Half&Half: New Tasks and Benchmarks for Studying Visual Common Sense

Ashish Singh[1*] Hang Su[1*] SouYoung Jin[1] Huaizu Jiang[1] Chetan Manjesh[1]
Geng Luo[1] Ziwei He[1] Li Hong[1] Erik G. Learned-Miller[1] Rosemary Cowell[2]
College of Information and Computer Sciences[1] Psychological and Brain Sciences[2]
University of Massachusetts Amherst
{ashishsingh, hsu, souyoungjin, hzjiang, cmanjesh, gluo, ziweihe, lhong, elm}@cs.umass.edu
rcowell@psych.umass.edu

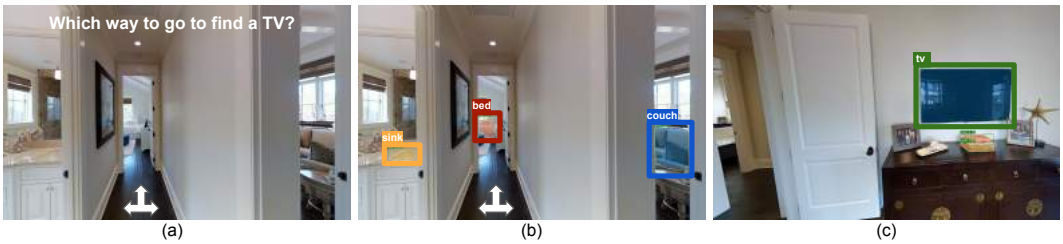

Figure 1: **Intelligent Search: (a)** Suppose we are in the hallway of an unfamiliar apartment, with three directions to go, and want to find a TV. Which direction should we go? **(b)** An intelligent agent, leveraging visual clues and reasoning that a TV is more likely to be near a couch, might decide to turn right. **(c)** In this case, such a choice leads quickly to a TV.

1. Introduction

The general recognition of objects, people, actions and scene types has been a core focus of computer vision research. However, now that we have achieved a degree of success in these problems, it is time to define new problems that will spur us to reach the next level of visual intelligence. The development of *visual common sense* is critical to the development of intelligent agents that can be useful in dynamic, novel environments.

But what exactly is visual common sense? We suggest that the ability to make intelligent assessments of where things might be, when not directly visible, is a critical and ubiquitous capability shared by humans and other intelligent beings, and is a fundamental component of visual common sense. Humans regularly demonstrate the ability to make decisions in the absence of explicit visual cue (Fig. 1). This sort of "intelligent search" is a prominent example of visual common sense, and we believe it represents a skill that will be essential in developing intelligent agents.

Closely related to our work are earlier efforts on incorporating contextual information for visual prediction [5, 10, 11, 9]. We believe a formal benchmark on such capabilities in the most basic forms can be a valuable addition.

1.1. The Half&Half visual prediction tasks

In this work, we formalize the problem of inferring the presence of what we cannot already see in an image. To do this, we rely on the fact that different views of an image depict the same scene. Hence, individual sections can be used as contextual cues for the other section. For this reason, we call these tasks the **Half&Half tasks**. We define three different Half&Half tasks (Fig. 2):

- **Image-to-Label task:** One half of the image is provided, and the task is to infer a categorical label for what is likely to be present in the other half amongst the given K choices(Fig. 3).
- **Label-to-Image task:** A target category and a set of K half images are provided. The task is to infer which candidate is most likely to have the target in its other half (Fig. 4).
- **Image-to-Image task:** A query image and K image choices are provided, all of them being half images. The task is to infer which of the choices is the most likely to be from the same image as the query (Fig. 5).

Figure 2: The Half&Half visual prediction tasks.

The three variants of the tasks were inspired by the common-sense reasoning capabilities of intelligent beings under uncertainty. Specifically, an agent trying to find a specific type of object should be able to decide whether the current direction is promising (Image-to-Label). And if not, given observations towards other directions, which one should be preferred (Label-to-Image)? Image-to-Image is modeling an intelligent reasoning capability to directly predict the next visual observations, which can enable an agent to prepare for imminent encounters.

Our hope is that the Half&Half benchmarks, and perhaps their next generations, drive forward the research in

designing intelligent agents by training and evaluating such systems for visual "common sense". We aim to make this benchmark public soon and plan to keep a public leaderboard for the benchmarks.

2. The Half&Half Benchmarks

In this section, we describe our three new benchmarks. Each benchmark is constructed to study one of the three variants of the Half&Half tasks introduced in Sec. 1.1. We make use of images and annotations from existing datasets originally created for object detection or scene understanding. As we will show in this section, the way we create the benchmarks requires no additional annotations compared to those standard recognition tasks. This allows us to directly make use of large-scale existing datasets.

2.1. The Image-to-Label benchmark

Figure 3: Half&Half Image-to-Label benchmark example

Image selection The benchmark is created using the training and validation images from MS-COCO [6]. We consider the left half of each image as the context for the objects present in the right half . From the above sets, we sample images that have at least one single object present in its right half. Furthermore, we discard images whose left half and right half contain any overlapping objects. As a design choice, we exclude the "person" category and consider only the remaining 79 categories since we observe that "person" is very common in MS-COCO and has significant co-occurrence with the majority of other categories. In total, we obtain $45,843$ images meeting the criteria above.

Problems and splits Out of all the obtained images, we create a random train/val/test split of $32,000/3,843/10,000$ images. Each of the training and validation images are provided with one image (the left half) and a set of labels from the right half. From the test images, we form test problems in the form of the Image-to-Label task. Fig. 3 shows an example. Five candidate categories are given where only one of them actually appears in the right half. For the correct candidate choice, we randomly pick one object category that exists in the right half among the ground truth. For the wrong candidates, we randomly select from all MS-COCO object categories not present in the whole image.

Evaluation During testing, we evaluate the performance of a model on the test problems based on whether it can pick the right candidate among the five choices, and also the rank that it assigns to the correct candidate. Specifically, benchmark users are required to report:

(1) Rank-1 Accuracy: $\frac{1}{N}\sum_i [\![r_i = 1]\!]$,
(2) Mean Reciprocal Rank (MRR): $\frac{1}{N}\sum_i \frac{1}{r_i}$.

Here N denotes the total number of test samples and r_i is the rank of the correct candidate in a model's output.

2.2. The Label-to-Image benchmark

Image selection Because of the close formulation between the Label-to-Image and Image-to-Label tasks, we can reuse the data collected for the Image-to-Label benchmark, with a few critical modifications. The same set of images, labels, and train/val/test image split are used. The differences only lie in the way the problems are formulated.

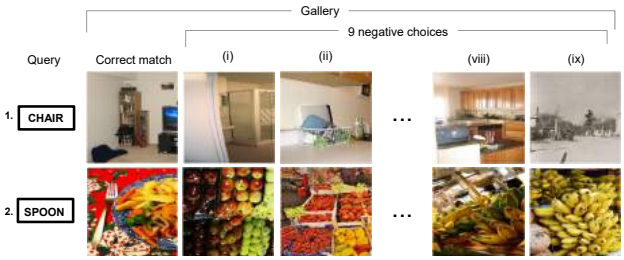

Figure 4: Half&Half Label-to-Image benchmark examples

Problems and splits As illustrated in Fig. 4, a Label-to-Image problem contains a query label and 10 gallery images. We create one such problem for each of the images in the benchmark using the following steps:

(1) Each (right-half object label, left-half image) pair from Image-to-Label benchmark is sampled as query object and correct candidate;
(2) From the remaining images in the split, images containing the query object label in their right halves are filtered out. The remaining left-half images are then ranked based on the similarity scores with the correct candidate and 9 images are selected randomly as wrong candidates from the top 100.

We follow [3] and use low-level visual features (GIST and color histogram) for computing the similarity. In total, we obtain $47,370/5,686/10,000$ train/val/test problem sets.

Evaluation During testing, the objective is to correctly pick the correct choice among the $K = 10$ gallery choices. The evaluated algorithm is required to provide a ranking among the choices for each test problem, and the two evaluation measures, rank-1 accuracy and MRR, are reported.

2.3. The Image-to-Image benchmark

Image selection Since the Image-to-Image task involves half images as both queries and candidate choices and requires no label annotations, we are able to consider any natural images for building the benchmark. We chose to use the SUN360 dataset [12], which offers a large collection of high-resolution 9104×4552 rectangular panorama images. Using panoramas allows us to cut image crops instead of using adjacent halves. By making the two "halves" some distance apart, they will have little overlap in content and offer diverse visual information.

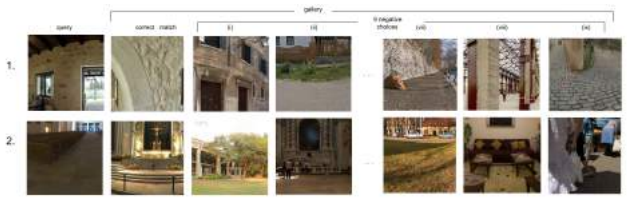

Figure 5: Half&Half Image-to-Image benchmark example

Problems and splits Among all images available in SUN360, we randomly sample $27,999$ training images and $29,142$ testing images. Each partition is further divided into a query partition (from which query and correct choices are drawn) and a gallery partition (from which negative choices are drawn). We construct one problem for each image in the query partition: one of its two crops is randomly chosen as the query image and the other one as the correct choice. Nine wrong choices are then randomly sampled from the corresponding gallery partition. A ranking procedure that is the same as the Label-to-Image benchmark is also applied to avoid trivial solutions. In total, we obtain $8,399$ training and $8,742$ testing problems, which are constructed from the training and testing partitions, respectively.

Evaluation During testing, the objective is to correctly pick the correct one among the $K = 10$ choices. The evaluated algorithm is required to pick a top choice for each test problem. Rank-1 accuracy is reported for evaluation.

3. Methods and Experiments

3.1. Image-to-Label

Our task for this benchmark is to identify object categories that are likely to be present in the right half by observing only the left half. We formulate this as a multi-category classification problem.

We define two types of classifiers: **symmetric** and **anti-symmetric**. A symmetric classifier is a standard CNN trained on the left-half image to predict the labels of objects in the *left half* itself, which is equivalent to a traditional classifier. Meanwhile, an anti-symmetric classifier is trained on the same set of left-half images, but with object categories present in the right half as the target labels.

For training the classifiers, we use the training split provided by the benchmark. Given the set of all left-half images, we train a CNN (ResNet-50 [4] pretrained on ImageNet) to predict the presence of object categories. The last FC layer is modified to match the 79 object categories we use according to the classifier (symmetric or anti-symmetric). If there are multiple categories for an image, we duplicate the left-half image in the training set and assign an individual category to each of the left-half images. We follow this approach so as to maintain consistent behavior with the benchmark-setting, where our candidate list only contains a single correct object category.

From the trained network, we obtain the posterior probability distribution over all 79 categories in the MS-COCO dataset. We evaluate the performance of our context driven model on the benchmark by computing the ranking of the five candidate categories in the candidate list according to their posterior probabilities.

Tab. 1 compares symmetric and anti-symmetric classifiers, as well as a MLP baseline using GIST [8].

Table 1: Evaluations on the Image-to-Label benchmark. For reference, chance performance is 20% acc. and 0.457 MRR.

Classifier	Rank-1 Acc.	MRR
MLP (GIST)	42.0%	0.635
Symmetric	58.7%	0.707
Anti-Symmetric	74.3%	0.855

3.2. Label-to-Image

For the Label-to-Image benchmark, our goal is to rank the candidate images based on the likelihood of containing the given query object. We propose following two methods.

Indirect training In this case, we use any classifier trained on the Image-to-Label task to compute posterior probabilities for the query object. Based on these posteriors, candidates are ranked. We directly use the anti-symmetric classifier trained for the Image-to-Label benchmark.

Direct training In the second method, we train a CNN to *directly* compare the given candidate images conditioned on the query label. We formalize this as a classification problem where, given the candidate set, the objective is to predict the most likely image to contain the query label. For each image in the candidate set, we compute a *class score* for the query label. To do this, we consider the output of the classification layer of a CNN. We then normalize the class scores across candidate images. This reflects the probability distribution among the candidate images given the query label. Finally, ten candidate images are ranked according to their respective posterior probabilities for the query label. We use the same ResNet-50 model (ImageNet pretrained) as our base classifier.

The direct and indirect classifiers are compared in Tab. 2. Direct training provides a noticeable gain, which suggests it is beneficial to have a training objective more closely aligned with the actual evaluation protocol.

Table 2: Evaluations on the Label-to-Image benchmark.

Classifier	Rank-1 Acc.	MRR
Indirect	44.7%	0.624
Direct	46.6%	0.646

3.3. Image-to-Image

For this task, given a query image and the 10 gallery images, the goal is to find the correct match coming from the same image. As two **baselines**, we compute the L2 distance of feature vectors from the last fully connected layer of a ResNet-18 [4] between the query image and each of the gallery images. We use models pre-trained on ImageNet [2] and Places365 [13].

In addition to the baselines, we also train networks with two different metric learning techniques. Suppose we have feature vectors, $\mathbf{x} \in \mathcal{R}^{D\times 1}$, $\mathbf{y} \in \mathcal{R}^{D\times 1}$, computed from the backbone network. With **bilinear metric learning**, a similarity score between $\mathbf{x}$ and $\mathbf{y}$ is computed by $\mathbf{x}^\intercal \mathbf{W} \mathbf{y}$ where $\mathbf{W} \in \mathcal{R}^{D\text{x}D}$ can be trained. We also train networks with **symmetric metric learning** that learns $\mathbf{L} \in \mathcal{R}^{D\text{x}D}$ in $(\mathbf{L}\mathbf{x})^\intercal (\mathbf{L}\mathbf{y})$.

All of our networks are trained using triplet loss [7], where a triplet is generated with a query image, the corresponding correct match, and one of the 9 negatives. Each model is then trained such that a query image is closer to the correct match than to the negative. We first train networks by freezing the parameters in the backbone network. After $\mathbf{W}$ and $\mathbf{L}$ are learned, we also fine-tune the entire network.

Table 3: Evaluations on the Image-to-Image benchmark.

Pre-trained	f.t.?	L2	Symm. Metric	Bilinear Metric
ImageNet		47.3%	54.0%	54.1%
ImageNet	✓	65.3%	64.8%	70.0%
Places365		56.2%	63.1%	65.1%
Places365	✓	67.2%	67.7%	69.0%

Results and comparisons are summarized in Tab. 3. A few observations can be made: (1) Place365 offers better pre-training compared to ImageNet for our problems; (2) various metric learning technique all help significantly compared to direct L2 distances; and (3) fine-tuning the backbone network offers consisting improvements.

4. Preliminary Results for Visual Navigation

(a) (b) (c)

Figure 6: Example of a navigation task built from the Active Vision Dataset. Query: "toilet"; Correct answer: (c).

In this section, we demonstrate how models trained on our benchmarks can be useful for visual navigation applications. For this preliminary evaluation, we build a small test set using the Active Vision dataset [1], originally designed for indoor navigation. We formulate the problem statement as: "Which is the best scene to find the target object?". We follow the Label-to-Image task formulation to create an approximation of visual navigation task by sampling a target object label and three candidate images (one correct, two wrong) to create a problem (see an example in Fig. 6).

Model trained on our Label-to-Image benchmark (the direct training variant) achieves 68% accuracy and human performance is at 98.3%. We conduct a human study with 6 participants. While our model does show significant advantage over chance performance (33%), there is still a large gap towards human performance, highlighting the importance of future research in this direction.

References

[1] P. Ammirato, P. Poirson, E. Park, J. Kosecka, and A. C. Berg. A dataset for developing and benchmarking active vision. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2017. 4

[2] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proc. CVPR*, pages 248–255. Ieee, 2009. 4

[3] J. Hays and A. A. Efros. Scene completion using millions of photographs. *ACM Transactions on Graphics (TOG)*, 26(3):4, 2007. 2

[4] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *CoRR*, 2015. 3, 4

[5] G. Heitz and D. Koller. Learning spatial context: Using stuff to find things. In *Proc. ECCV*, pages 30–43. Springer, 2008. 1

[6] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: Common objects in context. In *Proc. ECCV*, pages 740–755. Springer, 2014. 2

[7] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proc. CVPR*, June 2015. 4

[8] C. Siagian and L. Itti. Rapid biologically-inspired scene classification using features shared with visual attention. *pami*, 29(2):300–312, Jan. 2007. 3

[9] S. Song, A. Zeng, A. X. Chang, M. Savva, S. Savarese, and T. Funkhouser. Im2pano3d: Extrapolating 360 structure and semantics beyond the field of view. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3847–3856, 2018. 1

[10] J. Sun and D. W. Jacobs. Seeing what is not there: Learning context to determine where objects are missing. In *Proc. CVPR*, pages 1234–1242. IEEE, 2017. 1

[11] A. Torralba. Contextual priming for object detection. *IJCV*, 53(2):169–191, 2003. 1

[12] J. Xiao, K. A. Ehinger, A. Oliva, and A. Torralba. Recognizing scene viewpoint using panoramic place representation. In *Proc. CVPR*, pages 2695–2702. IEEE, 2012. 2

[13] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba. Places: A 10 million image database for scene recognition. *PAMI*, 2017. 4