Supplementary Material for
DynRefer: Delving into Region-level Multimodal Tasks via Dynamic Resolution

Yuzhong Zhao[1*] Feng Liu[1*] Yue Liu[1] Mingxiang Liao[1] Chen Gong[2]
Qixiang Ye[1] Fang Wan[1†]
[1]University of Chinese Academy of Sciences [2]University of Virginia
zhaoyuzhong20@mails.ucas.ac.cn liufeng20@mails.ucas.ac.cn
liuyue171@mails.ucas.ac.cn liaomingxiang20@mails.ucas.ac.cn
chengong@virginia.edu qxye@ucas.ac.cn wanfang@ucas.ac.cn

A. Structure of the Decoders in DynRefer

The structure of the decoders in DynRefer is shown in Fig. 1.

i) Image Region Tagging. As shown in Fig. 1(D_{tag}), the region representation x_v is first mapped to a low-dimension embedding with a linear projection layer. Meanwhile, predefined 4585 tags are encoded by a frozen CLIP [5] text encoder and multi-layer perceptrons. Then, a query-based decoder [11, 20] ("Transformer layers" in Fig. 1) is used to calculate the confidences of the tags. The ground-truth tags are parsed from the caption of the referred region as shown in Fig. 2. Finally, the confidences of the tags are optimized by asymmetric loss [14], which is robust to imprecise supervision.

ii) Region-text Contrastive Learning. As shown in Fig. 1(D_{rtc}), it has a similar structure to D_{tag}. D_{rtc} normalizes the outputs from the query-based decoder and projects them into similarity scores, which are optimized by the pairwise Sigmoid loss for Language-Image Pre-training [19].

iii) Language Modeling. As shown in Fig. 1(D_{llm}), following ControlCap [21], random control words parsed from the ground-truth captions are combined to a sentence, $i.e.$, "`white dog, sofa[SEP]`". The sentence is encoded into the control embedding by the tokenizer and word embedding layer of the large language model. After that, a learnable memory unit is added to the control embedding. Finally, the control embedding and the projected region representation are concatenated and jointly sent into the large language model for text generation.

B. Inference with Trained Decoders.

With trained decoders, the region representation x_v can be decoded into region-level language descriptions, including tags, categories, attributes and captions. Their production are elaborated below:

i) *tags*. The tags of the region are generated by D_{tag}. Following [4, 20], we use a set of 4585 tags. During inference, we first query the decoder with the predefined tags to get the confidences. Then, the tags are filtered by a predefined tagging threshold.

ii) *categories*. The category of the region is generated by D_{rtc}. During inference, we query the decoder with the template "`a photo of a {cls}`" and select the category with the highest score.

iii) *attributes*. The attributes of the region are generated by D_{rtc}. During inference, we first query the decoder with attribute templates following OVAD [1], *e.g.*, "`the object has {attr}`". Then, attributes with high scores are selected as the results.

iv) *captions*. The caption of the region is generated by D_{llm}. During inference, we first use the tags of high confidence to form a control sentence, $i.e.$, "`{tag1}, {tag2}, {tag3}, ···, [SEP]`". Then, the control sentence and the region representation are used to control the language language model for caption generation.

C. Details of the Control Embeddings

Following ControlCap [21], we introduce control words to alleviate the caption degeneration issue, which refers to the fact that pre-trained multimodal models tend to predict the most frequent captions but miss the less frequent ones. During training, the control words are parsed from the ground-truth captions (Fig. 2) and are randomly dropped in accordance with a Bernoulli distribution, which is detailed in Fig. 2. The remaining control words are shuffled and combined with a `[SEP]` token to form a control sentence, $i.e.$, "`white dog, sofa[SEP]`" in Fig. 1. The sentence is encoded into the control embedding by the tokenizer and word embedding layer of the large language model. During inference, we build the control embeddings with high-confidence tags from the outputs of DynRefer.

[*] Equal contribution. † Corresponding Author.

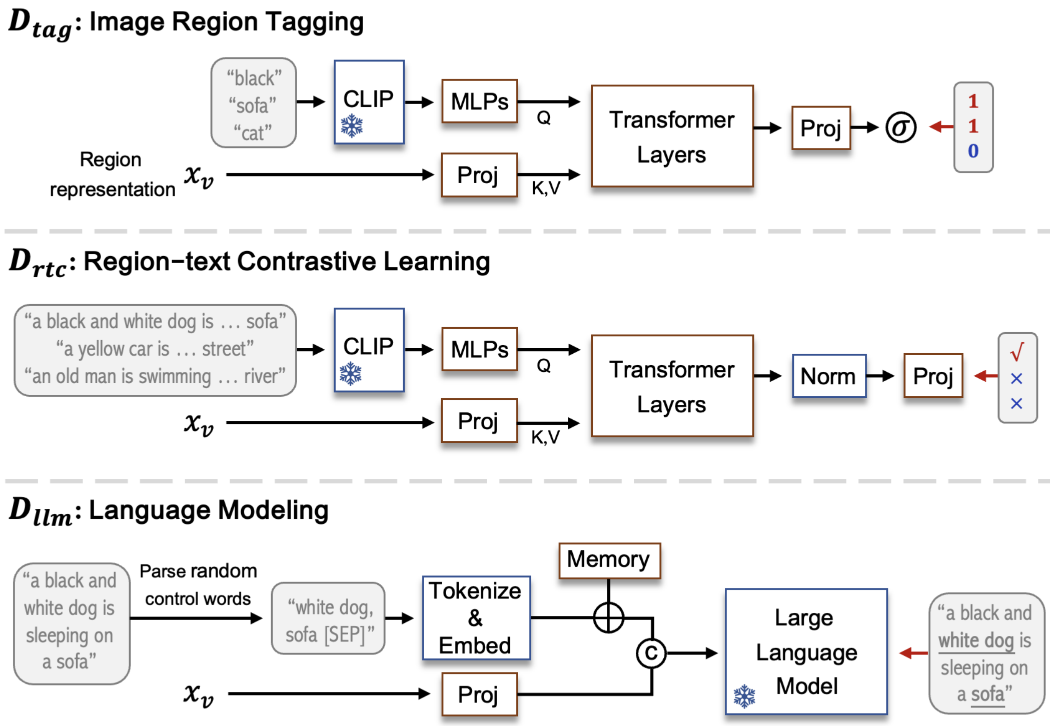

D_{tag}: Image Region Tagging
"black"
"sofa"
"cat"
CLIP
MLPs
Q
Transformer Layers
Proj
σ
1
1
0
Region representation
x_v
Proj
K,V
D_{rtc}: Region−text Contrastive Learning
"a black and white dog is … sofa"
"a yellow car is … street"
"an old man is swimming … river"
CLIP
MLPs
Q
Transformer Layers
Norm
Proj
√
×
×
x_v
Proj
K,V
D_{llm}: Language Modeling
"a black and white dog is sleeping on a sofa"
Parse random control words
"white dog, sofa [SEP]"
Tokenize & Embed
Memory
C
Large Language Model
"a black and white dog is sleeping on a sofa"
x_v
Proj

Figure 1. The detailed structure of multimodal decoders D_* of DynRefer. "Proj" is a linear projection layer. "σ" is the sigmoid activation function. "Memory" is a learnable embedding. The "Transformer Layers" denotes query-based decoders [11, 20] that contains only cross-attention layers and feed forward networks.

Table 1. Evaluation of the align module of DynRefer on region-level multimodal benchmarks.

	Align module	Inference	Vis. FLOPs	OVAD	COCO	VG-COCO	RefCOCOg	
				mAP (%)	Acc (%)	mAP (%)	CIDEr	METEOR
1	✗	No prior	790G	27.3	88.4	47.1	17.9	113.6
2	✓	No prior	792G	27.3	88.9	47.3	18.2	117.7
3	✗	Image prior	790G	28.1	**90.5**	46.6	17.7	113.5
4	✓	Image prior	792G	**28.7**	90.3	**47.4**	**18.2**	**118.6**

D. Illustration of pHASH operation

The pHASH (Perceptual Hash) operation is a hashing algorithm that generates a "perceptual fingerprint" of an image based on its visual characteristics. The key features of pHASH operation are summaries as follows:

i) Perceptual Similarity: The pHASH operation is designed to generate similar hash values for visually similar images. It focuses on the aspects of the image that humans perceive ($e.g.$, shapes, colors).

ii) Tolerance to Minor Modifications: The pHASH operation is robust to minor changes like resizing, cropping, compression, or slight color variations. This tolerance makes it ideal for detecting duplicates or near-duplicates of images.

iii) Fixed-Length Output: The output of the pHASH operation is always a fixed-length binary string (e.g., 64 or 128 bits), regardless of the size of the input image. This makes it easy to compare images of varying sizes.

iv) Fast Computation: The pHASH operation is optimized for speed and is computationally efficient, allowing it to be used for large amount image comparisons.

E. Detailed Experimental Settings

The detailed model, dataset, evaluation settings of DynRefer is summarized as follows:

Model implementation. DynRefer is implemented upon the LAVIS [8] framework, where large language model and vision resampler are respectively initialized by FlanT5$_{\text{XL}}$ [2] and Q-former [9]. All the sampled views are resized to 224×224 resolution. All models are trained using 8 NVIDIA A800 GPUs by 5 epochs, with the Adam optimizer where the batch size is set to 512. The total training time is less

Table 2. Evaluation of the inference strategy of DynRefer on region-level multimodal benchmarks.

	Training	Inference	Vis. FLOPs	OVAD	COCO	VG-COCO	RefCOCOg	
				mAP (%)	Acc (%)	mAP (%)	CIDEr	METEOR
1	Stochastic 2-view	No prior	530G	26.1	87.8	46.6	17.9	114.4
2	Stochastic 2-view	Image prior	530G	27.5	89.3	46.8	17.9	114.7
3	Stochastic 2-view	Task prior	530G	28.1	90.2	47.0	18.1	115.6
4	Stochastic 3-view	No prior	792G	27.3	88.9	47.3	**18.2**	117.7
5	Stochastic 3-view	Image prior	792G	28.7	90.3	**47.4**	**18.2**	**118.6**
6	Stochastic 3-view	Task prior	792G	**29.4**	**90.4**	**47.4**	**18.2**	118.3

Table 3. Analysis of parameter composition of DynRefer. Modules that contain very few parameters are omitted for clarity.

	ViT	Align module	Vision Resampler	D_{tag}	D_{rtc}	CLIP	LLM
Trainable	✗	✓	✓	✓	✓	✗	✗
Parameters (%)	23.78	0.20	2.53	0.05	0.05	2.99	68.79
Flops (G)	783.5	2.1	6.4	6.2	0.4	6.5	80.1

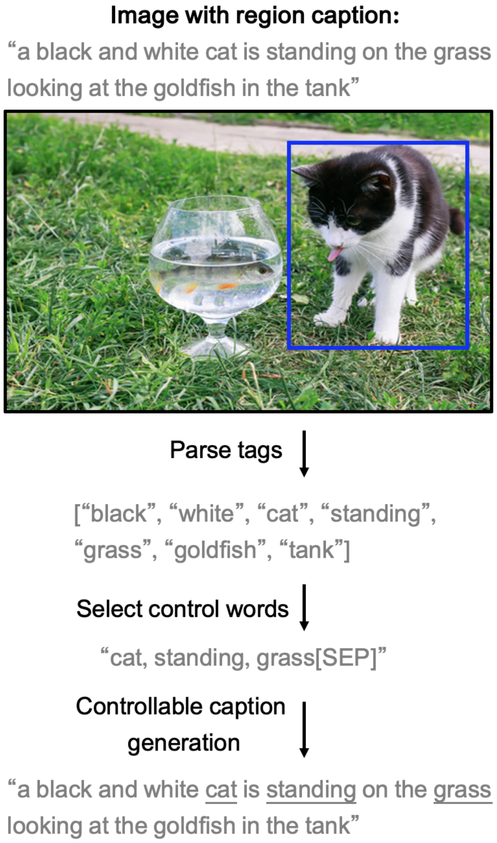


Image with region caption:
“a black and white cat is standing on the grass looking at the goldfish in the tank”
Parse tags
[“black”, “white”, “cat”, “standing”, “grass”, “goldfish”, “tank”]
Select control words
“cat, standing, grass[SEP]”
Controllable caption generation
“a black and white cat is standing on the grass looking at the goldfish in the tank”

Figure 2. Illustration of the generation process of tags and control words used in DynRefer.

than 20 hours. The initial learning rate is set to 1×10^{-4} with a cosine learning rate decay. The detailed hyperparameters during training and inference are shown in Tab. 4.

Table 4. Detailed hyperparameters during training and inference.

Training	**Value**
GPUs	8× A800 80G
batch size	512
training epochs	5
learning policy	cosine annealing
initial learning rate	1e-4
minimum learning rate	0
weight decay ratio	0.05
warmup steps	5000
Inference	**Value**
number of beams	5
number of views	3 (default)
view selection	Image prior (default)

Considering that dense captioning requires the model to initially generate dense bounding-boxes, we utilize a GRiT [16] model trained on the VG to acquire object locations. During the inference stage, we use the bounding boxes and object scores predicted by GRiT, and then replace its predicted caption with DynRefer to get the final result.

Datasets. For all tasks, DynRefer is trained using Visual Genome (VG) [7] and RefCOCOg [17]. For ablation studies, DynRefer is trained using VG-COCO [15] and RefCOCOg [17]. For evaluation, we evaluate the region-level captioning performance on VG, VG-COCO [15], and RefCOCOg, the open vocabulary attribute detection performance on OVAD [1], and the region recognition performance on COCO [10]

Evaluation Metrics. For region-level captioning, the METEOR score and CIDEr score are adopted as the evaluation metrics following [3, 13, 18]. For dense captioning, mean Average Precision (mAP) [6] is adopted as the evaluation metric following [6, 12]. The mAP is calculated across a range of thresholds for both localization and language accuracy, $i.e.$, the intersection over union (IoU) thresholds (0.3, 0.4, 0.5, 0.6, 0.7) are used for localization and the METEOR score' thresholds (0, 0.05, 0.1, 0.15, 0.2, 0.25) is adopted for evaluating the language generation. Since DynRefer lacks the capability to perform object detection, we utilize a GRiT [16] model trained on VG to acquire object locations. For open vocabulary attribute detection, mAP is adopted as the evaluation metric following OVAD [1]. For region recognition, mAP and Accuracy (Acc.) are are adopted as the evaluation metrics following [3, 22].

F. Additional Experimental Results

We provide additional experimental results in the supplementary as follows:

Stochastic Multi-view Embedding: Align module. The effectiveness of the align module is validated in Tab. 1. By spatially aligning the region embeddings across multiple views, DynRefer achieves a 0.6% improvement in mAP on OVAD, a 0.8% improvement in mAP on VG-COCO, and a 5.1 increase in METEOR on RefCOCOg. These results validate the effectiveness of the proposed align module.

Selectively Multimodal Referring. As shown in Tab. 2, we evaluate DynRefer under different view counts and inference strategies. In the "No prior" strategy, views are randomly selected for each sample. In the "Task prior" strategy, the view containing the referred region is always selected, and the top-(n-1) views are chosen based on the results from Fig. 4 for an n-view model. In the "Image prior" strategy, views are selected according to Eq. 1 in the main paper. For the 2-view DynRefer model, the performance of different strategies ranks as: “Task prior > Image prior > No prior”. For the 3-view model, the ranking is: “Task prior $\approx$ Image prior > No prior”. While the “Task prior” strategy works well, the "Image prior" strategy offers greater flexibility. It is task-independent and can dynamic select views to each image region. This makes it particularly suitable for models that need to handle multiple tasks with a unified region representation. Based on these advantages, we adopt “Image prior” as the default inference strategy.

Statistics of Parameters and FLOPs. The parameter and flop composition of DynRefer is shown in Tab. 3. DynRefer has few trainable parameters and can be trained efficiently.

Additional Visualization Results. We provide additional visualization results of Fig. 5 and Fig. 6 in the main document. The results are shown in Fig. 5 and Fig. 3 4 6 7 8.

G. Limitations

Though DynRefer significantly outperforms previous state-of-the-arts on multiple multimodal tasks, it still doesn’t perfectly mimic the visual cognition system of human. A real human can adjust the resolution of visual inputs in a more dynamic and flexible way. Better simulation strategy can be explored in the future work.

References

[1] Maria A Bravo, Sudhanshu Mittal, Simon Ging, and Thomas Brox. Open-vocabulary attribute detection. In *IEEE CVPR*, pages 7041–7050, 2023. 1, 3, 4

[2] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Y. Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*, 2022. 2

[3] Qiushan Guo, Shalini De Mello, Hongxu Yin, Wonmin Byeon, Ka Chun Cheung, Yizhou Yu, Ping Luo, and Sifei Liu. Regiongpt: Towards region understanding vision language model. In *IEEE CVPR*, pages 13796–13806, 2024. 3, 4

[4] Xinyu Huang, Youcai Zhang, Jinyu Ma, Weiwei Tian, Rui Feng, Yuejie Zhang, Yaqian Li, Yandong Guo, and Lei Zhang. Tag2text: Guiding vision-language model via image tagging. *arXiv preprint arXiv:2303.05657*, 2023. 1

[5] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, 2021. 1

[6] Justin Johnson, Andrej Karpathy, and Li Fei-Fei. Densecap: Fully convolutional localization networks for dense captioning. In *IEEE CVPR*, pages 4565–4574, 2016. 4

[7] 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. *IJCV*, pages 32–73, 2017. 3

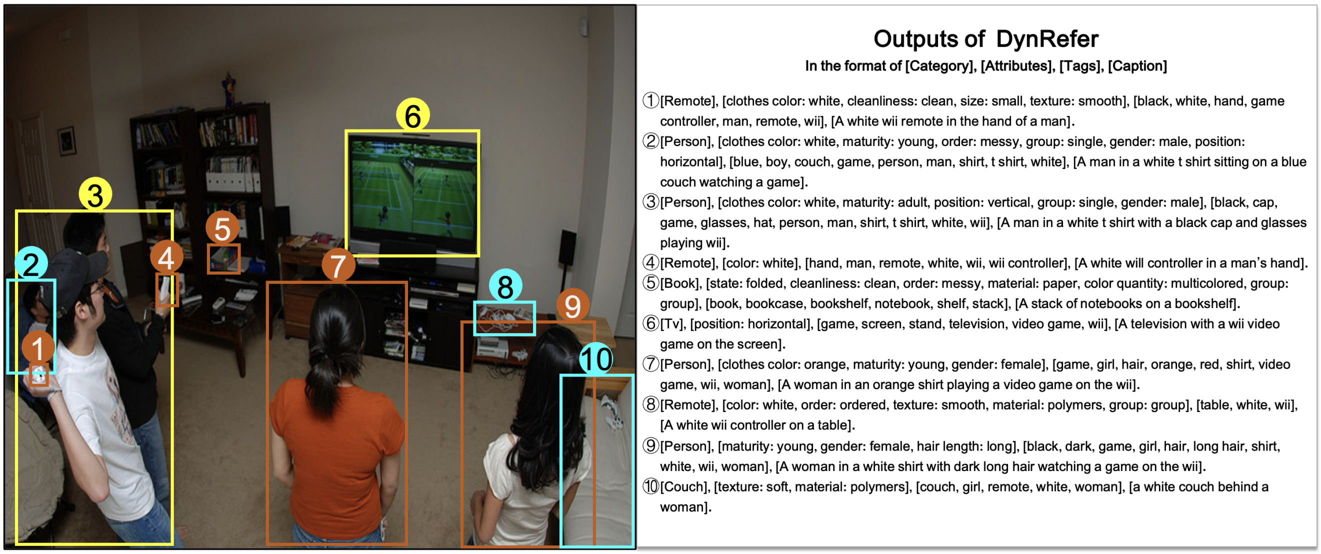
[8] Dongxu Li, Junnan Li, Hung Le, Guangsen Wang, Silvio Savarese, and Steven CH Hoi. Lavis: A library for language-vision intelligence. *arXiv preprint arXiv:2209.09019*, 2022. 2

[9] Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H. Hoi. BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, pages 19730–19742, 2023. 2

[10] 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 *ECCV*, pages 740–755, 2014. 3

[11] Shilong Liu, Lei Zhang, Xiao Yang, Hang Su, and Jun Zhu. Query2label: A simple transformer way to multi-label classification. *arXiv preprint arXiv:2107.10834*, 2021. 1, 2

[12] Yanxin Long, Youpeng Wen, Jianhua Han, Hang Xu, Pengzhen Ren, Wei Zhang, Shen Zhao, and Xiaodan Liang.



Outputs of DynRefer

In the format of [Category], [Attributes], [Tags], [Caption]

①[Remote], [clothes color: white, cleanliness: clean, size: small, texture: smooth], [black, white, hand, game controller, man, remote, wii], [A white wii remote in the hand of a man].

②[Person], [clothes color: white, maturity: young, order: messy, group: single, gender: male, position: horizontal], [blue, boy, couch, game, person, man, shirt, t shirt, white], [A man in a white t shirt sitting on a blue couch watching a game].

③[Person], [clothes color: white, maturity: adult, position: vertical, group: single, gender: male], [black, cap, game, glasses, hat, person, man, shirt, t shirt, white, wii], [A man in a white t shirt with a black cap and glasses playing wii].

④[Remote], [color: white], [hand, man, remote, white, wii, wii controller], [A white will controller in a man's hand].

⑤[Book], [state: folded, cleanliness: clean, order: messy, material: paper, color quantity: multicolored, group: group], [book, bookcase, bookshelf, notebook, shelf, stack], [A stack of notebooks on a bookshelf].

⑥[Tv], [position: horizontal], [game, screen, stand, television, video game, wii], [A television with a wii video game on the screen].

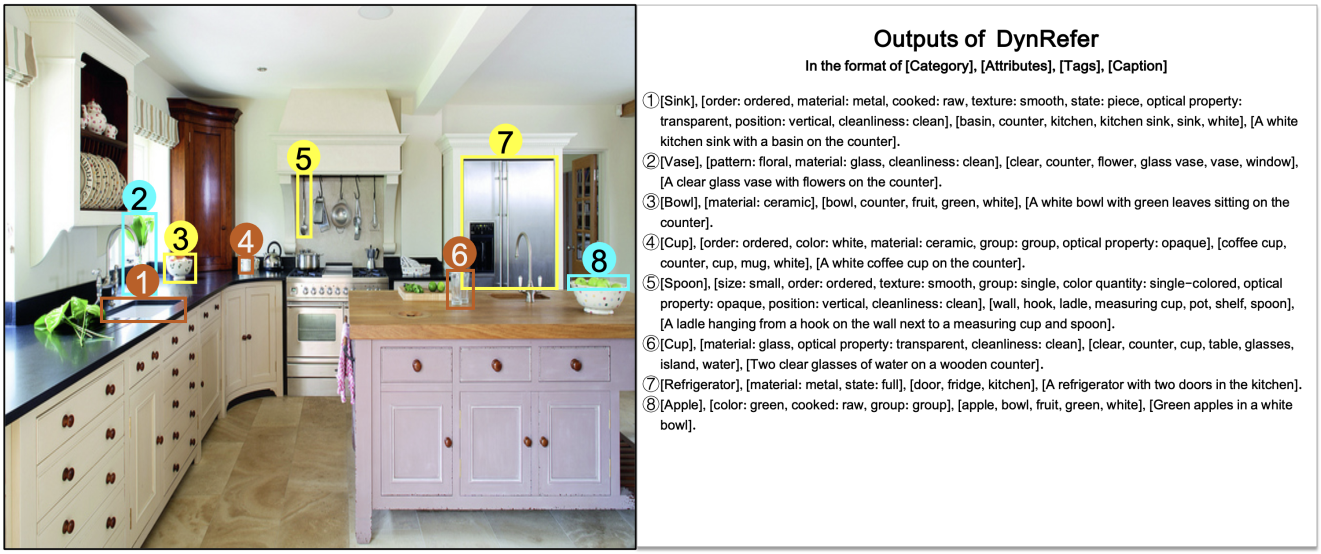
⑦[Person], [clothes color: orange, maturity: young, gender: female], [game, girl, hair, orange, red, shirt, video game, wii, woman], [A woman in an orange shirt playing a video game on the wii].

⑧[Remote], [color: white, order: ordered, texture: smooth, material: polymers, group: group], [table, white, wii], [A white wii controller on a table].

⑨[Person], [maturity: young, gender: female, hair length: long], [black, dark, game, girl, hair, long hair, shirt, white, wii, woman], [A woman in a white shirt with dark long hair watching a game on the wii].

⑩[Couch], [texture: soft, material: polymers], [couch, girl, remote, white, woman], [a white couch behind a woman].

Figure 3. More results of Fig. 6 in the main paper, $i.e.$, illustration of DynRefer's multi-task capability.



Outputs of DynRefer

In the format of [Category], [Attributes], [Tags], [Caption]

①[Sink], [order: ordered, material: metal, cooked: raw, texture: smooth, state: piece, optical property: transparent, position: vertical, cleanliness: clean], [basin, counter, kitchen, kitchen sink, sink, white], [A white kitchen sink with a basin on the counter].

②[Vase], [pattern: floral, material: glass, cleanliness: clean], [clear, counter, flower, glass vase, vase, window], [A clear glass vase with flowers on the counter].

③[Bowl], [material: ceramic], [bowl, counter, fruit, green, white], [A white bowl with green leaves sitting on the counter].

④[Cup], [order: ordered, color: white, material: ceramic, group: group, optical property: opaque], [coffee cup, counter, cup, mug, white], [A white coffee cup on the counter].

⑤[Spoon], [size: small, order: ordered, texture: smooth, group: single, color quantity: single−colored, optical property: opaque, position: vertical, cleanliness: clean], [wall, hook, ladle, measuring cup, pot, shelf, spoon], [A ladle hanging from a hook on the wall next to a measuring cup and spoon].

⑥[Cup], [material: glass, optical property: transparent, cleanliness: clean], [clear, counter, cup, table, glasses, island, water], [Two clear glasses of water on a wooden counter].

⑦[Refrigerator], [material: metal, state: full], [door, fridge, kitchen], [A refrigerator with two doors in the kitchen].

⑧[Apple], [color: green, cooked: raw, group: group], [apple, bowl, fruit, green, white], [Green apples in a white bowl].

Figure 4. More results of Fig. 6 in the main paper, $i.e.$, illustration of DynRefer's multi-task capability.

Capdet: Unifying dense captioning and open-world detection pretraining. In *IEEE CVPR*, pages 15233–15243, 2023. 4

[13] Hanoona Rasheed, Muhammad Maaz, Sahal Shaji, Abdelrahman Shaker, Salman Khan, Hisham Cholakkal, Rao M Anwer, Eric Xing, Ming-Hsuan Yang, and Fahad S Khan. Glamm: Pixel grounding large multimodal model. In *IEEE CVPR*, pages 13009–13018, 2024. 3

[14] Tal Ridnik, Emanuel Ben-Baruch, Nadav Zamir, Asaf Noy, Itamar Friedman, Matan Protter, and Lihi Zelnik-Manor. Asymmetric loss for multi-label classification. In *IEEE CVPR*, pages 82–91, 2021. 1

[15] Zhuang Shao, Jungong Han, Demetris Marnerides, and Kurt Debattista. Region-object relation-aware dense captioning via transformer. *IEEE TNNLS*, 2022. 3

[16] Jialian Wu, Jianfeng Wang, Zhengyuan Yang, Zhe Gan, Zicheng Liu, Junsong Yuan, and Lijuan Wang. Grit: A generative region-to-text transformer for object understanding. In *ECCV*, pages 207–224. Springer, 2025. 3, 4

[17] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *ECCV*, pages 69–85, 2016. 3

[18] Yuqian Yuan, Wentong Li, Jian Liu, Dongqi Tang, Xinjie Luo, Chi Qin, Lei Zhang, and Jianke Zhu. Osprey: Pixel understanding with visual instruction tuning. In *IEEE CVPR*, pages 28202–28211, 2024. 3

[19] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *IEEE ICCV*, pages 11975–11986, 2023. 1

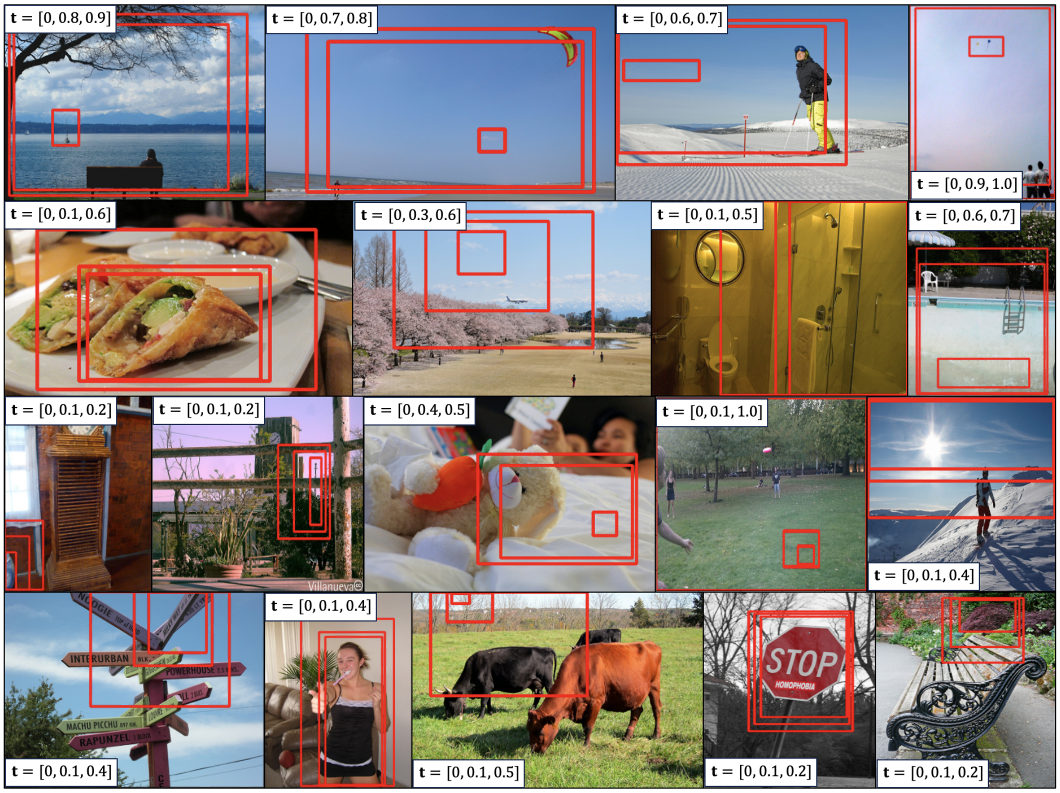
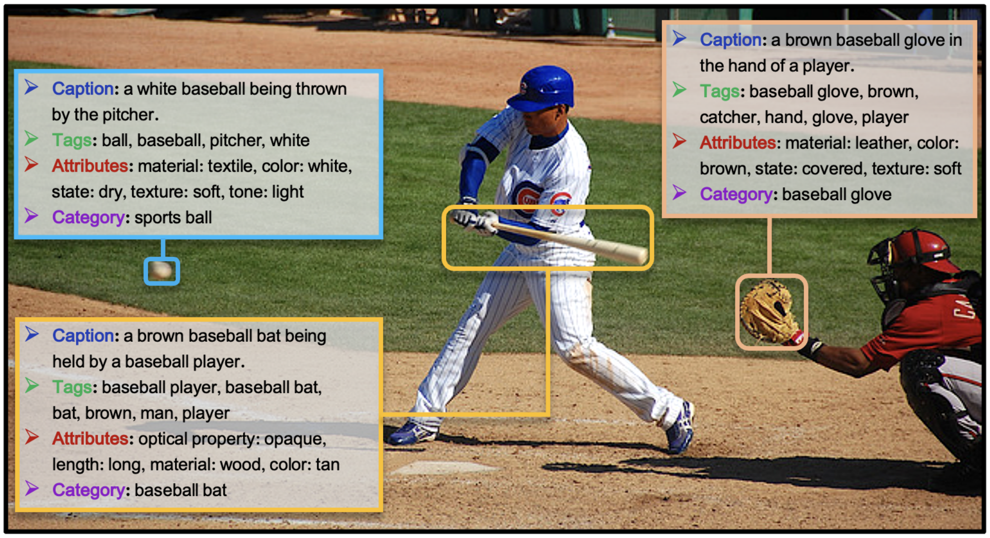


Figure 5. More results of Fig. 5 in the main paper, *i.e.*, visualization of selected views using image prior.



- **Caption**: a white baseball being thrown by the pitcher.
- **Tags**: ball, baseball, pitcher, white
- **Attributes**: material: textile, color: white, state: dry, texture: soft, tone: light
- **Category**: sports ball

- **Caption**: a brown baseball glove in the hand of a player.
- **Tags**: baseball glove, brown, catcher, hand, glove, player
- **Attributes**: material: leather, color: brown, state: covered, texture: soft
- **Category**: baseball glove

- **Caption**: a brown baseball bat being held by a baseball player.
- **Tags**: baseball player, baseball bat, bat, brown, man, player
- **Attributes**: optical property: opaque, length: long, material: wood, color: tan
- **Category**: baseball bat

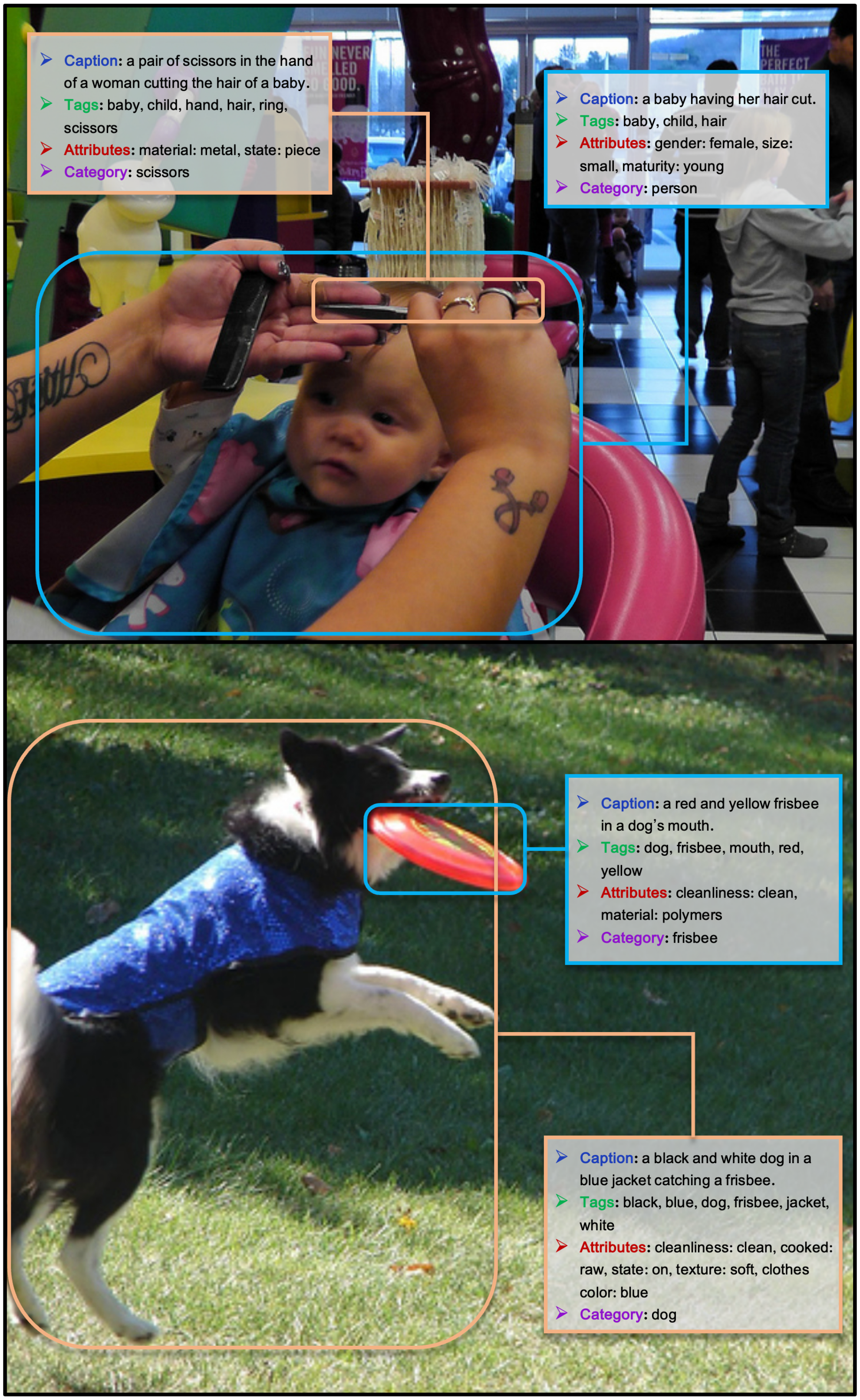
Figure 6. Illustration of DynRefer's multi-task capability. It can generate captions, tags, attributes, categories, using a single model, for any referred regions.

[20] Youcai Zhang, Xinyu Huang, Jinyu Ma, Zhaoyang Li, Zhaochuan Luo, Yanchun Xie, Yuzhuo Qin, Tong Luo, Yaqian Li, Shilong Liu, et al. Recognize anything: A strong image tagging model. *arXiv preprint arXiv:2306.03514*, 2023. 1, 2

[21] Yuzhong Zhao, Yue Liu, Zonghao Guo, Weijia Wu, Chen Gong, Qixiang Ye, and Fang Wan. Controlcap: Controllable region-level captioning. In *European Conference on Computer Vision*, pages 21–38. Springer, 2024. 1

[22] Yiwu Zhong, Jianwei Yang, Pengchuan Zhang, Chunyuan Li, Noel Codella, Liunian Harold Li, Luowei Zhou, Xiyang Dai,

Lu Yuan, Yin Li, et al. Regionclip: Region-based language-image pretraining. In *IEEE CVPR*, pages 16793–16803, 2022. 4



- Caption: a pair of scissors in the hand of a woman cutting the hair of a baby.
- Tags: baby, child, hand, hair, ring, scissors
- Attributes: material: metal, state: piece
- Category: scissors

- Caption: a baby having her hair cut.
- Tags: baby, child, hair
- Attributes: gender: female, size: small, maturity: young
- Category: person

- Caption: a red and yellow frisbee in a dog's mouth.
- Tags: dog, frisbee, mouth, red, yellow
- Attributes: cleanliness: clean, material: polymers
- Category: frisbee

- Caption: a black and white dog in a blue jacket catching a frisbee.
- Tags: black, blue, dog, frisbee, jacket, white
- Attributes: cleanliness: clean, cooked: raw, state: on, texture: soft, clothes color: blue
- Category: dog

Figure 7. Illustration of DynRefer's multi-task capability. It can generate captions, tags, attributes, categories, using a single model, for any referred regions.

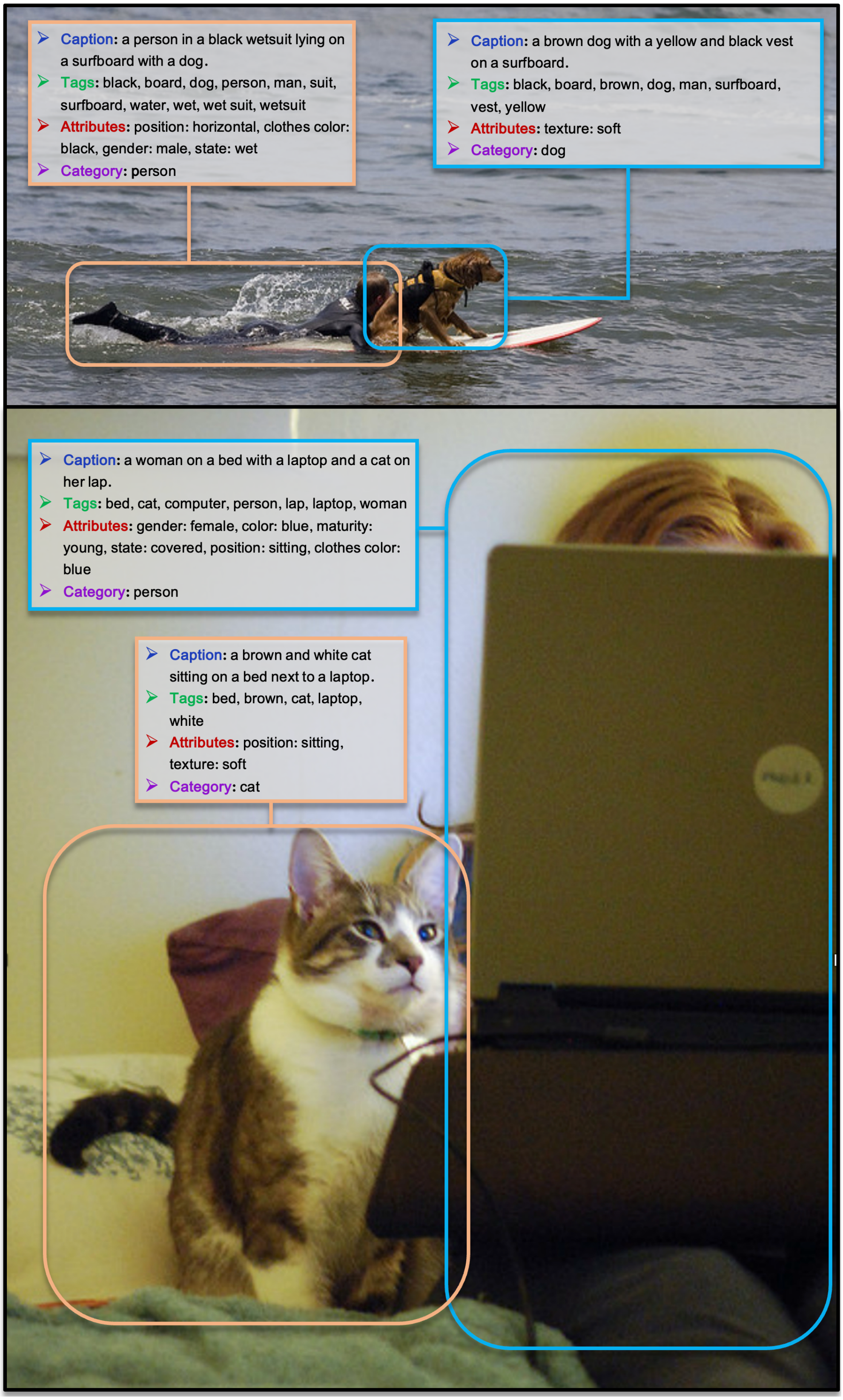

➤ Caption: a person in a black wetsuit lying on a surfboard with a dog.
➤ Tags: black, board, dog, person, man, suit, surfboard, water, wet, wet suit, wetsuit
➤ Attributes: position: horizontal, clothes color: black, gender: male, state: wet
➤ Category: person
➤ Caption: a brown dog with a yellow and black vest on a surfboard.
➤ Tags: black, board, brown, dog, man, surfboard, vest, yellow
➤ Attributes: texture: soft
➤ Category: dog
➤ Caption: a woman on a bed with a laptop and a cat on her lap.
➤ Tags: bed, cat, computer, person, lap, laptop, woman
➤ Attributes: gender: female, color: blue, maturity: young, state: covered, position: sitting, clothes color: blue
➤ Category: person
➤ Caption: a brown and white cat sitting on a bed next to a laptop.
➤ Tags: bed, brown, cat, laptop, white
➤ Attributes: position: sitting, texture: soft
➤ Category: cat

Figure 8. Illustration of DynRefer's multi-task capability. It can generate captions, tags, attributes, categories, using a single model, for any referred regions.