

Imagen Editor and EditBench: Advancing and Evaluating Text-Guided Image Inpainting

Su Wang* Chitwan Saharia* Ceslee Montgomery*
Jordi Pont-Tuset Shai Noy Stefano Pellegrini Yasumasa Onoe
Sarah Laszlo David J. Fleet Radu Soricut Jason Baldridge
Mohammad Norouzi† Peter Anderson† William Chan†

Google Research

A. Appendix

A.1. Human Evaluation Results

In this section we include further analysis and additional details of the human evaluation results reported in the main paper. In Fig. 1 we provide a breakdown of single-image human evaluations for the EditBench Mask-Rich prompts, illustrating the average proportion of objects and attributes correctly rendered for each model. Focusing on a conservative bar for performance, correctly rendering *at least one* of the three objects-attribute pairs in a Mask Rich prompt, Imagen Editor achieves over 85% across comparisons. This further supports the conclusions of the main paper, object-masking (IM vs. IM_{RM}) improves object rendering, attribute rendering, and attribute binding (i.e., object and attribute both correct, far right column).

Impact of Mask Size. In Fig. 2 we report single-image human evaluations by mask size (Small, Medium and Large). In general, performance trends are consistent with the aggregate results, however object-masking (IM_{RM} vs. IM) is more beneficial for small and medium masks than with large masks. For reference, Fig. 3 provides examples of the different mask sizes in each bucket.

Single image vs. Side-by-Side. A key consideration in evaluating text alignment was whether to compare model outputs *side-by-side*, in keeping with prior works [1, 3, 4, 7], or whether to evaluate each model separately – judging a *single image* at a time. We report both but focus primarily on single-image evaluations for two reasons:

- *Fine-grained Evaluation.* While it is reasonable to ask the annotators simple comparative questions such as *which image matches the caption better?*, the task becomes more cognitive taxing and prone to error when multiple attributes and objects are compared [2].
- *Avoiding Combinatorial Explosion.* The single image format facilitates pairwise comparison without eliciting

judgments for a (often impractically) large number of model pairs when more models are investigated. In addition, we avoid exposing annotators to the same outputs multiple times, which might introduce exposure bias.

While the single image evaluation format may be subject to calibration biases, i.e., not all annotators will have the same threshold for judging correctness, we control for this by ensuring that all model evaluations are performed in a batch presented in random order to a large pool of annotators.

Significance of single-model human evaluation. We include 95% confidence error bars calculated with bootstrap resampling in all single-model human evaluations (Figs. 7, 9 and 10). If error bars do not overlap, the difference in scores is significant. In particular, the difference between Imagen Editor and the other models is significant in all cases. We confirmed this, as suggested, using the one-sided, two-sample proportions z test. Illustratively, when comparing Imagen Editor’s overall image-text alignment performance (Fig. 7) vs the next best model the p-values were as follows for each prompt type: Full, $P = 3.5 \times 10^{-10}$, Mask-Simple, $P = 2.0 \times 10^{-8}$, Rich, $P = 5.2 \times 10^{-7}$. In Tab. 1 and Tab. 2 (correlation between automatic metrics and human evaluations) the 95% confidence intervals calculated with bootstrap resampling are <1%.

Sampling Strategy and Number of Evaluations. In *single image* evaluations we evaluated 4 edited image samples for each prompt from each of the four models. In total this gave: prompts (240) \times prompt types (3) \times image samples (4) \times models (4) = 11,520 outputs rated by annotators. In the *side-by-side* evaluation of Mask-Rich prompts, we evaluated 3 model pairs (Imagen Editor vs. Stable Diffusion, DALL-E 2 and Imagen Editor_{RM}), resulting in: 3 \times 240 (images) \times 1 (prompt types) \times 3 (votes from different annotators) = 2,160 ratings. In side-by-side evaluations, an image was selected at random from the 4 samples per model.

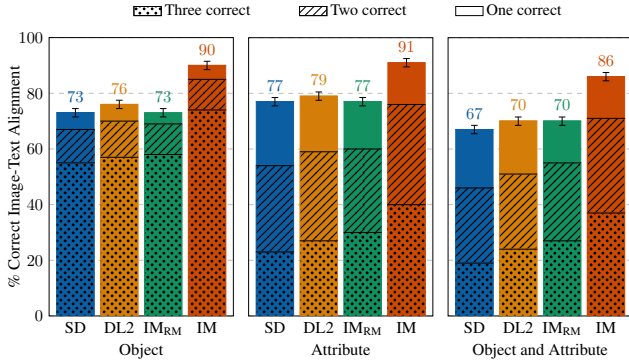


Figure 1. Single-image human evaluations on EditBench Mask-Rich illustrating the *number of objects and attributes* correct. Comparing IM vs. IM_{RM}, object-masking improves the rendering of objects and attributes as well as attribute binding.

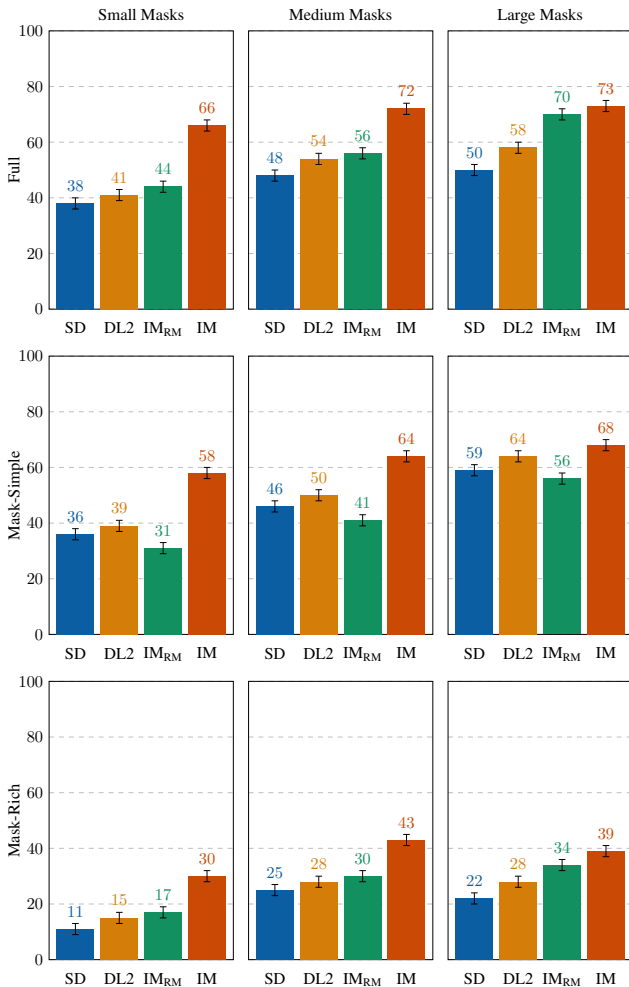


Figure 2. Single-image human evaluations on EditBench by *mask size* (columns) and *prompt type* (rows). Imagen Editor is preferred in all comparisons and object-masking during training is particularly beneficial for small masks (IM vs. IM_{RM}).

Annotators. We use a total of 18 US-based annotators and the evaluation load was spread approximately equally. Each annotator spent roughly ~ 30 s per prompt.

Crowdsourcing UI. We illustrate the actual interface used in our human evaluations in Figs. 5, 6.

A.2. Imagen Editor object masking

We apply a bounding box based masking strategy for Imagen Editor, which is an adaptation of random mask policy used in previous work. During training the mask is the union of a random mask and an object detection bounding box as described in Fig. 4.



Figure 3. Examples of different mask sizes from the Small, Medium and Large buckets reported in Fig 2. Mask sizes were determined by binning mask-to-image area ratios into 3 quantiles as follows: Small (5.7–21.5%), Medium (21.5–36.9%), and Large (>36.9%).

```
def get_mask(image):
    # Binary mask using random masking policy.
    # 0 indicates the masked region.
    random_mask = get_random_mask(image)

    # Call the object detector to get the top
    # bounding box prediction.
    bbox = get_bbox_from_image(image)

    # Create a mask from the bbox. Specifically, set the
    # region enclosed by bbox to 0s.
    bbox_mask = get_mask_from_bbox(bbox)

    mask = bbox_mask * random_mask

    return mask
```

Figure 4. Bounding box based mask generation for Imagen Editor. We adapt the random mask policy used in [5, 6].

A.3. Examples and Failure Cases

In Fig. 7 we provide further examples comparing outputs from Imagen Editor when trained with object-masking vs. random masking. We find that object masking makes the model noticeably more robust when handling richer prompts with more details of objects and their attributes. To illustrate the variety of samples evaluated from each model, in Figs. 8–11 we illustrate sampled outputs from Stable Diffusion, DALL-E 2, Imagen Editor_{RM} and Imagen Editor respectively.

Imagen Editor Failure Cases. In Fig. 12, we further explore Imagen Editor failure cases. We focus on attribute types as Fig. 1 shows that, even in the case of more complex, Mask Rich prompts, models are relatively strong at getting the majority of objects mentioned correct. As is consistent with our breakdown of Mask-Simple prompts by Attribute type, a qualitative review of Imagen Editor failure cases on Mask-Simple prompts supports that Imagen Editor is fairly strong on *color* and *material*. Where there were failure cases the objects or the colors tended to be uncommon (i.e. “butter-colored” letters or “silver” llama in the figure). *Size* and *shape* are admittedly often more challenging because they can be more ambiguous. Yet still there are a handful of cases where the size attribute appears to


be ignored (i.e. “tiny octopus” in figure). In almost all cases, some object and often it’s more common attributes are inpainted (an example of this is the “pentagon-shaped block” instead of a cube-shaped block). Finally, *count* is notoriously challenging and the most clear failure case of the various models. Rarely do they render too many objects. Almost always they render far too few, often over 50% of objects are missing.

References

- [1] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. *Proceedings of CVPR*, abs/2111.14818, 2022. 1
- [2] Sara Dolnicar, Bettina Grün, and Friedrich Leisch. Quick, simple and reliable: forced binary survey questions. *International Journal of Market Research*, 2011. 1
- [3] Gwanghyun Kim and Jong Chul Ye. Diffusionclip: Text-guided image manipulation using diffusion models. *arXiv preprint arXiv:2110.02711*, 2021. 1
- [4] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Bob McGrew Pamela Mishkin, Ilya Sutskever, and Mark Chen. GLIDE: Towards Photorealistic Image Generation and Editing with Text-Guided Diffusion Models. In *arXiv:2112.10741*, 2021. 1
- [5] Chitwan Saharia, William Chan, Huiwen Chang, Chris A. Lee, Jonathan Ho, Tim Salimans, David J. Fleet, and Mohammad Norouzi. Palette: Image-to-Image Diffusion Models. In *arXiv:2111.05826*, 2021. 3
- [6] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Free-form image inpainting with gated convolution. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4471–4480, 2019. 3
- [7] Yufan Zhou, Ruiyi Zhang, Jiuxiang Gu, Chris Tensmeyer, Tong Yu, Changyou Chen, Jinhui Xu, and Tong Sun. Interactive image generation with natural-language feedback. In *Proceedings of AAI*, 2022. 1

Task: Review the Prompt and pay special attention to the red box region in the Image. Then, answer the question below.

Prompt: a flat-shaped cat hanging on the cabinets in a kitchen.



Does the image match the caption?

Yes


No

(a) Full prompt single image evaluation with binary selection for overall text-image alignment.

Task: Review the Prompt and pay special attention to the red box region in the Image. Then, look at each "Object" / "Attribute" / "Object + Attribute" row. For each row, check the box next to the "Object" if it appears in the red box in the Image and check the box next to the "Attribute" if it appears in the red box in the Image. Finally, check the box next to the "Object + Attribute" if the "Object" and "Attribute" pair appear together in the red box in the Image.

Note: some objects or attributes may be repeated, but please still check the box each time if they appear.

Prompt: a short letter "C".



Object	Attribute	Object + Attribute
<input type="checkbox"/> letter "C"	<input type="checkbox"/> short	<input type="checkbox"/> letter "C" IS short
<input type="checkbox"/> No objects or attributes matched the image		

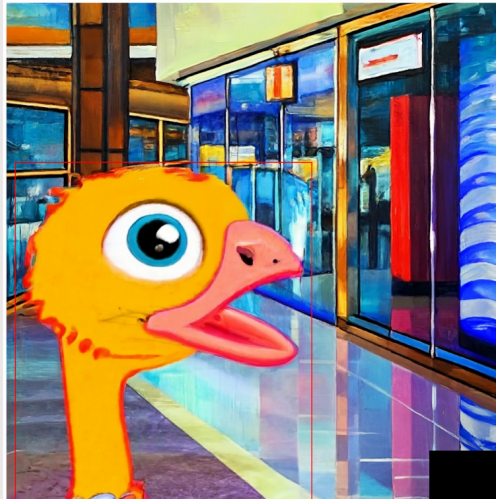
(b) Mask-Simple prompt evaluation where annotators assess existence of the correct object and attribute separately and together to measure correct attribute binding.

Figure 5. Crowdsourcing UI illustration: Full prompts & Mask-Simple prompts.

Task: Review the Prompt and pay special attention to the red box region in the Image. Then, look at each "Object" / "Attribute" / "Object + Attribute" row. For each row, check the box next to the "Object" if it appears in the red box in the Image and check the box next to the "Attribute" if it appears in the red box in the Image. Finally, check the box next to the "Object + Attribute" if the "Object" and "Attribute" pair appear together in the red box in the Image.

Note: some objects or attributes may be repeated, but please still check the box each time if they appear.

Prompt: an orange ostrich with brownish body and an orange tail.



Object

- ostrich
- body
- tail

Attribute

- orange colored
- brownish
- orange

Object + Attribute

- ostrich IS orange colored
- body IS brownish
- tail IS orange

No objects or attributes matched the image

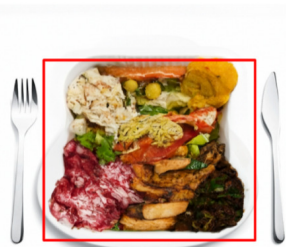
Submit

(a) *Mask-Rich* prompt evaluation, which is similar to *Mask-Simple* (Fig. 5b) but involves 3 pairs of attributes and objects.

Task : Evaluate the given images and rate them in order

More instructions on how to complete the task are available in this [guidelines doc](#)

Caption: a platter of food placed in a triangle shape in a white plate.



1. Which image is more realistic?

- Model 1
- Model 2

2. Which image matches with the caption better?

- Model 1
- Model 2

Submit

(b) Side-by-Side evaluation. The second (text-image alignment) question only appears after the first (realism) question is answered.

Figure 6. Crowdsourcing UI illustration: *Mask-Rich* prompts & *Side-by-side* evaluations.

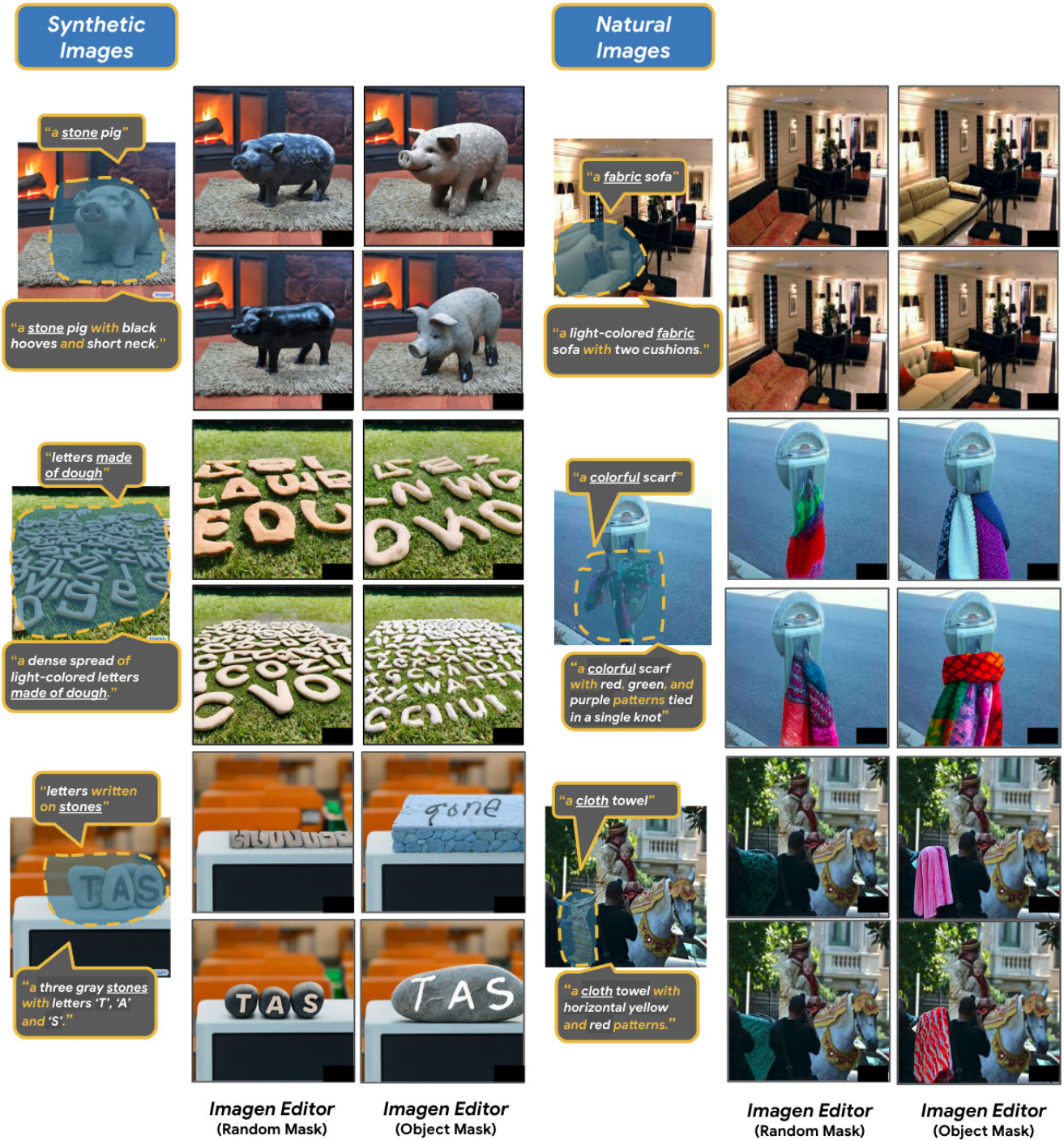


Figure 7. Additional examples comparing the random and object-masking strategies on *Mask-Simple* and *Mask-Rich* prompts. Imagen Editor is substantially more robust at handling richer attribute/object specifications, as confirmed by human evaluations.

"a stone pig with black hooves and short neck."



"a dense spread of light-colored letters made of dough."



"a three gray stones with letters 'T', 'A' and 'S'."



"a light-colored fabric sofa with two cushions."



"a colorful scarf with red, green, and purple patterns tied in a single knot"



"a cloth towel with horizontal yellow and red patterns."



Figure 8. Stable Diffusion examples.

"a stone pig with black hooves and short neck."



"a dense spread of light-colored letters made of dough."



"a three gray stones with letters 'T', 'A' and 'S'."



"a light-colored fabric sofa with two cushions."



"a colorful scarf with red, green, and purple patterns tied in a single knot"



"a cloth towel with horizontal yellow and red patterns."



Figure 9. DALL-E 2 examples.

"a stone pig with black hooves and short neck."



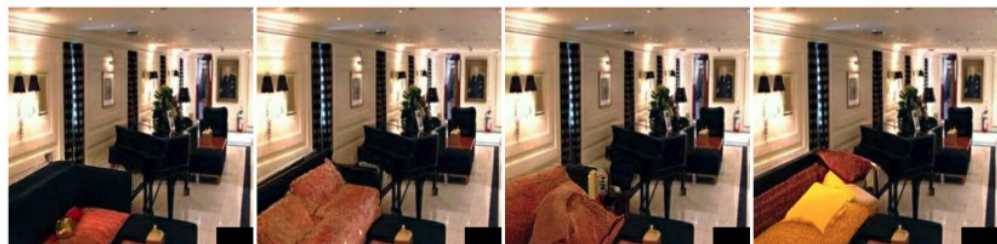
"a dense spread of light-colored letters made of dough."



"a three gray stones with letters 'T', 'A' and 'S'."



"a light-colored fabric sofa with two cushions."



"a colorful scarf with red, green, and purple patterns tied in a single knot"



"a cloth towel with horizontal yellow and red patterns."



Figure 10. Imagen Editor (random masking) examples.

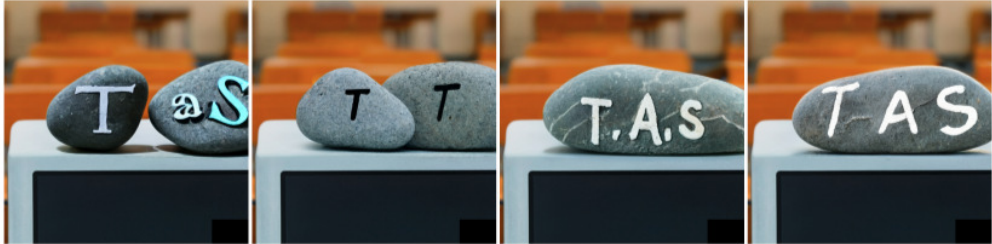
"a stone pig with black hooves and short neck."



"a dense spread of light-colored letters made of dough."



"a three gray stones with letters 'T', 'A' and 'S'."



"a light-colored fabric sofa with two cushions."



"a colorful scarf with red, green, and purple patterns tied in a single knot"



"a cloth towel with horizontal yellow and red patterns."



Figure 11. Imagen Editor (object masking) examples.

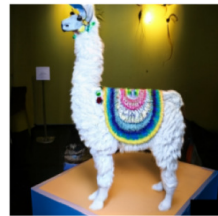
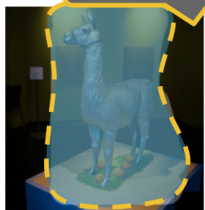
material

"letters in glass blocks"



color

"a silver llama"



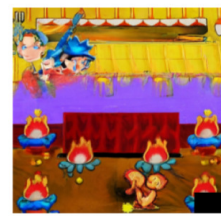
size

"a tiny octopus"



count

"six monkeys"



shape

"The letter 'M' on a pentagon-shaped block"



Figure 12. Imagen Editor failure cases by attribute. **Material** - the blocks don't quite have the transparency property you'd expect of letters encased in a glass box. **Color** - the llama is not quite silver colored in any case. The object "llama" is object type = "uncommon" denoting that a "silver" llama is likely out of distribution for the model and therefore, more challenging. **Size** - the octopus is never quite "tiny" this is wrong in at least two ways: not absolutely (with respect to the image size), nor relatively (compared to the car). **Count** - count is among the more challenging attributes and anecdotally, models are often off by 50% or more. **Shape** - all image samples revert to the standard shape for a block: a cube.