

# T-Rex: Counting by Visual Prompting

Qing Jiang<sup>1,2\*</sup>, Feng Li<sup>1,3\*</sup>, Tianhe Ren<sup>1</sup>, Shilong Liu<sup>1,4\*</sup>, Zhaoyang Zeng<sup>1</sup>  
 Kent Yu<sup>1</sup>, Lei Zhang<sup>1†</sup>

<sup>1</sup>International Digital Economy Academy (IDEA)

<sup>2</sup>SCUT

<sup>3</sup>HKUST

<sup>4</sup>Tsinghua

mountchicken@outlook.com, fliay@connect.ust.hk, lius120@mails.tsinghua.edu.cn

{rentianhe, zengzhaoyang, kentyu, leizhang}@idea.edu.cn

[trex-counting.github.io](http://trex-counting.github.io)

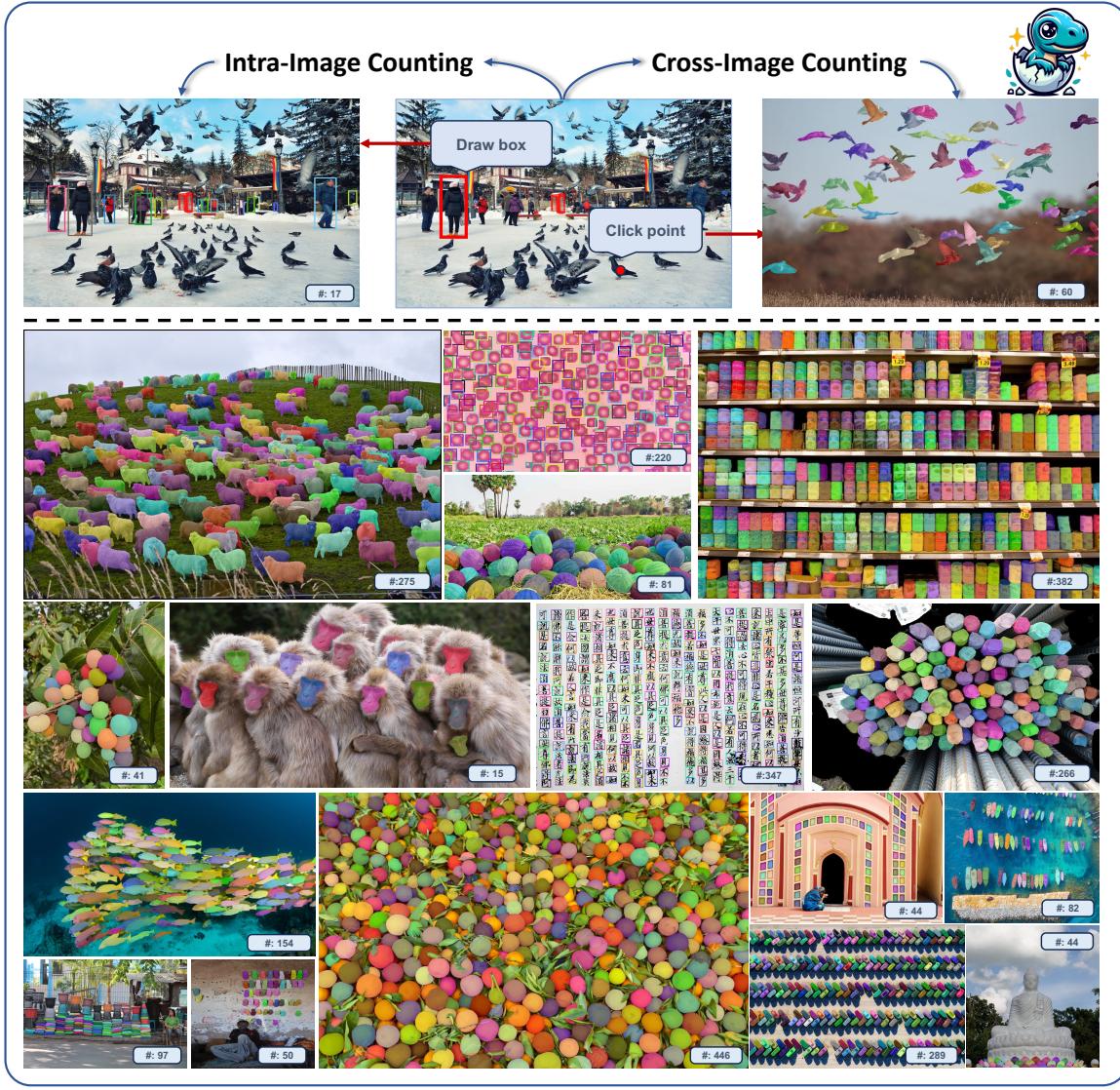


Figure 1. We introduce an interactive object counting model, T-Rex. Given boxes or points specified on the reference image, T-Rex can detect all instances on the target image that exhibit similar pattern with the specified object, which are then summed to obtain the counting result. We use SAM[12] to generate mask prompted on the detected boxes by T-Rex for better visualization.

## Abstract

We introduce T-Rex<sup>12</sup>, an interactive object counting model designed to first detect and then count any objects. We formulate object counting as an open-set object detection task with the integration of visual prompts. Users can specify the objects of interest by marking points or boxes on a reference image, and T-Rex then detects all objects with a similar pattern. Guided by the visual feedback from T-Rex, users can also interactively refine the counting results by prompting on missing or false detected objects. T-Rex has achieved state-of-the-art performance on several class-agnostic counting benchmarks. To further exploit its potential, we established a new counting benchmark encompassing diverse scenarios and challenges. Both quantitative and qualitative results show that T-Rex possesses exceptional zero-shot counting capabilities. We also present various practical application scenarios for T-Rex, illustrating its potential in the realm of visual prompting.

## 1. Introduction

Computer Vision is currently undergoing a revolution dominated by foundation models [5, 10, 12, 24, 31, 32, 35, 40] and multi-modal large-language models [2, 4, 6, 22, 25, 29, 30, 38, 39, 45, 50]. These models have demonstrated remarkable performance across a wide spectrum of tasks, including segmentation [5, 12, 16], object detection [23, 47, 52], understanding [2, 17, 22, 29, 30, 39], and generation [32, 35, 48]. However, among these tasks, the critical task of object counting has received relatively less attention.

Object counting, the task of estimating the number of specific objects present in an image, is in high demand across numerous practical fields, such as transportation, agriculture, industry, biology, etc. Existing solutions for object counting can be broadly categorized into four types:

- **As density map regression task.** A common approach [8, 11, 13, 28, 33, 37] is to regress a 2D density map, the summation of which is used as the counting result. Although effective, the *less intuitive visualization* of the density map [8] makes it difficult for users to assess the accuracy of the counting results.
- **As closed-set detection task.** Another straightforward solution involves employing a closed-set detector (E.g. YOLO [34]) to detect objects, where the summation of

<sup>\*</sup>This work was done when Qing Jiang, Feng Li, and Shilong Liu were interns at IDEA.

<sup>†</sup>Project lead.

<sup>1</sup>We named our model "T-Rex" to signify its visual detection capability, akin to the Tyrannosaurus Rex's exceptional visual acuity, which is estimated to be 13 times better than that of a human. [Wikipedia link](#).

<sup>2</sup>Technical Report. Work in progress.

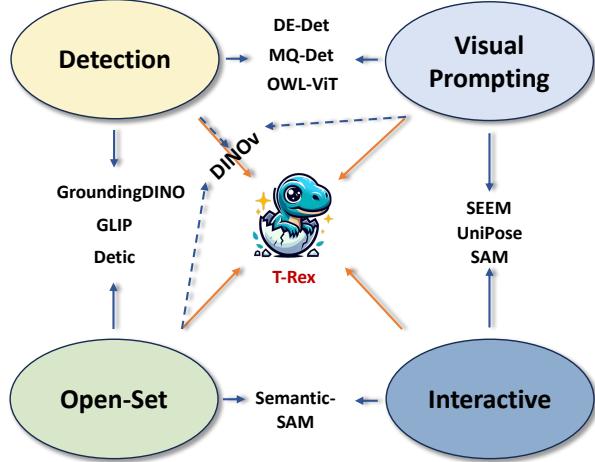


Figure 2. T-Rex is an object counting model, which is characterized by four features: detection-based, visual promptable, interactive, and open-set. Listed methods are: Grounding DINO [23], GLIP [18], Detic [51], SAM [12], Semantic-SAM [15], SEEM [53], UniPose [42], DE-ViT [49], MQ-Det [41], OWL-ViT [26].

the number of detected boxes serves as the counting result. However, *limited by the fixed categories*, this method requires data re-collection and re-training efforts for novel categories, which is time-consuming and labor-intensive.

- **As open-vocabulary detection task.** To overcome the limitations of closed-set detection methods, an alternative approach is to adapt open vocabulary detector (E.g. Grounding DINO [23]) to detect arbitrary objects through text prompts. However, the task of counting poses a significant challenge as many objects *do not have concise textual descriptions*, making object specification by text difficult.
- **As MLLM QA task.** Multi-modal Large Language Models (MLLM) can also be used for object counting through question answering [14, 43]. However, the issue of hallucination [9] in multi-modal LLMs affects the *confidence level* of their counting results, as users may be skeptical of a numerical output from an MLLM without additional supporting evidence.

By highlighting the limitations of existing counting solutions, we argue that a practical counting system should possess the following four properties: a) **Intuitive Visual Feedback:** It should provide highly interpretable visual feedback (e.g. bounding box), allowing users to verify the accuracy of the counting results. b) **Open-Set:** It should be capable of counting any objects, without constraints on predefined categories. c) **Visual Promptable:** It should allow users to specify the objects for counting through visual examples, given the limitation of text to describe various objects. d) **Interactive:** It should enable users to actively participate in the counting process to correct errors made

by the model.

Guided by this design philosophy, we develop an detection-based counting model, called T-Rex, as shown in Fig. 2. Users can specify the object of interest by marking boxes or points on the reference image. T-Rex, in return, detects all instances with a similar pattern in the target image, and the cumulative sum of the detected boxes represents the counting result. With the visual feedback from T-Rex, users can interactively add additional prompts on missed or falsely-detected objects. This interactive process allows for continual refinement over T-Rex’s prediction, empowering users to confidently access the accuracy of the counting results. Notably, this interactive process remains fast and resource-efficient, as each round of interaction only requires forwarding the decoder of T-Rex.

T-Rex has achieved state-of-the-art results on two counting benchmarks [28, 33]. To further measure its potential, we introduce a new counting benchmark, CA-44, which comprises 44 datasets across eight domains, presenting diverse and formidable challenges. Our experimental findings demonstrate that T-Rex possesses strong zero-shot counting capabilities and can achieve good performance in various scenarios. Finally, we explore a wide range of application scenes of T-Rex. With its versatile counting capabilities and interactive features, T-Rex has the potential to make substantial contributions to various domains, such as retail, transportation, agriculture, industry, livestock, medicine, etc.

## 2. Related Works

**Object Counting.** Methods for object counting can be divided into class-specific and class-agnostic. Class-specific approaches typically use object detection models to count specific categories, such as people [1, 46], cars [7, 27], or animals [3]. These methods are limited to predefined classes and require additional data labeling for new categories, which is time-consuming and labor-intensive. To overcome the limitations of close-set object detector, class-agnostic methods [8, 11, 28, 33, 37] have been developed to regress density map [13] based on the correlation feature between image and a few visual exemplars. However, these techniques often lack an intuitive visualization, making it difficult for users to verify the model’s accuracy.

**Interactive Models.** Interactive models have shown significant promise in aligning human intentions within the field of computer vision. SAM[12] presents an interactive segmentation model capable of accommodating point, box, and text-based input, achieving remarkable zero-shot segmentation by leveraging an large-scale dataset SA-1B[12]. SEEM [53] and Semantic-SAM [15] have extended to more general segmentation models that can output the semantic meaning for the segmented mask. In contrast, the field of interactive object detection is less explored.

## 3. Overview of T-Rex

We briefly introduce the T-Rex model. T-Rex comprises of three components, including an image encoder, a prompt encoder and a box decoder, as illustrated in Fig. 3. Given a target image input  $\mathbf{I}_{tgt}$  and optionally a reference image input  $\mathbf{I}_{ref}$  (the target image can also serve as the reference image in the absence of a separate reference image), the image encoder first extracts the visual features  $\mathbf{E}_{tgt}, \mathbf{E}_{ref}$ . Then, using the user-drawn boxes or points as prompts  $\mathbf{P}$  for the target object on the reference image, the prompt encoder extracts the encoded visual prompt  $\mathbf{P}_{enc}$  from the reference image feature  $\mathbf{E}_{ref}$ . Finally, the box decoder combines the target image feature  $\mathbf{E}_{tgt}$  and the encoded visual prompt  $\mathbf{P}_{enc}$  as inputs, outputting detected boxes  $\mathbf{B}$  along with their associated confidence scores  $\mathbf{S}$ . A predetermined score threshold is applied to filter the detected boxes, and the remaining boxes are summed to produce the final object count.

$$\mathbf{E}_{tgt} = \text{ImageEncoder}(\mathbf{I}_{tgt}) \quad (1)$$

$$\mathbf{E}_{ref} = \text{ImageEncoder}(\mathbf{I}_{ref}) \quad (2)$$

$$\mathbf{P}_{enc} = \text{PromptEncoder}(\mathbf{P}, \mathbf{E}_{ref}) \quad (3)$$

$$\mathbf{B}, \mathbf{S} = \text{BoxDecoder}(\mathbf{P}_{enc}, \mathbf{E}_{tgt}) \quad (4)$$

$$\#\text{Count} = \sum \text{ThreshFilter}(\mathbf{B}, \mathbf{S}) \quad (5)$$

### 3.1. Workflows

T-Rex offers three major interactive workflows as shown in Fig. 4. We explain each workflow and its application below.

**Positive-only Prompt Mode.** In most counting scenarios, users typically only need to click once or draw one box, and T-Rex can effectively detect all objects with a similar pattern. However, in cases involving dense and small objects, a single round of prompting may be insufficient. In such cases, users have the option to incorporate additional prompts to the missed region, based on the visual feedback from T-Rex. This iterative refinement approach allows for more accurate counting results.

**Positive with Negative Prompt Mode.** In scenarios where interference from other similar objects is present, T-Rex may generate falsely detected boxes. As illustrated in Fig. 4, when a prompt is directed at a green apple, T-Rex might erroneously detect the orange tangerine due to the strong geometric resemblance between these two object types. In such cases, users can rectify the counting result by adding negative prompts to the falsely detected object.

**Cross-Image Prompt Mode.** T-Rex additionally offers support for cross-image counting, allowing for the combination of different reference and target images. This functionality proves especially useful in automatic annotation scenarios, where users need to prompt only once on one image, and T-Rex can automatically annotate other images that exhibit a similar object pattern to the prompted image.

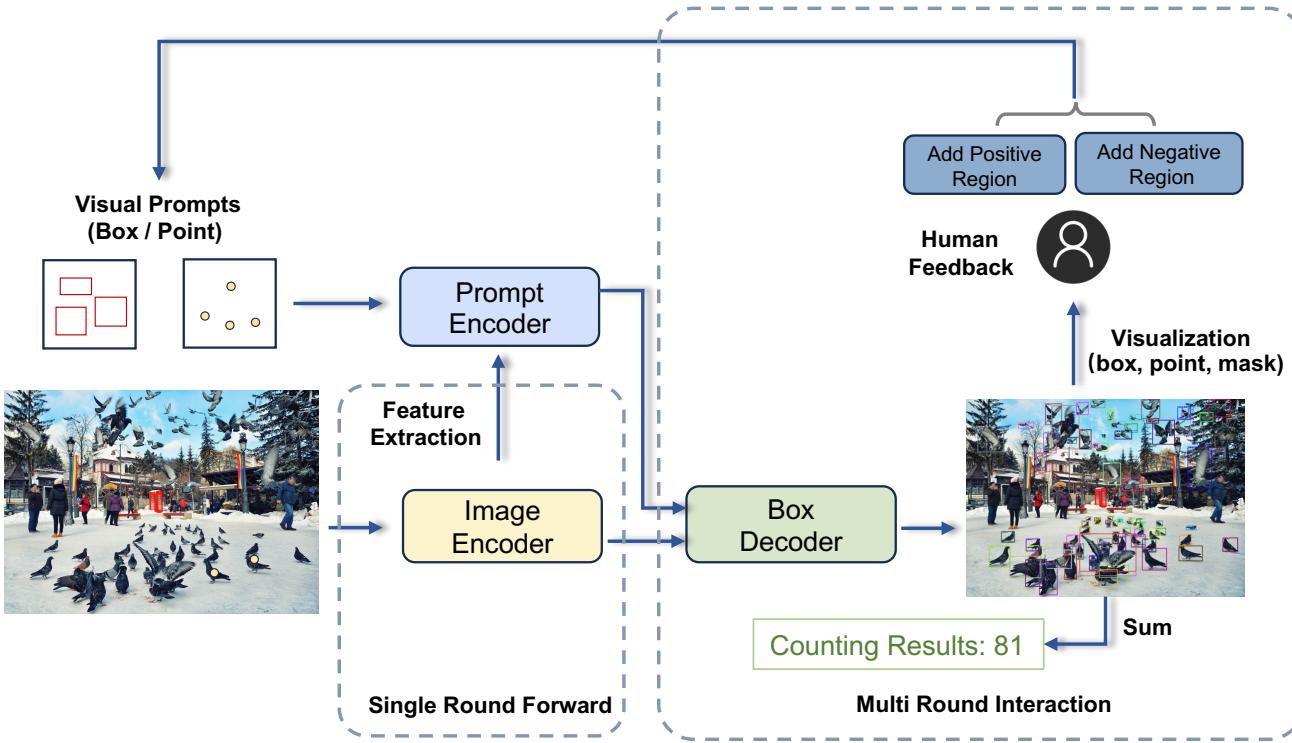


Figure 3. Overview of the T-Rex model. T-Rex is a detection-based model comprising an image encoder to extract image feature, a prompt encoder to encode visual prompts (points or boxes) provided by users, and a box decoder to output the detected boxes.



Figure 4. T-Rex offers three major interactive workflows, which are applicable to most scenarios in real-world applications.

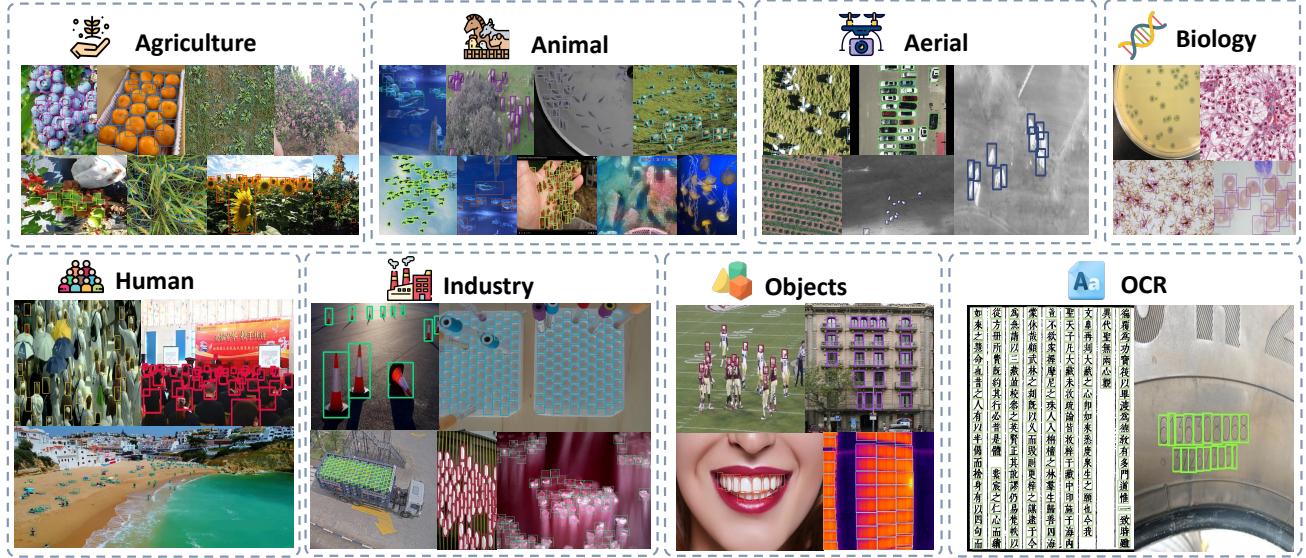


Figure 5. An Overview of the proposed CA-44 benchmark. CA-44 consists of 44 datasets across eight domains and mainly comprises images with small and densely packed objects.

### 3.2. Discussion

T-Rex is essentially an open-set object detection model. Compared with open-vocabulary object detectors [18, 23, 26, 51] that rely on text-based prompt, T-Rex adopts visual prompt instead. Since in many real-world counting applications, text descriptions may not sufficiently capture all object details, and employing visual prompts offers a more direct and versatile alternative.

In the context of the object counting task, a paramount consideration revolves around the need for highly reliable prediction from the model. Given that the counting results are represented as statistical values, even a minor discrepancy in the predicted value signifies a failure counting. Hence, we design T-Rex to be interactive, allowing users to iteratively rectify counting results based on the visual feedback, thus enhancing the counting accuracy. Regarding the structure of model, T-Rex requires only a single forward pass through the Image Encoder, while subsequent rounds of interaction involve only the Prompt Encoder and Box Decoder. This streamlined approach ensures that the entire interaction process remains lightweight and fast.

## 4. Count Anything Benchmark

To conduct a holistic performance evaluation of the T-Rex model, we develop a new object counting benchmark named CA-44. This benchmark includes a total of 44 datasets covering eight distinct domains, as depicted in Fig. 5.

**Dataset Distribution.** The majority of the datasets included in CA-44 were collected from [Roboflow](#) and underwent additional filtering procedures applied. For instance,

Category	# Datasets	# Images	# Instances
Industrial	6	1,781	114,688
Object	4	3,644	94,328
Biology	4	1,834	63,235
OCR	2	305	86,986
Animal	8	4,674	122,069
Human	3	1,119	44,212
Aerial	5	5,011	156,824
Agriculture	12	11,717	488,719
Total	44	30,085	1,171,061

Table 1. Dataset distribution of CA-44.

we eliminated images containing fewer than 10 instances, since object counting is mostly focused on dense scenes. The composition of the CA-44 dataset is detailed in Table 1.

**Statistics.** As illustrated in Fig. 6, the CA-44 benchmark primarily features images with small and densely packed objects. These characteristics reflect the common attributes of scenes in the object counting domain.

## 5. Experiment Results

**Settings.** In addition to our proposed CA-44 benchmark, we also conduct evaluations on the commonly-used counting dataset FSC147 [33] and the more challenging dataset FSCD-LVIS [28]. FSC147 comprises 147 categories of objects and 1190 images in the test set and FSCD-LVIS comprises 377 categories and 1014 images in the test set. Both two datasets provide three bounding boxes of exemplar objects for each image, which we will use as the visual prompt for T-Rex.

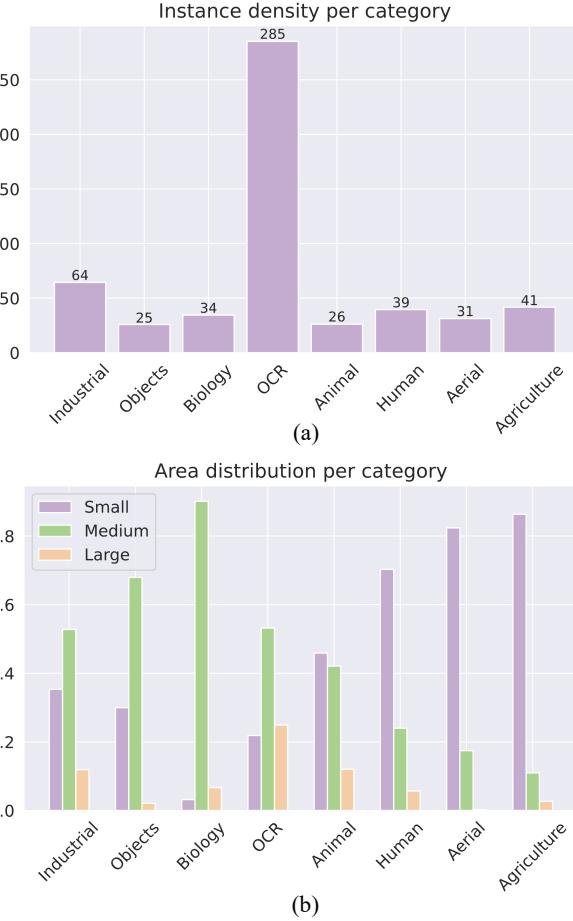


Figure 6. Statistics of the CA-44 benchmark, highlighting the prevalence of small and dense objects. Object size categorization follows the COCO dataset [20], where objects with an area smaller than  $32^2$  are classified as small, those with an area between  $32^2$  and  $96^2$  as medium, and objects with area greater than  $96^2$  as large.

**Metrics.** We adopt two metrics for evaluation: the Mean Average Error (MAE) metric, a widely employed standard in object counting and the Normalized Mean Average Error (NMAE) metric for more intuitive results. The mathematical expressions for the two metrics are as follows:

$$\text{MAE} = \frac{1}{J} \sum_{j=1}^J |c_j^* - c_j| \quad (6)$$

$$\text{NMAE} = \frac{1}{J} \sum_{j=1}^J \frac{|c_j^* - c_j|}{c_j^*} \quad (7)$$

where  $J$  represents the total number of test images,  $c_j^*$  and  $c_j$  denote the ground truth (GT) and the predicted number of objects for image  $j$ , respectively.

Type	Method	MAE $\downarrow$
Density Map	FamNet [33]	26.76
	BMNet+ [36]	16.89
	LaoNet [19]	15.78
	CountTR [21]	12.06
	LOCA [37]	12.53
Detection	T-Rex (Ours)	<b>10.59</b>

Table 2. One-shot counting evaluation on FSC147 test-set. One-shot indicates that each image utilizes one exemplar box as visual prompt.

Type	Method	MAE $\downarrow$
Density Map	FamNet [33]	22.08
	BMNet+ [36]	14.62
	SAFECount [44]	14.32
	CountTR [21]	11.95
	LOCA [37]	10.79
Detection	T-Rex (Ours)	<b>8.72</b>

Table 3. Three-shot counting evaluation on FSC147 test-set. Three-shot indicates that each image utilizes three exemplar box as visual prompts.

Type	Method	MAE $\downarrow$	NMAE $\downarrow$
Density Map	FamNet [33]	22.08	1.82
	Counting-DETR [28]	18.51	0.45
Detection	T-Rex (Ours)	<b>12.87</b>	<b>0.27</b>

Table 4. Three-shot counting evaluation on FSCD-LVIS test-set.

## 5.1. Results on FSC147 and FSCD-LVIS

The results on FSC147 and FSCD-LVIS are presented in Table 2, Table 3, and Table 4. T-Rex demonstrates state-of-the-art performance when compared to other density map regression-based methods, in both one-shot and three-shot settings. Beyond competitive performance, T-Rex also provides a user-friendly interactive counting interface. As a detection-based method, T-Rex offers intuitive visual feedback, allowing users to iteratively refine counting results and make informed judgments regarding their completeness. This interactive process enables T-Rex to achieve high reliability, contrasting with the less robust density map regression methods.

## 5.2. Results on CA-44

Results on CA-44 are visually presented in Fig. 7. T-Rex significantly outperforms other methods, including the open-vocabulary detector Grounding DINO [23] and various density map regression-based methods, across all categories within CA-44. This showcases T-Rex’s exceptional zero-shot counting capabilities and its adaptability at tackling challenges spanning diverse domains. In comparison to the open-vocabulary detector Grounding DINO, T-Rex with

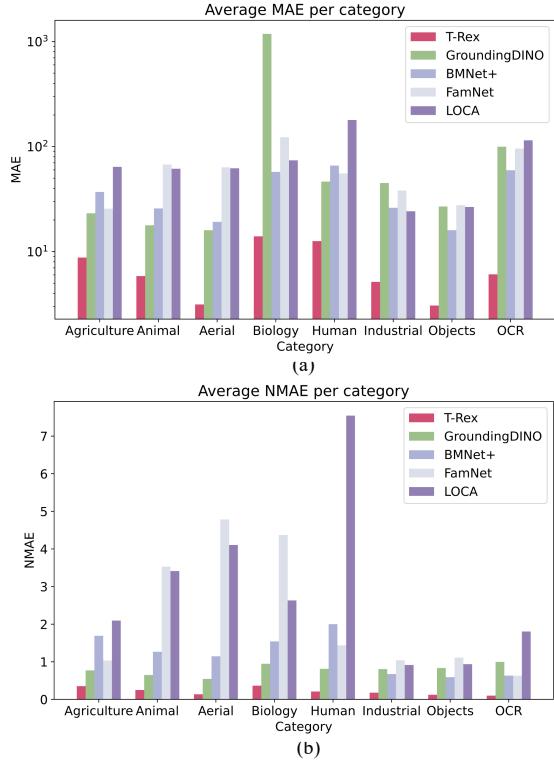


Figure 7. Results on Full CA-44. We compare T-Rex with open-vocabulary detector Grounding DINO [23] and density map regression-based methods [33, 36, 37].

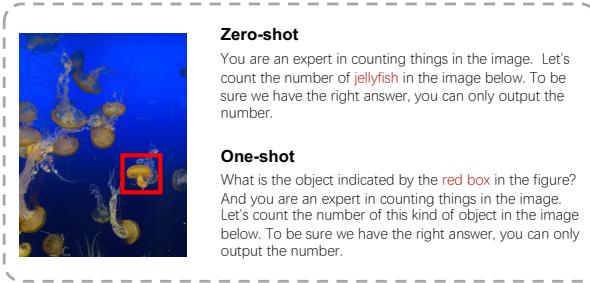


Figure 8. Prompts that are used to evaluate the counting ability of GPT-4V.

visual prompt is more competitive. This underscores the limitations of text in providing sufficient descriptions, highlighting the significance of introducing visual prompts as a method.

We have also conducted a comparison with the state-of-the-art multi-modality model, GPT-4V [29], which has previously demonstrated counting capabilities [43]. For an optimized evaluation, we selected a total of 100 images from CA-44 for testing. We tested GPT-4V in two settings: a zero-shot approach, where we inform GPT-4V about the objects in the image for counting, and a one-shot approach, where we annotate the objects in the image with bounding

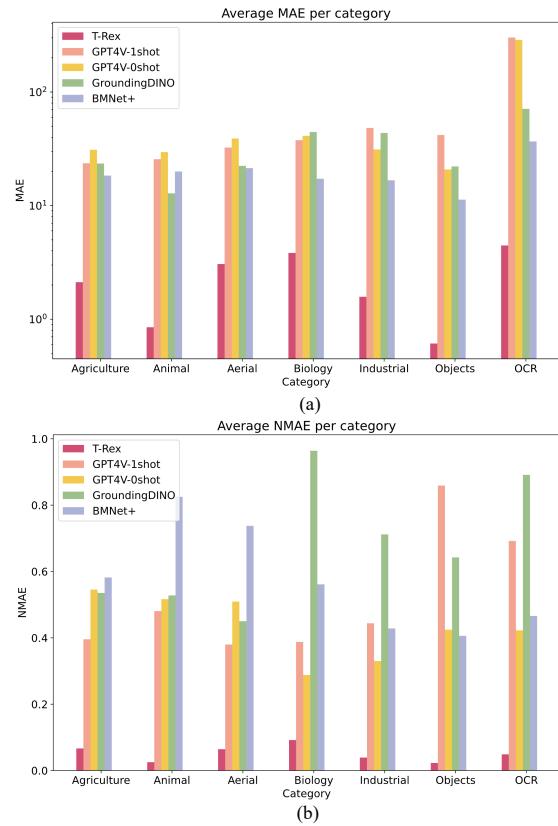


Figure 9. Results on subset of CA-44, sampling a total of 100 images from each category. GPT4V-0shot denotes zero-shot approach where we inform GPT-4V about the objects in the image and instruct it to count them. GPT4V-1shot denotes the one-shot approach where we annotate the objects in the image with bounding boxes and request GPT-4V to count them.

boxes for GPT-4V to count. The One-shot approach is similar to T-Rex in that both methods rely on visual prompts. Prompts used are detailed in Fig. 8. Results presented in Fig. 9 illustrate that T-Rex outperforms GPT-4V in counting accuracy, suggesting that while large multi-modality models perform well in understanding tasks, they may still lack the nuanced capabilities for accurate object perception.

## 6. Conclusion

In this paper, we have introduced T-Rex, an innovative model for interactive object counting, characterized by its ability to detect and count objects using visual prompt. T-Rex represents a significant advancement in visual prompting methodologies within computer vision, paralleling the successes observed in NLP with LLMs facilitating Human-AI interactions through text prompts. This parallel suggests vast possibilities, that the application of visual prompts in computer vision could herald a comparable breakthrough like those in NLP.

## 7. Limitations

T-Rex , in its intial version, has several limitations. We list a series of failure cases in Fig. 22, where T-Rex may perform poorly. **Single-Target Scenes.** When only a single prompt is used against the background, T-Rex tends to misidentify dense objects clusters. **Dense Multi-Object Scenes.** T-Rex struggles in scenes with densely populated multi-object types, often leading to false detections. Addressing this issue may require either multiple iterations of prompting or the use of negative prompts. **Cross-Image Workflow.** A notable limitation emerges in cross-image workflow, especially when T-Rex is applied to scenes with a single target. In such scenarios, there is a significant risk of over-fitting, where T-Rex tends to ignore the user’s prompt on the reference image. For example, even when prompted on tomatoes, T-Rex may still detect silkworm eggs. We will continue to improve the performance and robustness of T-Rex.

## 8. Acknowledgments

We would like to express our deepest gratitude to multiple teams within IDEA for their substantial support in the T-Rex project. We sincerely appreciate the CVR team, whose essential contributions and technical expertise were pivotal in realizing the project’s goals. We thank Wei Liu, Xiaohui Wang, and Yakun Hu from the Product team for their strategic insights and innovative input in the development of the demo. Appreciation is also extended to Yuanhao Zhu and Ce Feng from the Front-End team for their technical excellence and dedication. The robust solutions provided by Weiqiang Hu, Xiaoke Jiang, and Zhiqiang Li from the Back-End team were also crucial in supporting the project’s infrastructure. We also thank Jie Yang for helpful discussion and Ling-Hao Chen for helping in video demos.

## 9. Potential Applications

In this section, we explore the application of T-Rex in various fields. As a detection-based model, it can be used as an object counter, as well as an automatic-annotation tool.

- Agriculture: Fig. 10.
- Industry: Fig. 11, Fig. 12, Fig. 13.
- Livestock and Wild Animal: Fig. 14, Fig. 15.
- Biology: Fig. 16.
- Medicine: Fig. 17.
- OCR: Fig. 17.
- Retail: Fig. 18.
- Electronic: Fig. 19.
- Transportation: Fig. 19.
- Logistics: Fig. 20.
- Human: Fig. 20.
- Cartoon: Fig. 21

## References

- [1] Shahira Abousamra, Minh Hoai, Dimitris Samaras, and Chao Chen. Localization in the crowd with topological constraints. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 872–881, 2021. 3
- [2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 35:23716–23736, 2022. 2
- [3] Carlos Arteta, Victor Lempitsky, and Andrew Zisserman. Counting in the wild. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VII 14*, pages 483–498. Springer, 2016. 3
- [4] Xi Chen, Xiao Wang, Lucas Beyer, Alexander Kolesnikov, Jialin Wu, Paul Voigtlaender, Basil Mustafa, Sebastian Goodman, Ibrahim Alabdulmohsin, Piotr Padlewski, et al. Pali-3 vision language models: Smaller, faster, stronger. *arXiv preprint arXiv:2310.09199*, 2023. 2
- [5] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1290–1299, 2022. 2
- [6] Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15180–15190, 2023. 2
- [7] Meng-Ru Hsieh, Yen-Liang Lin, and Winston H Hsu. Drone-based object counting by spatially regularized regional proposal network. In *Proceedings of the IEEE international conference on computer vision*, pages 4145–4153, 2017. 3
- [8] Yifeng Huang, Viresh Ranjan, and Minh Hoai. Interactive class-agnostic object counting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22312–22322, 2023. 2, 3
- [9] Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38, 2023. 2
- [10] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021. 2
- [11] Ruixiang Jiang, Lingbo Liu, and Changwen Chen. Clip-count: Towards text-guided zero-shot object counting. *arXiv preprint arXiv:2305.07304*, 2023. 2, 3
- [12] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick. Segment anything. In *Proceedings of the*

- IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4015–4026, 2023. 1, 2, 3
- [13] Victor Lempitsky and Andrew Zisserman. Learning to count objects in images. *Advances in neural information processing systems*, 23, 2010. 2, 3
- [14] Bo Li, Peiyuan Zhang, Jingkang Yang, Yuanhan Zhang, Fanyi Pu, and Ziwei Liu. Otterhd: A high-resolution multi-modality model. *arXiv preprint arXiv:2311.04219*, 2023. 2
- [15] Feng Li, Hao Zhang, Peize Sun, Xueyan Zou, Shilong Liu, Jianwei Yang, Chunyuan Li, Lei Zhang, and Jianfeng Gao. Semantic-sam: Segment and recognize anything at any granularity. *arXiv preprint arXiv:2307.04767*, 2023. 2, 3
- [16] Feng Li, Hao Zhang, Huaiuze Xu, Shilong Liu, Lei Zhang, Lionel M Ni, and Heung-Yeung Shum. Mask dino: Towards a unified transformer-based framework for object detection and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3041–3050, 2023. 2
- [17] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023. 2
- [18] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10965–10975, 2022. 2, 5
- [19] Hui Lin, Xiaopeng Hong, and Yabin Wang. Object counting: You only need to look at one. *arXiv preprint arXiv:2112.05993*, 2021. 6
- [20] 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 *European Conference on Computer Vision*, pages 740–755. Springer, 2014. 6
- [21] Chang Liu, Yujie Zhong, Andrew Zisserman, and Weidi Xie. Countr: Transformer-based generalised visual counting. *arXiv preprint arXiv:2208.13721*, 2022. 6
- [22] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023. 2
- [23] Shilong Liu, ZhaoYang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023. 2, 5, 6, 7
- [24] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12009–12019, 2022. 2
- [25] Tengchao Lv, Yupan Huang, Jingye Chen, Lei Cui, Shuming Ma, Yaoyao Chang, Shaohan Huang, Wenhui Wang, Li Dong, Weiyao Luo, et al. Kosmos-2.5: A multimodal literate model. *arXiv preprint arXiv:2309.11419*, 2023. 2
- [26] Matthias Minderer, Alexey Gritsenko, Austin Stone, Maxim Neumann, Dirk Weissenborn, Alexey Dosovitskiy, Aravindh Mahendran, Anurag Arnab, Mostafa Dehghani, Zhuoran Shen, et al. Simple open-vocabulary object detection. In *European Conference on Computer Vision*, pages 728–755. Springer, 2022. 2, 5
- [27] T Nathan Mundhenk, Goran Konjevod, Wesam A Sakla, and Kofi Boakye. A large contextual dataset for classification, detection and counting of cars with deep learning. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part III* 14, pages 785–800. Springer, 2016. 3
- [28] Thanh Nguyen, Chau Pham, Khoi Nguyen, and Minh Hoai. Few-shot object counting and detection. In *European Conference on Computer Vision*, pages 348–365. Springer, 2022. 2, 3, 5, 6
- [29] OpenAI. Gpt-4v(ision) system card. 2023. 2, 7
- [30] Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. *arXiv preprint arXiv:2306.14824*, 2023. 2
- [31] 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. 2
- [32] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 2
- [33] Vireesh Ranjan, Udbhav Sharma, Thu Nguyen, and Minh Hoai. Learning to count everything. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3394–3403, 2021. 2, 3, 5, 6, 7
- [34] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016. 2
- [35] 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. 2
- [36] Min Shi, Hao Lu, Chen Feng, Chengxin Liu, and Zhiguo Cao. Represent, compare, and learn: A similarity-aware framework for class-agnostic counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9529–9538, 2022. 6, 7
- [37] Nikola Đukić, Alan Lukežić, Vitjan Zavrtanik, and Matej Kristan. A low-shot object counting network with iterative prototype adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18872–18881, 2023. 2, 3, 6, 7
- [38] Jianfeng Wang, Zhengyuan Yang, Xiaowei Hu, Linjie Li, Kevin Lin, Zhe Gan, Zicheng Liu, Ce Liu, and Lijuan Wang. Git: A generative image-to-text transformer for vision and language. *arXiv preprint arXiv:2205.14100*, 2022. 2
- [39] Wenhui Wang, Zhe Chen, Xiaokang Chen, Jianne Wu, Xizhou Zhu, Gang Zeng, Ping Luo, Tong Lu, Jie Zhou, Yu

- Qiao, et al. Visionllm: Large language model is also an open-ended decoder for vision-centric tasks. *arXiv preprint arXiv:2305.11175*, 2023. 2
- [40] Wenhui Wang, Jifeng Dai, Zhe Chen, Zhenhang Huang, Zhiqi Li, Xizhou Zhu, Xiaowei Hu, Tong Lu, Lewei Lu, Hongsheng Li, et al. Internimage: Exploring large-scale vision foundation models with deformable convolutions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14408–14419, 2023. 2
- [41] Yifan Xu, Mengdan Zhang, Chaoyou Fu, Peixian Chen, Xiaoshan Yang, Ke Li, and Changsheng Xu. Multi-modal queried object detection in the wild. *arXiv preprint arXiv:2305.18980*, 2023. 2
- [42] Jie Yang, Ailing Zeng, Ruimao Zhang, and Lei Zhang. Unipose: Detection any keypoints. *arXiv preprint arXiv:2310.08530*, 2023. 2
- [43] Zhengyuan Yang, Linjie Li, Kevin Lin, Jianfeng Wang, Chung-Ching Lin, Zicheng Liu, and Lijuan Wang. The dawn of lmms: Preliminary explorations with gpt-4v (ision). *arXiv preprint arXiv:2309.17421*, 9, 2023. 2, 7
- [44] Zhiyuan You, Kai Yang, Wenhan Luo, Xin Lu, Lei Cui, and Xinyi Le. Few-shot object counting with similarity-aware feature enhancement. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 6315–6324, 2023. 6
- [45] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022. 2
- [46] Cong Zhang, Hongsheng Li, Xiaogang Wang, and Xiaokang Yang. Cross-scene crowd counting via deep convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 833–841, 2015. 3
- [47] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv preprint arXiv:2203.03605*, 2022. 2
- [48] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023. 2
- [49] Xinyu Zhang, Yuting Wang, and Abdeslam Boularias. Detect every thing with few examples, 2023. 2
- [50] Yiyuan Zhang, Kaixiong Gong, Kaipeng Zhang, Hongsheng Li, Yu Qiao, Wanli Ouyang, and Xiangyu Yue. Metatransformer: A unified framework for multimodal learning. *arXiv preprint arXiv:2307.10802*, 2023. 2
- [51] Xingyi Zhou, Rohit Girdhar, Armand Joulin, Philipp Krähenbühl, and Ishan Misra. Detecting twenty-thousand classes using image-level supervision. In *European Conference on Computer Vision*, pages 350–368. Springer, 2022. 2, 5
- [52] Zhuofan Zong, Guanglu Song, and Yu Liu. Detrs with collaborative hybrid assignments training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6748–6758, 2023. 2
- [53] Xueyan Zou, Jianwei Yang, Hao Zhang, Feng Li, Linjie Li, Jianfeng Gao, and Yong Jae Lee. Segment everything everywhere all at once. *arXiv preprint arXiv:2304.06718*, 2023. 2, 3

# Agriculture



Figure 10. T-Rex applied to agriculture. To more intuitively visualize overlapping and dense objects, we prompt SAM with the boxes predicted by T-Rex to obtain segmentation masks. The red boxes on the original image are visual prompts for positive examples and the blue boxes are visual prompts for negative examples.

# Industry

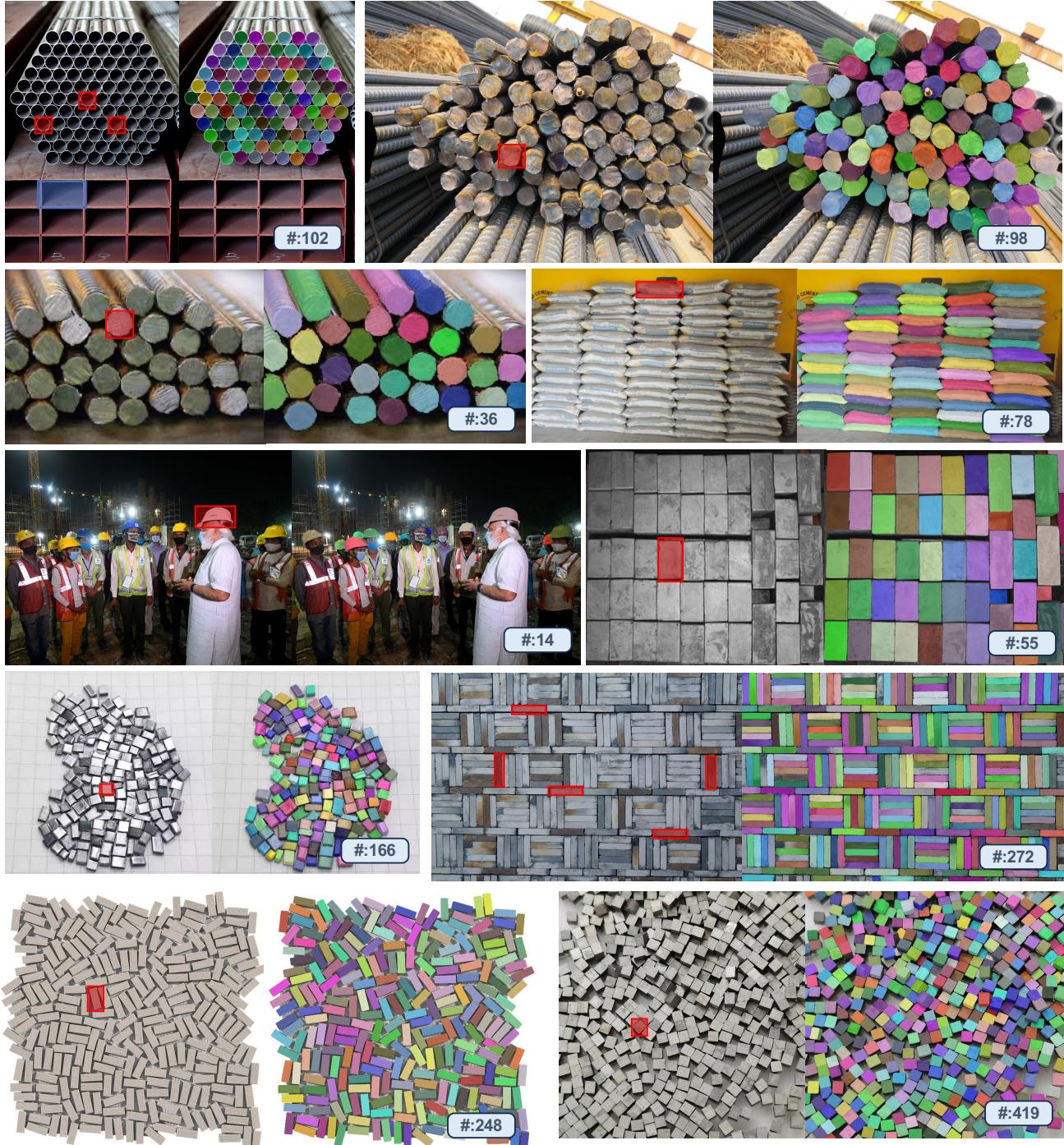


Figure 11. T-Rex applied to industry. To more intuitively visualize overlapping and dense objects, we prompt SAM with the boxes predicted by T-Rex to obtain segmentation masks. The red boxes on the original image are visual prompts for positive examples and the blue boxes are visual prompts for negative examples.

# Industry

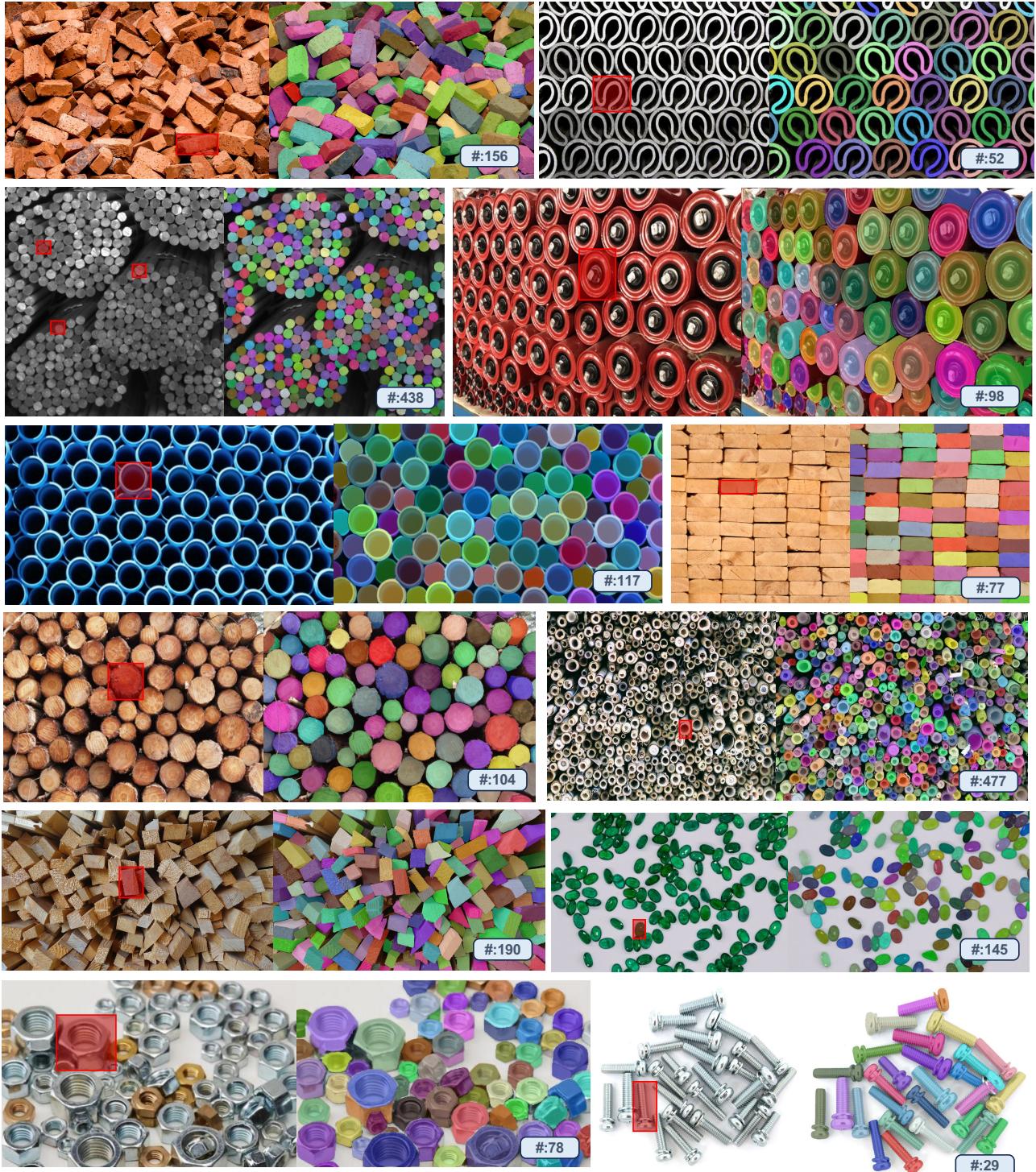


Figure 12. T-Rex applied to industry. To more intuitively visualize overlapping and dense objects, we prompt SAM with the boxes predicted by T-Rex to obtain segmentation masks. The red boxes on the original image are visual prompts for positive examples and the blue boxes are visual prompts for negative examples.

### Cross-Image Counting

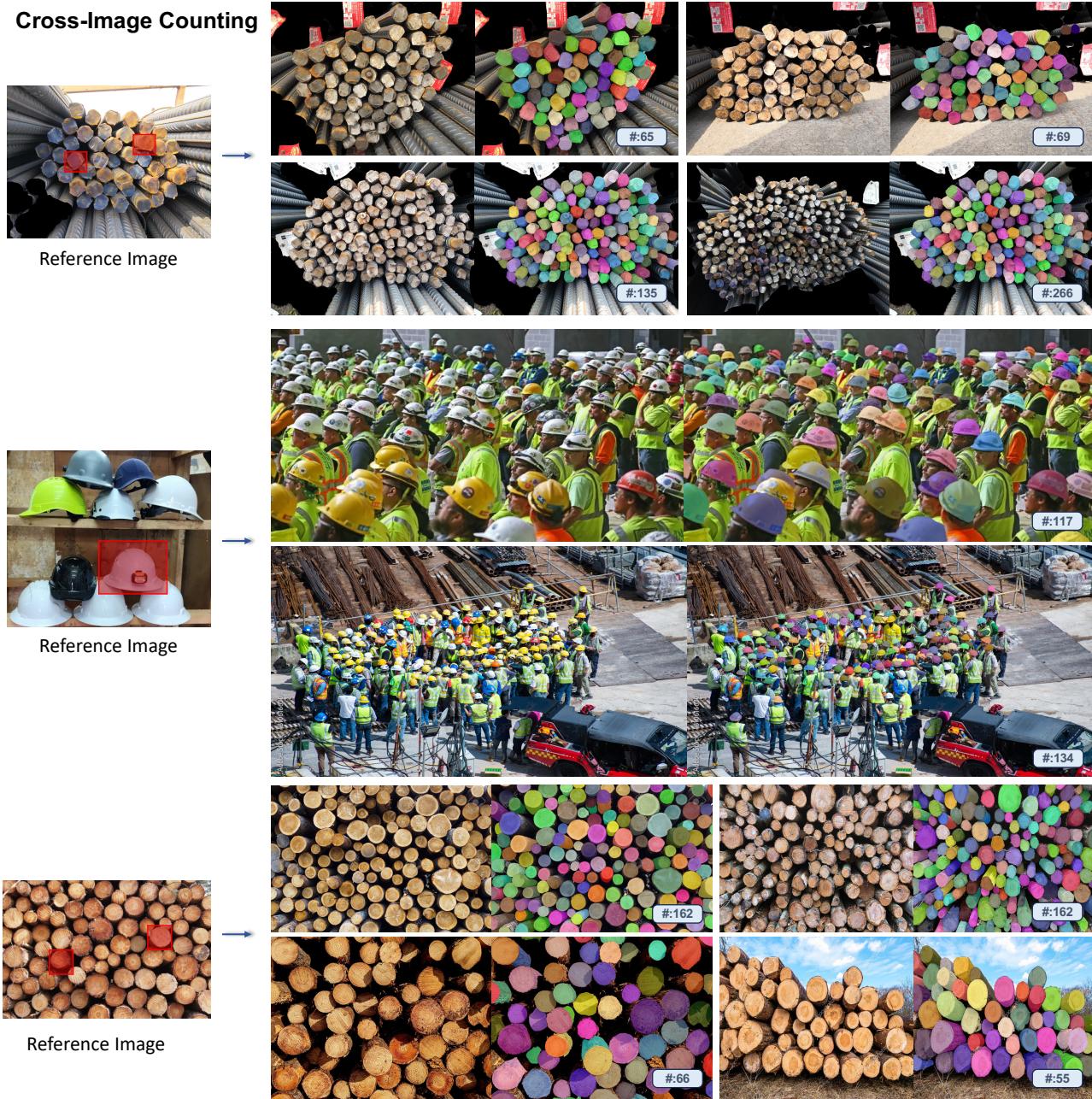


Figure 13. T-Rex applied to industry in cross-image reference mode. To more intuitively visualize overlapping and dense objects, we prompt SAM with the boxes predicted by T-Rex to obtain segmentation masks. The red boxes on the original image are visual prompts for positive examples and the blue boxes are visual prompts for negative examples.

## Livestock / Wild Animal



Figure 14. T-Rex applied to livestock and wild animal. To more intuitively visualize overlapping and dense objects, we prompt SAM with the boxes predicted by T-Rex to obtain segmentation masks. The red boxes on the original image are visual prompts for positive examples and the blue boxes are visual prompts for negative examples.

## Livestock / Wild Animal

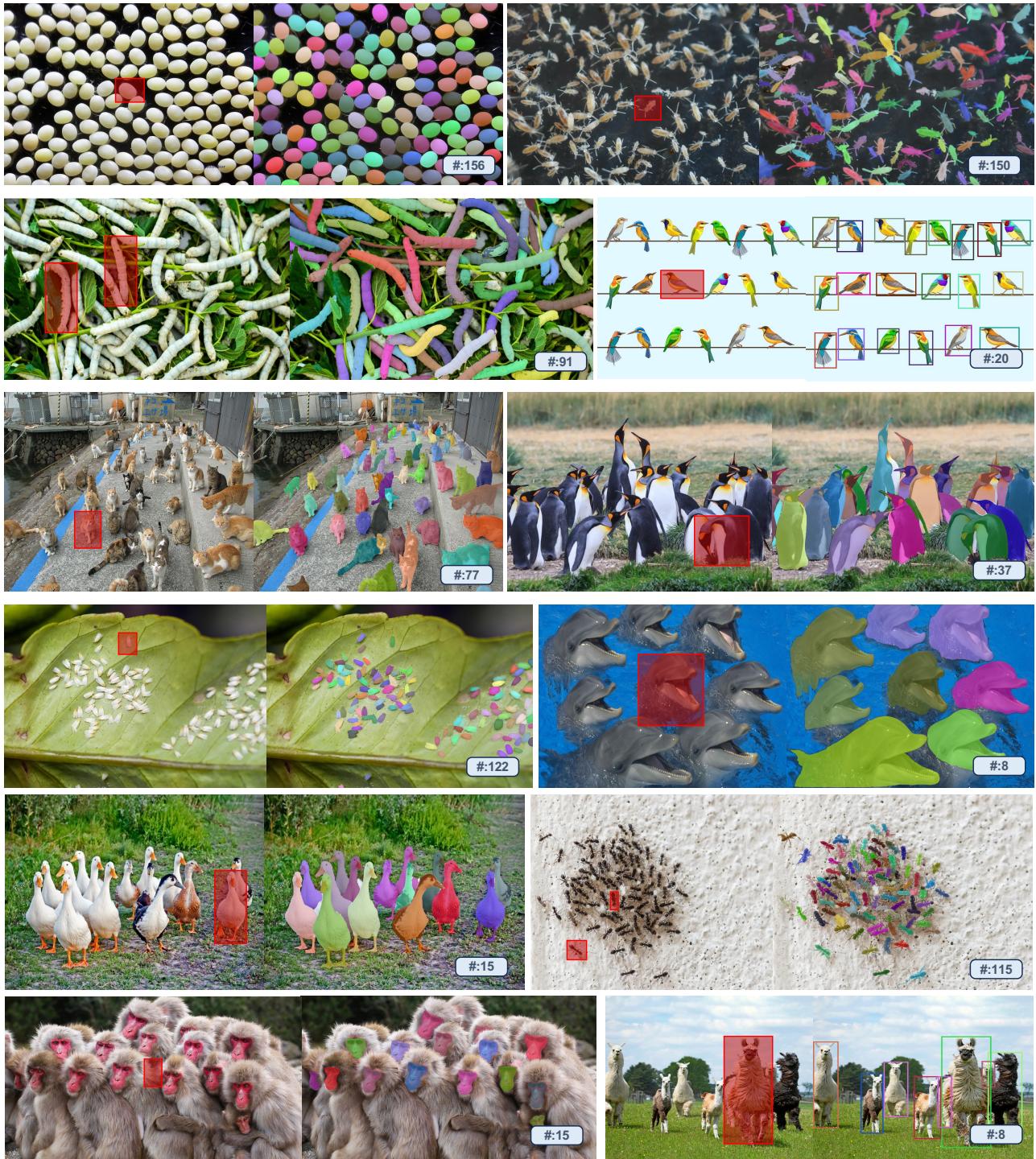


Figure 15. T-Rex applied to livestock and wild animal. To more intuitively visualize overlapping and dense objects, we prompt SAM with the boxes predicted by T-Rex to obtain segmentation masks. The red boxes on the original image are visual prompts for positive examples and the blue boxes are visual prompts for negative examples.

# Biology

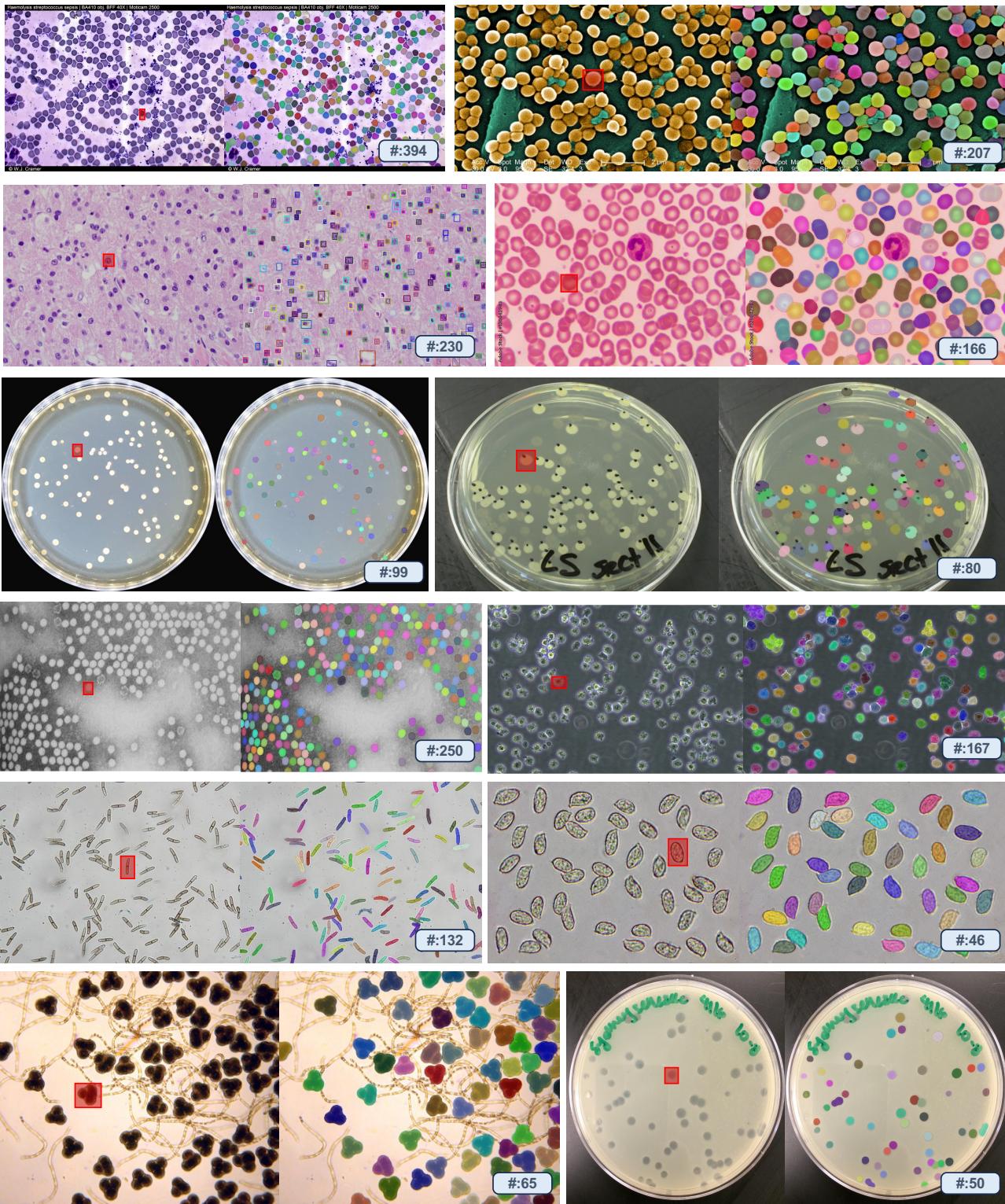
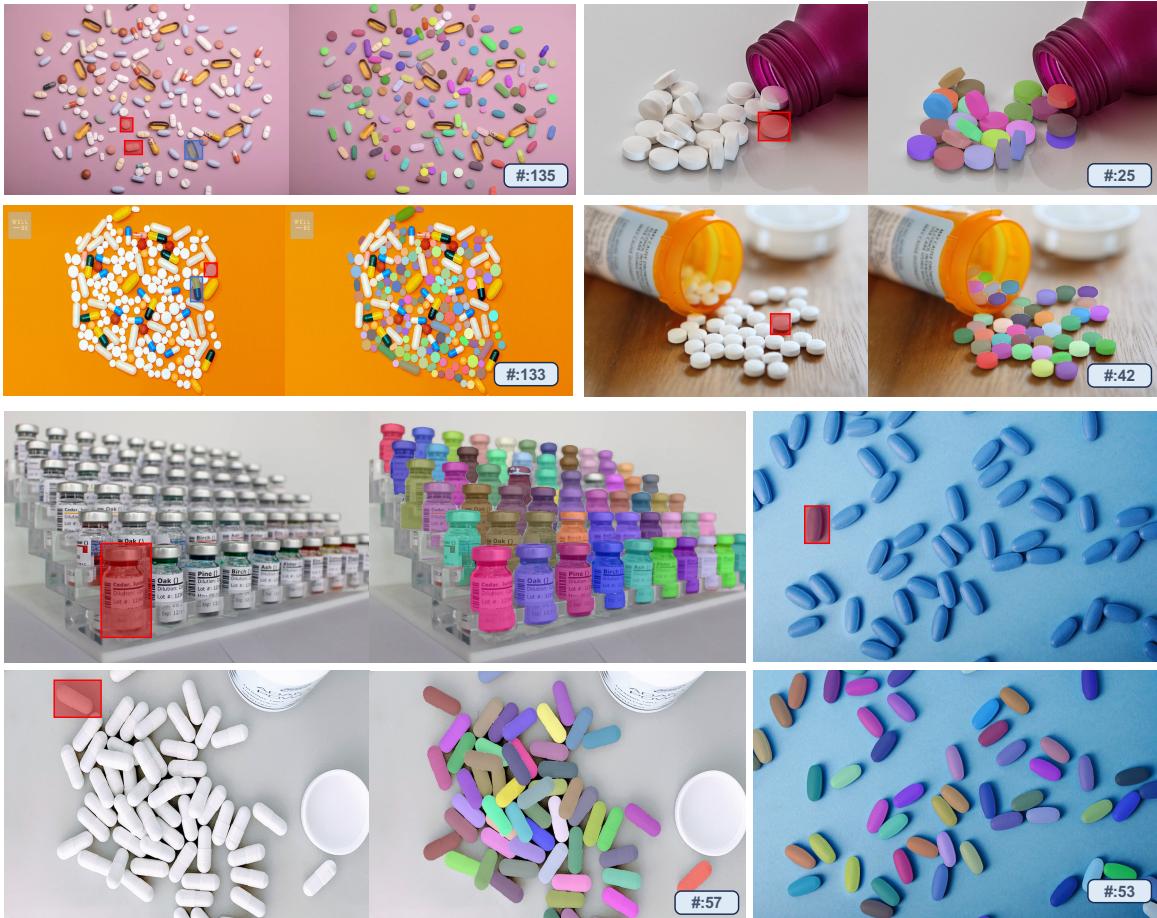


Figure 16. T-Rex applied to biology. To more intuitively visualize overlapping and dense objects, we prompt SAM with the boxes predicted by T-Rex to obtain segmentation masks. The red boxes on the original image are visual prompts for positive examples and the blue boxes are visual prompts for negative examples.

# Medicine



OCR

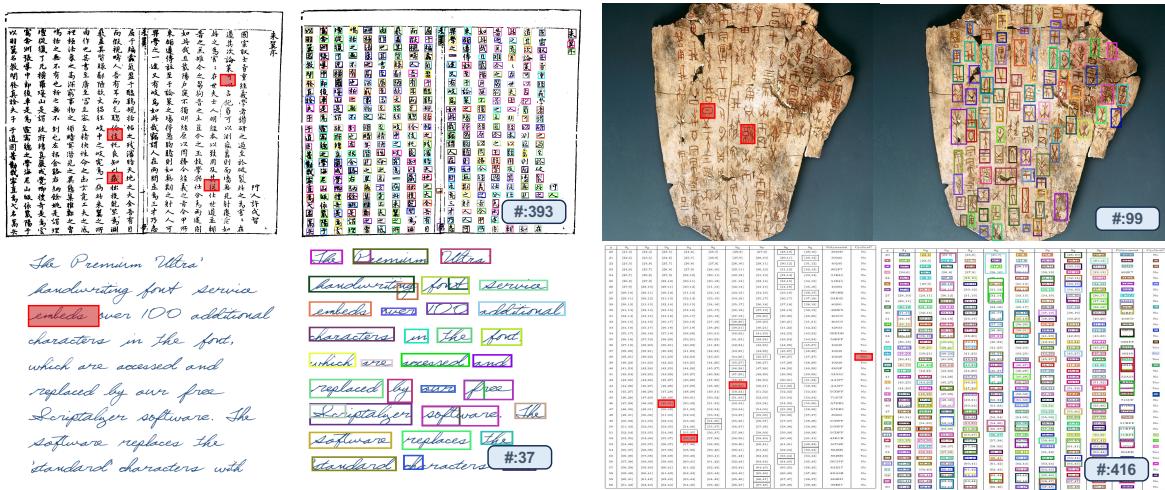


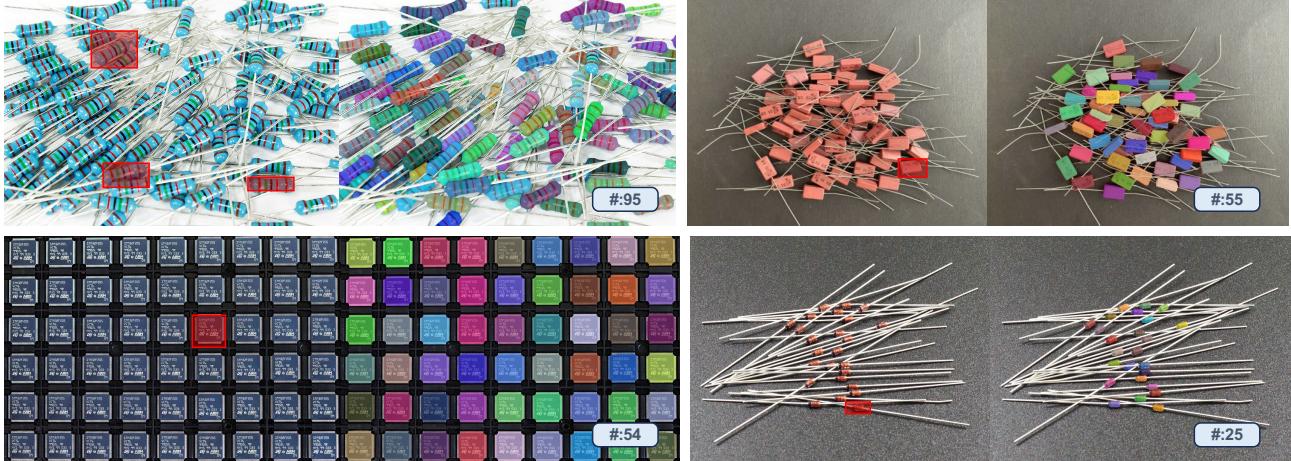
Figure 17. T-Rex applied to medicine and OCR. To more intuitively visualize overlapping and dense objects, we prompt SAM with the boxes predicted by T-Rex to obtain segmentation masks. The red boxes on the original image are visual prompts for positive examples and the blue boxes are visual prompts for negative examples.

## Retail



Figure 18. T-Rex applied to retail. To more intuitively visualize overlapping and dense objects, we prompt SAM with the boxes predicted by T-Rex to obtain segmentation masks. The red boxes on the original image are visual prompts for positive examples and the blue boxes are visual prompts for negative examples.

## Electronic



## Transportation

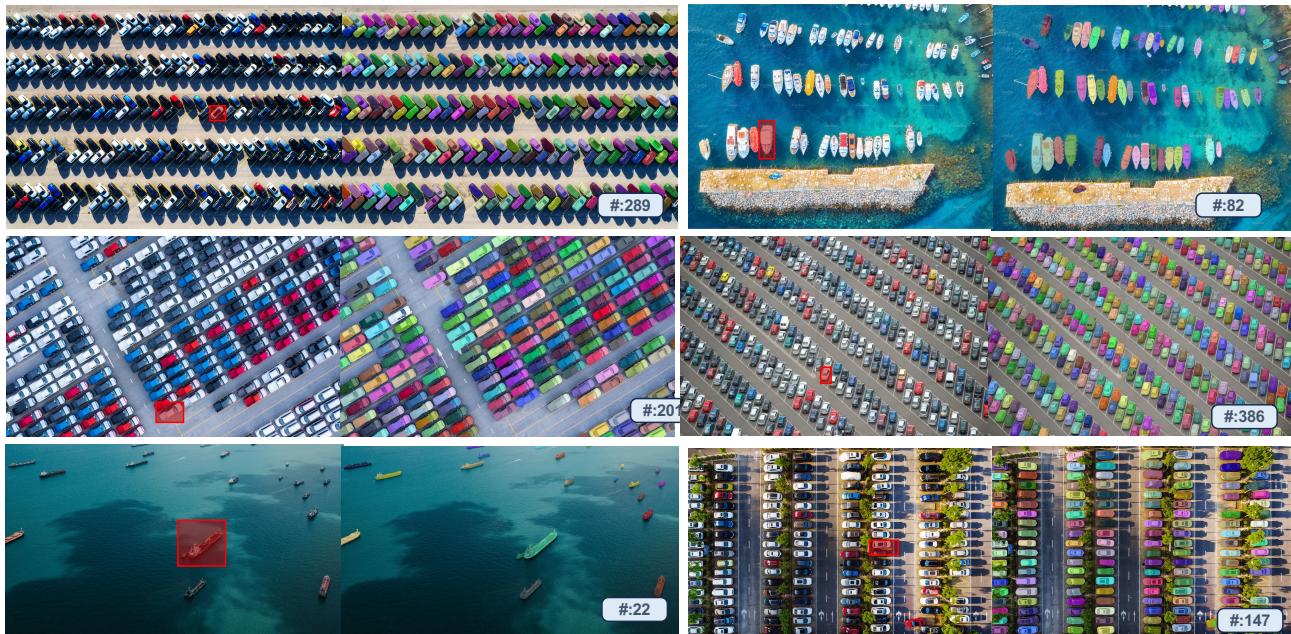


Figure 19. T-Rex applied to electronic and transportation. To more intuitively visualize overlapping and dense objects, we prompt SAM with the boxes predicted by T-Rex to obtain segmentation masks. The red boxes on the original image are visual prompts for positive examples and the blue boxes are visual prompts for negative examples.

# Logistics



# Human



Figure 20. T-Rex applied to logistics and human. To more intuitively visualize overlapping and dense objects, we prompt SAM with the boxes predicted by T-Rex to obtain segmentation masks. The red boxes on the original image are visual prompts for positive examples and the blue boxes are visual prompts for negative examples.

## Cartoon

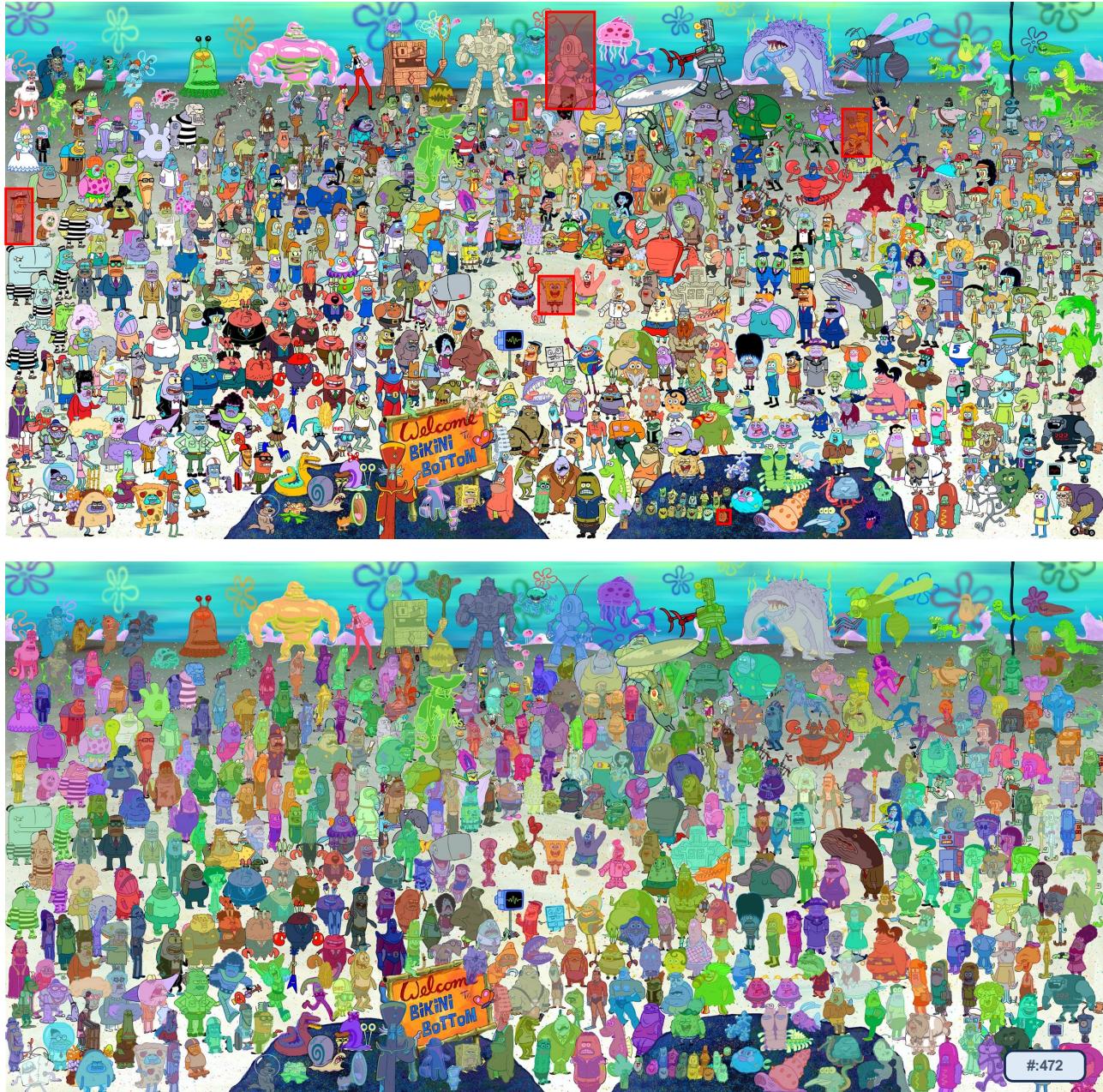


Figure 21. T-Rex applied to cartoon. To more intuitively visualize overlapping and dense objects, we prompt SAM with the boxes predicted by T-Rex to obtain segmentation masks. The red boxes on the original image are visual prompts for positive examples and the blue boxes are visual prompts for negative examples.

## Known Failure Cases



Figure 22. Failure Cases in T-Rex: (a) In single-target scenes, if only a single prompt is introduced at the background, T-Rex will still detect dense objects in the figure. (b) In dense multi-object scenes, T-Rex can produce false detections, requiring multiple prompts or negative prompts for correction. (c) During cross-image workflows, T-Rex risks over-fitting in single-target dense scenes, leading to erroneous detections like mistaking silkworm eggs for tomatoes.