3: Examples of pointing heatmap.

when hand keypoints are detectable and *Arm* pointing direction otherwise. We design heuristics to determine which hand the referrer is using for pointing. In the case of *Hand* pointing, the primary pointing direction is from the wrist to the index fingertip. In the case of *Arm* pointing, the primary pointing direction is along the direction from elbow to hand. Following Fan *et al*. [1], we generate the pointing heatmap by a Gaussian distribution to model the variation of a pointing ray, *i.e*., a ray starting from the pointing wrist w.r.t. the primary pointing direction d_p,

$$p(\theta_o|d_p) \propto \frac{1}{\sigma} \exp(\frac{\theta_o^2}{2\sigma^2}), \tag{1}$$

where θ_o is the angle between a pointing ray and the primary pointing direction. We generate the pointing heatmap by computing θ_o for each grid in the image and use Eq. (1) to estimate the probability of this grid being pointed at by the primary pointing direction d_p. Fig. 3 shows some examples of the pointing heatmap. We choose 15° and 30° as the standard deviations during experiments, denoted as $\text{RPN}_{\text{pointing15}}$ and $\text{RPN}_{\text{pointing30}}$, respectively.

2.3. Implementation Details

Image ERU Following [11, 10, 6], we use YOLO's loss for training the bounding box prediction. More specifically, for each anchor box, we predict the relative offset and confidence score. A cross-entropy loss between the anchor box scores and the one hot ground-truth selection vector, a regression loss of the relative location and size offset and the same the regularization loss [10] over the word attention are used to train the model. The same weight is applied to all loss terms. During training, we re-scale all the images with the long edge to 256 pixels by padding. We train our model on a single Titan RTX A6000 GPU for 100 epochs with a batch size of 32. We use the RMSProp[8] optimizer, with an initial learning rate of 10^{-4} which decays by half every 10 epochs.

Video ERU Apart from the same loss for bounding box prediction in Image ERU, we further add the binary cross-entropy loss as the supervision for recognizing the canonical frames. For the ConvLSTM and Transformer baselines, we train our model on one A6000 GPU for 50 epochs with a batch size of 4. For the Frame-based model, we train on a single A6000 GPU for 50 epochs with a batch size of 32. The same optimizer and learning rate are used as in the Image ERU setting.

RPN+Saliency We generate Region of Interests (RoIs) by Region Proposal Network (RPN) from Faster R-CNN [7] pre-trained on the MSCOCO dataset [5]. We set the RoIs score threshold to be 0.05 and the Non-Maximum Suppression (NMS) threshold to be 0.5 as hyperparameters during the inference. For pointing saliency map, We train MSI-Net [4] on the *YouRefIt* dataset to predict the salient regions by considering both the latent scene structure and the gestural cues. We use the ground-truth referred bounding box as the saliency ground-truth and train the model on a single GTX 1070ti GPU for 100 epochs with a batch size of 10 and a learning rate of 10^{-5} using the Adam [2] optimizer.

3. Additional Image ERU Experiments

Table 2 in the paper demonstrates the necessity of embodiment because both language and gestures are required to achieve high performance. We further consolidate this argument by additional baselines: **Average** bounding box from RoIs, choose **Random** bounding box from RoIs, **Majority** (detects and picks chairs and bottles), choose bounding box using only the **Object Type**, **ReSC**$_{crop}$ (trained on images that crop out the human) and **ReSC**$_{box}$ (trained on images that mask out the human by a black box). Results are shown in Table 1. Of note, ReSC$_{crop}$ obtains a similar performance compared with ReSC[10] trained on original images. The reason is that most of the gestural information is still retained when directly cropping out the human, as illustrated in the following Fig. 4.

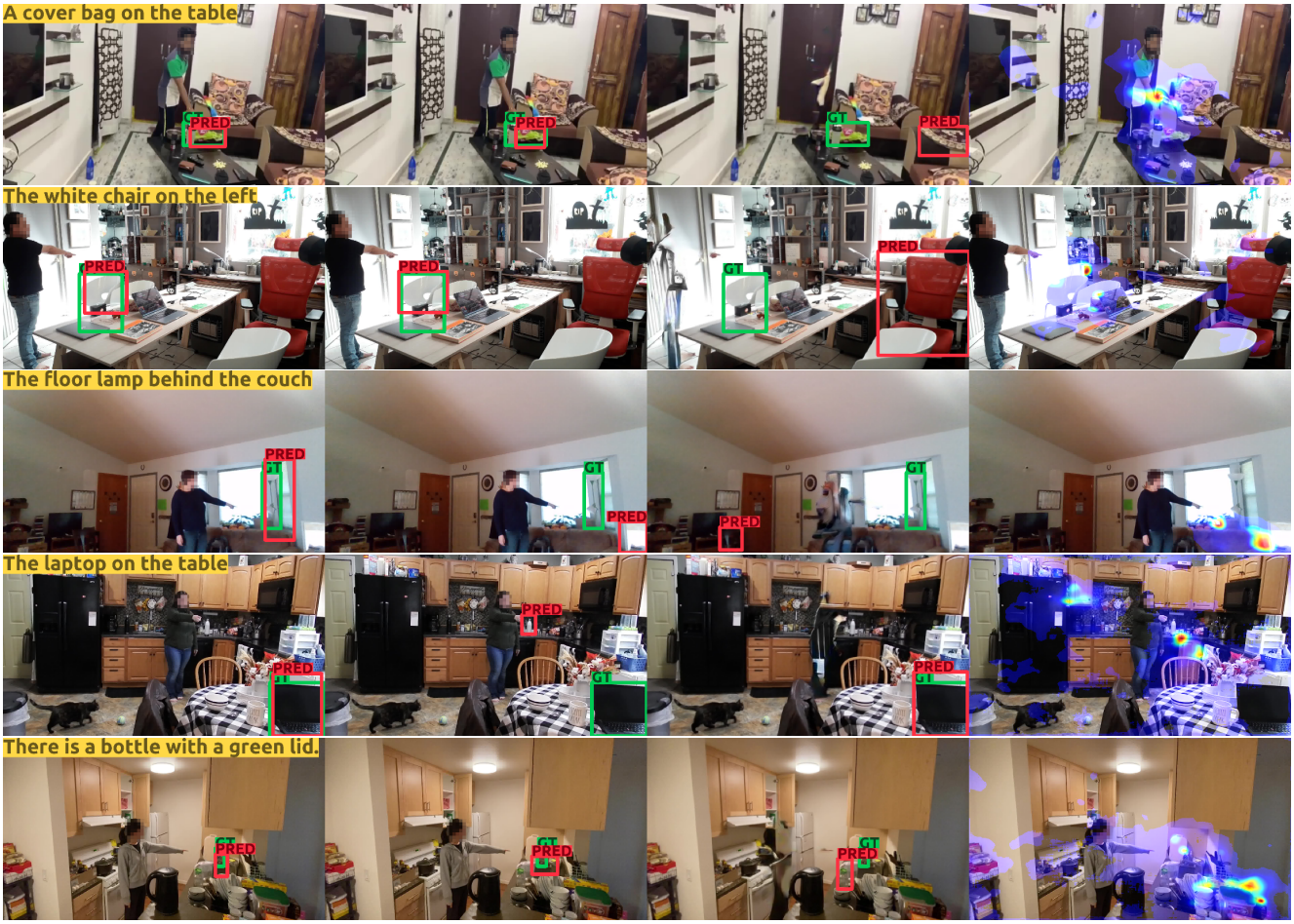
Figure 4: Example of cropped, masked and inpainted images.

Table 1: Additional results for Image ERU.

IoU=0.5	*all*	*small*	*medium*	*large*
Average	0.24	0.0	0.25	0.46
Random	5.6	4.8	5.3	6.6
Majority	6.5	4.3	8.2	6.0
Object Type	16.3	9.4	18.6	20.5
ReSC$_{inpaint}$	25.7	8.1	32.4	36.5
ReSC$_{box}$	22.1	9.2	28.6	31.7
ReSC$_{crop}$	35.2	16.1	48.8	40.1

4. Additional Qualitative Results

We show additional qualitative results of Image and Video Embodied Reference Understanding (ERU) in Fig. 5 and Fig. 6, respectively.



(a) Ours$_{Full}$ (b) Ours$_{no_lang}$ (c) ReSC$_{inpaint}$ (d) Saliency Map

Figure 5: **Qualitative results in Image ERU of representative models with various information sources and pointing saliency map.** Green/red boxes are the predicted/ground-truth reference targets. Sentences used during the references are shown at the top-left corner.

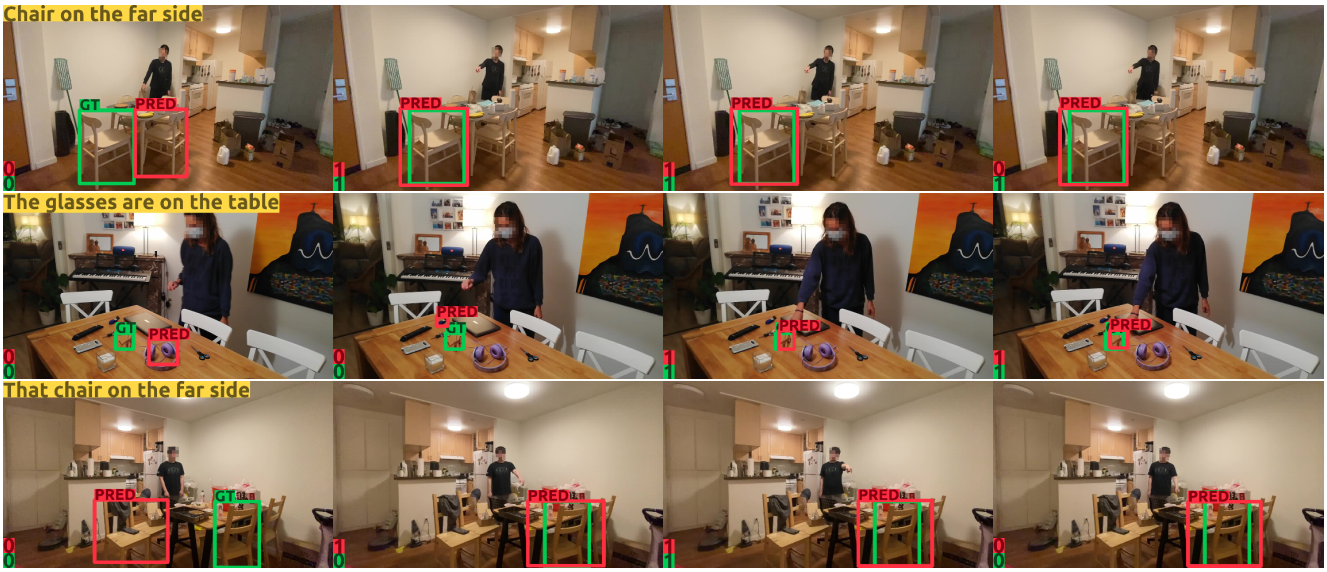


Figure 6: **Qualitative results in Video ERU of the ConvLSTM model.** Each row represents four selected frames from one reference clip. Green/red boxes indicate the predicted/ground-truth reference targets. 0 denotes non-canonical frame, and 1 canonical frame.

References

[1] Lifeng Fan, Yixin Chen, Ping Wei, Wenguan Wang, and Song-Chun Zhu. Inferring shared attention in social scene videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[2] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 2

[3] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision (IJCV)*, 123(1):32–73, 2017. 1

[4] Alexander Kroner, Mario Senden, Kurt Driessens, and Rainer Goebel. Contextual encoder–decoder network for visual saliency prediction. *Neural Networks*, 129:261–270, 2020. 2

[5] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2014. 2

[6] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018. 2

[7] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 39(6):1137–1149, 2016. 2

[8] Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31, 2012. 2

[9] Carl Vondrick, Donald Patterson, and Deva Ramanan. Efficiently scaling up crowdsourced video annotation. *International Journal of Computer Vision (IJCV)*, 101(1):184–204, 2013. 1

[10] Zhengyuan Yang, Tianlang Chen, Liwei Wang, and Jiebo Luo. Improving one-stage visual grounding by recursive sub-query construction. *arXiv preprint arXiv:2008.01059*, 2020. 2

[11] Zhengyuan Yang, Boqing Gong, Liwei Wang, Wenbing Huang, Dong Yu, and Jiebo Luo. A fast and accurate one-stage approach to visual grounding. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2019. 2