Predicting Goal-directed Human Attention Using Inverse Reinforcement Learning

Zhibo Yang[1], Lihan Huang[1], Yupei Chen[1], Zijun Wei[2], Seoyoung Ahn[1],
Gregory Zelinsky[1], Dimitris Samaras[1], Minh Hoai[1]
[1]Stony Brook University, [2]Adobe Inc.

Abstract

Human gaze behavior prediction is important for behavioral vision and for computer vision applications. Most models mainly focus on predicting free-viewing behavior using saliency maps, but do not generalize to goal-directed behavior, such as when a person searches for a visual target object. We propose the first inverse reinforcement learning (IRL) model to learn the internal reward function and policy used by humans during visual search. We modeled the viewer's internal belief states as dynamic contextual belief maps of object locations. These maps were learned and then used to predict behavioral scanpaths for multiple target categories. To train and evaluate our IRL model we created COCO-Search18, which is now the largest dataset of high-quality search fixations in existence. COCO-Search18 has 10 participants searching for each of 18 target-object categories in 6202 images, making about 300,000 goal-directed fixations. When trained and evaluated on COCO-Search18, the IRL model outperformed baseline models in predicting search fixation scanpaths, both in terms of similarity to human search behavior and search efficiency. Finally, reward maps recovered by the IRL model reveal distinctive target-dependent patterns of object prioritization, which we interpret as a learned object context.

1. Introduction

Human visual attention comes in two forms. One is bottom-up, where prioritization is based solely on processing of the visual input. The other is top-down, where prioritization is based on many top-down information sources (object context of a scene, semantic relationships between objects, etc. [14, 38, 54]). When your food arrives at a restaurant, among your very first attention movements will likely be to the fork and knife (Fig. 1), because they are important to your goal of having dinner.

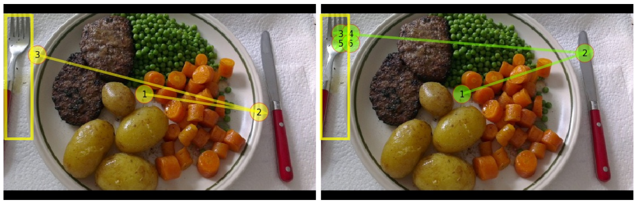

Figure 1: **Predicting fixations in a visual search task**. Left: behavioral scanpath shown in yellow. Right: predicted scanpath in green. The search target is the fork, shown in the yellow bounding box.

Goal-directed attention control underlies all the tasks that we **try** to do, thus making its prediction a more challenging and important problem than predicting the bottom-up control of attention by a visual input. One of the strongest forms of top-down attention control is in the definition of a target goal. Arguably the simplest goal-directed task is visual search—there is a target object and the task is to find it. Humans are very efficient and flexible in the image locations that they choose to fixate while searching for a target-object goal, making the prediction of human search behavior important for both behavioral and computer vision, e.g. robotic visual systems [16, 39]. In this paper, we introduce Inverse Reinforcement Learning as a computational model of human attention in visual search.

Gaze prediction in visual search. We aim to predict the fixation patterns made during the visual search of an image. These patterns can be either spatial (fixation density maps) or spatial+temporal (scanpaths). Most fixation-prediction models are of free-viewing behavior. A critical difference between search and free-viewing tasks is that search fixations are guided towards a target-object goal, whereas in free viewing there are no explicit goals. Prioritization of fixation locations during free viewing is thought to be controlled by bottom-up saliency, and since Itti's [25] seminal work the prediction of free-viewing fixations using saliency maps has grown into a large literature [6, 8, 9, 12, 23, 26, 27, 32–34, 36]. However, saliency model predictions do not generalize to fixations made in goal-directed attention tasks, such as target object search [20, 31].

Code and dataset are available at https://github.com/cvlab-stonybrook/Scanpath_Prediction.

Dataset	Search	Image	Class	Subj/img	Fixation
SALICON [27]	✗	10000	-	60	4600K*
POET [42]	✗	6270	10	5	178K
People900 [15]	✓	912	1	14	55K
MCS [56]	✓	2183	2	1-4	16K
PET [18]	✓	4135	6	4	30K
COCO-Search18	✓	6202	18	10	300K

Table 1: **Comparison of fixation datasets**. Previous datasets either did not use a search task, or had very few target-object classes, subjects, or fixations. *: Fixations are approximated by mouse clicks.

Early models of target guidance during search used simple targets having features that were known to the searcher [53]. This work expanded to include computational models using images of objects and scenes as inputs [15, 57, 58], and the inclusion of target spatial relationships [4] and global scene context [49] to help guide attention to targets and improve search efficiency. There have only been a few attempts to use deep network models to predict human search fixations [2, 52, 59]. Critically, all of these models use algorithms and knowledge about a particular source of information (target features, meaning, context, etc), to prioritize image locations for fixation selection.

Inverse Reinforcement Learning. Our approach to search-fixation prediction is the opposite. Instead of an algorithm to prioritize locations in an image, we use Inverse Reinforcement Learning (IRL) [1, 17, 21, 41, 60] to learn sequences of search fixations by treating each as a potential source of reward. IRL, a form of imitation learning, aims to recover an expert's underlying reward function through repeated observation. Most IRL algorithms [17, 55, 60] simultaneously learn an optimal policy and the reward function on which the policy is optimized. Although early IRL algorithms [41, 60] were often restricted to problems with low-dimensional state spaces, deep maximum entropy IRL [55] can handle raw image inputs. Recent work [17, 21] applies adversarial training [19] to learn the underlying reward function and the policy, treating each as (part of) the discriminator and the generator in adversarial training, respectively. The discriminator assigns high reward to an expert's behavior and low reward to a non-expert's behavior, where behavior is represented as state-action pairs. The generator/policy is optimized using a reinforcement learning algorithm to get higher reward by behaving more like the expert. Here, we use the GAIL (generative adversarial imitation learning) algorithm [21], because it can imitate behaviors in complex, high-dimensional environments [21]. We define a unified information-maximization framework to combine diverse information sources, in order to select maximally-rewarding locations to fixate, thus increasing accuracy and applicability of human search fixation prediction.

Search Fixation Datasets. Our other significant contribution is the COCO-Search18 dataset, currently the world's largest dataset of images annotated with human gaze fixations collected during search. COCO-Search18 is needed because the best models of goal-directed attention will likely be trained on goal-directed behavior data. For free-viewing fixations, the currently best model is DeepGaze II [34], trained on SALICON [27]. SALICON is a crowd-sourced dataset of images annotated with mouse clicks indicating attentionally salient image locations.

There is nothing comparable to SALICON to train goal-directed attention models. Moreover, the existing suitably large datasets, each suffer from some weakness that limits their usefulness (Tab. 1), the most common being that the fixation behavior was not collected during a visual search task (as in [37, 42]). Datasets using a search task either had people search for multiple targets simultaneously [18] or used only one target category (people [15]) or two (microwaves and clocks [56]). These inadequacies demand a new, larger, and higher-quality dataset of search fixations for model training. We use multiple fixation-based behavioral search metrics to interrogate COCO-Search18, which we predict using IRL and other state-of-the-art methods.

Contributions. **(1)** We apply Inverse Reinforcement Learning (GAIL) to the problem of predicting fixation scanpaths during visual search, the first time this has been done for a goal-directed attention. **(2)** In order to apply IRL to scanpath prediction we needed to integrate changes in fixation location with changes in the state representation, a problem that we solved using Dynamic Contextual Beliefs. DCB is a novel state encoder that updates beliefs about peripherally-viewed objects (an object context) based on the movements of a simulated fovea. **(3)** We introduce COCO-Search18, a large-scale, high-quality dataset of COCO images annotated with the fixations of 10 people searching for 18 target-object categories. COCO-Search18 makes possible the deep network modeling of goal-directed attention. **(4)** We show through model comparison and with multiple metrics that our IRL model outperforms other state-of-the-art methods in predicting search scanpaths. We also show that the IRL model (i) learns an object's scene context; (ii) generalizes to predict the behavior of new subjects, and (iii) needs less data to achieve good performance compared to other models. **(5)** Finally, we learn how to quantify a reward function for the fixations in a search task. This will make possible a new wave of experimental investigation that will ultimately result in better understanding of goal-directed attention.

2. Scanpath Prediction Framework

We propose an IRL framework (Fig. 2) to model human visual search behavior. A person performing a visual search task can be considered a goal-directed agent, with their fix-

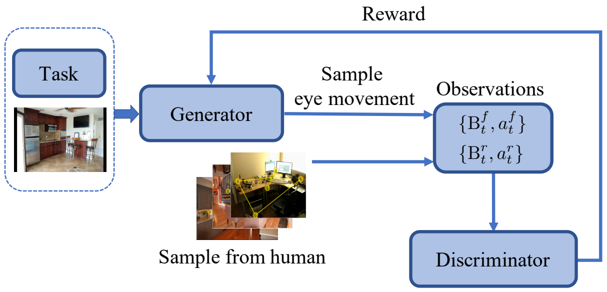

Figure 2: **Overview of the IRL framework**. The generator (policy) generates fake state-action pairs $\{B_t^f, a_t^f\}$ by sampling eye movements from given images and tasks. The discriminator (reward function) is trained to differentiates real human state-action pairs $\{B_t^r, a_t^r\}$ from the generated ones and provides reward to train the generator. The states B_t^f and B_t^r use DCB representations.

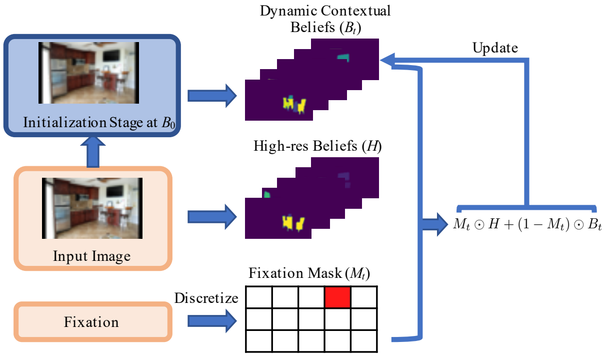

Figure 3: **Overview of DCB**. First, an input image and its low-res image counterpart are converted into the high-res beliefs and low-res beliefs. The initial state B_0 is set as the low-res belief. At each fixation which is discretized into a binary fixation mask M_t with 1's at the fixation location and 0's elsewhere, a new state is generated by applying Eq. (1).

ations being a sequential decision process of the agent. At each time step, the agent attends to (fixates) a specific location within the image and receives a version of the image that is blurred to approximate the human viewer's visual state, what we call a retina-transformed image. This is an image that has high-resolution (non-blurred) information surrounding the attended location, and lower-resolution information outside of this central simulated fovea [45]. The state of the agent is determined by the sequence of visual information that accumulates over fixations toward the search target (Sec. 2.1), with each action of the agent depending on the state at that time during the evolving state representation. The goal of the agent is to maximize internal rewards through changes in gaze fixation. While it is difficult to behaviorally measure how much reward is received from these fixations, with IRL this reward can be assumed to be a function of the state and the action, and this function can be jointly learned using the imitation policy (Sec. 2.2).

2.1. State Representation Modeling

To model the state of a viewer we propose a novel state representation for accumulating information through fixations that we term a **Dynamic-Contextual-Belief (DCB)**. As shown in Fig. 3, DCB is composed of three components: 1) Fovea, which receives a high-resolution visual input only from the image region around the fixation location; 2) Contextual beliefs, which represent a person's gross "what" and "where" understanding of a scene in terms of level of class confidence; and 3) Dynamics, which actively collects information with each fixation made during search. We discuss each component in greater detail below.

Fovea: The primate visual system has a fovea, which means that high-resolution visual information is available only at a central fixated location. To accumulate information from the visual world, it is therefore necessary to selectively fixate new image locations. Visual inputs outside of the fovea have lower resolution, with the degree of blur depending on the distance between the peripherally-viewed input and the fovea. Rather than implementing a full progressive blurring of an image input (i.e., a complete retina-transformed image, as in [56]), for computational efficiency here we use a local patch from the original image as the high-resolution foveal input and a blurred version of the entire image to approximate low-resolution input from peripheral vision.

Contextual Belief: Attention is known to be guided to target (and target-like) objects during search, but more recently it has been suggested that attention is also guided to "anchor objects" [7, 50], defined as those objects having a learned spatial relationship to a target that can help in the efficient localization of that target. For example, people often look at the wall when searching for a TV because TVs are often found hanging on the wall. Inspired by this, we propose to model, not only the target features (as in [59]), but also other objects and background information in the state.

We hypothesize that people have an internal scene parser that takes an image input and generates belief maps for that image based on all the objects and background classes in that person's knowledge structure. We believe these belief maps also guide movements of the fovea to capture high-resolution information and form better beliefs. We approximate these belief maps using a Panoptic-FPN [29] for panoptic segmentation [30]. Given an image, Panoptic-FPN generates a pixel-level mask for each "thing" class (object) and each "stuff" class (background) in the image. There are 80 "thing" categories (including a single "other" class for the 80 "thing" classes) and 54 "stuff" categories [10, 30, 35]. We create a mask for each category by grouping all mask instances belonging to the same category and use the belief maps of the 134 categories as the primary component of the state representation. We term these belief maps *contextual beliefs* because the collective non-target beliefs constitute a context of spatial cues that might affect the selection of fixations during the search for a target.

Dynamics refers to the change in the state representation that occurs following each fixation. We propose a simple yet effective heuristic to model state dynamics (see Fig. 3). Initially, the state is based on the contextual beliefs on the low-resolution image corresponding to a peripheral visual input. For each fixation by the searcher, we update the state by replacing the portion of the low-resolution belief maps with the corresponding high-resolution portion obtained at the new fixation location. The state is updated as follows:

$$B_0 = L \text{ and } B_{t+1} = M_t \odot H + (1 - M_t) \odot B_t, \quad (1)$$

where B_t is the belief state after t fixations, M_t is the circular mask generated from the t^{th} fixation, L and H are the belief maps of "thing" and "stuff" locations for low-resolution and high-resolution images, respectively. Humans have different search behaviors on the same image given different search targets. To capture this, we augment the state by concatenating it with a one-hot task vector. Please refer to the supplementary material for more detail.

2.2. Reward and Policy Learning

We learn the reward function and the policy for visual search behavior using Generative Adversarial Imitation Learning (GAIL) [21]. As shown in Fig. 2, GAIL is an adversarial framework with a discriminator and a generator. The policy is the generator that aims to generate state-action pairs that are similar to human behavior. The reward function (the logarithm of the discriminator output) maps a state-action pair to a numeric value. We train the generator and discriminator with an adversarial optimization framework to obtain the policy and reward functions.

Let D and G denote the discriminator and the generator, respectively. The discriminator aims to differentiate human state-action pairs from fake state-action pairs generated by the policy. This corresponds to maximizing the following objective function:

$$\begin{aligned}\mathcal{L}_D =& \mathbb{E}_r[\log(D(S,a))] + \mathbb{E}_f[\log(1 - D(S,a))] \\ &- \lambda \mathbb{E}_r[\|\nabla D(S,a))\|^2]. \end{aligned} \quad (2)$$

In the above objective function, $\mathbb{E}_r$ denotes the expectation over the distribution of real state-action pairs, while $\mathbb{E}_f$ denotes the expectation over the fake samples generated by the generator (i.e., the policy). The last term of the above objective is the expected squared norm of the gradients, which is added for faster convergence [46]. The reward function is defined based on the discriminator:

$$r(S,a) = \log(D(S,a)). \quad (3)$$

The generator aims to fool the discriminator, and its objective is to maximize the log likelihood of the generated state-action pairs, i.e., to maximize: $\mathcal{L}_G = \mathbb{E}_f[\log(D(S,a))] = \mathbb{E}_f[r(S,a)]$.

The generator is an RL policy, hence its objective can be equivalently reformulated as an RL objective and optimized by Proximal Policy Optimization [48]:

$$\mathcal{L}_\pi = \mathbb{E}_\pi[\log(\pi(a|S))A(S,a)] + H(\pi). \quad (4)$$

We use GAE [47] to estimate the advantage function A which measures the gain of taking action a over the policy's default behavior. $H(\pi) = -\mathbb{E}_\pi[\log(\pi(a|S))]$, the entropy in max-entropy IRL [60].

3. COCO-Search18 Dataset

COCO-Search18 is a large-scale and high-quality dataset of search fixations obtained by having 10 people viewing 6202 images in the search for each of 18 target-object categories. Half of these images depicted an instance of the designated target object and the other half did not, meaning that we adopted a standard target-present (TP) or target-absent (TA) search task. All images in COCO-Search18 were selected from the COCO trainval set [35]. Five criteria were imposed when selecting the TP images: **(1)** No images depicting a person or an animal (to avoid known strong biases to these categories that might skew our measures of attention control [11, 28]). **(2)** The image should include one and only one instance of the target. **(3)** The size of the target, measured by the area of its bounding box, must be >1% and <10% of the area of the search image. **(4)** The target should not be at the center of the image, enforced by excluding an image if the target bounding box overlapped with the center cell of a 5x5 grid. **(5)** The original image ratio (width/height) must be between 1.2 and 2.0 to accommodate the display screen ratio of 1.6. After applying these exclusion criteria, and excluding object categories that had less than 100 images of exemplars, we were left with 32 object categories (out of COCO's 80) to use as search targets. To exclude images in which the target was highly occluded or otherwise difficult to recognize, we trained a patch-based classifier for target recognition (described in supplemental material) and only selected images in which the cropped target-object patch had a classification confidence in the top 1%. Finally, we manually excluded images depicting digital clocks from the clock target category (because the features of analog and digital clocks are very different and this would be expected to reduce data quality by creating variability in the search behavior), as well as images depicting objectionable content. This left us with 3101 TP images over 18 target categories. To select the same number of TA images for each of these 18 categories, we randomly sampled COCO trainval images with the following constraints: **(1)** The image should not depict an instance of the target, and **(2)** The image must include at least two instances of

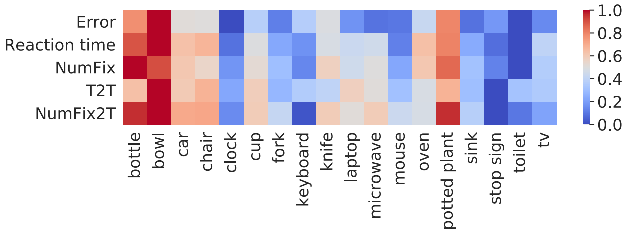

Figure 4: Normalized gaze data [0,1] on response error, reaction time, number of fixations (NumFix), time to target (T2T), and number of fixations to target (NumFix2T) averaged over 10 subjects searching for 18 categories in TP images. Redder color indicates harder search targets, bluer color indicates easier search.

the target's siblings, as defined in COCO. For example, a microwave sibling can be an oven, a toaster, a refrigerator, or a sink, which are under the parent category of appliance. We did this to discourage TA responses from being based on scene type (e.g., a city street scene would be unlikely to contain a microwave).

Each of the 10 student participants (6 males, age range 18-30, normal or corrected-to-normal vision) viewed all 6202 images, and their eye position throughout was sampled at 1000Hz using an EyeLink 1000 eyetracker (SR Research) in tower-mount configuration under controlled laboratory conditions. For each subject, data collection was distributed over six sessions in six days, with each session having equal number TP trials and TA trials (∼500 each) randomly interleaved. Each session required ∼2 hours. For each image, subjects made a TP or TA judgment by pressing a 'yes' or 'no' button on a game pad. They searched all the images for one target category before preceding to next category. A total of 299,037 fixations were extracted from the eye position data, over the 10 subjects, although only data from the TP fixations will be reported here (Fig. 4). TP fixations occurring on error trials, or after fixation on the target, were discarded. This left 100,232 TP search fixations to use for training and testing. All model evaluations are based on 70% training, 10% validation, and 20% test, random splits of COCO-Search18, within each target category.

4. Experiments

We evaluate the proposed framework and its constituent components in multiple experiments. We first compare the scanpath predictions by the IRL-based algorithm to predictions from various heuristic methods and behavior cloning methods using ConvNets and convolutional LSTM. We then study the algorithm's ability to generalize to new human subjects. Finally, we analyze context effects, the value of having more training data, and report on an ablation study. We used only the target-present trials from COCO-Search18, leaving target-absent data for future analyses.

4.1. Comparing Scanpath Prediction Models

Comparison methods. We compare the IRL algorithm for predicting scanpaths to several baselines, heuristics, and behavior cloning methods: (1) **Random scanpath**: we predict the scanpath for an input image by randomly selecting a human scanpath for the same search target but in a different input image. (2) **Detector**: we train a simple ConvNet to predict the location of the target and sample a sequence of fixation locations based on the detection confidence map over the image. Regions with higher confidence scores are more likely to be sampled. (3) **Fixation heuristics**: rather than sampling from a detector's confidence map, here we generate fixations by sampling from a fixation density map produced by a ConvNet (with a similar network architecture as the Detector) trained on human fixation density maps. (4) **BC-CNN** is a behavior cloning method, where we train a ConvNet to predict the next fixation location from the DCB state representation. Note that this state representation and network structure are identical to the one used by the IRL policy described in Sec. 2.1. (5) **BC-LSTM** is a behavior cloning method similar to BC-CNN, but the state representation and update are done with a convolutional LSTM. Instead of having the simple predefined update rule used by both IRL and BC-CNN, as shown in Eq. (1), BC-LSTM aims to learn a recurrent update rule automatically with an LSTM: $B_{t+1} = ConvLSTM(B_t, I_t)$, where $ConvLSTM$ denotes a convolutional LSTM cell [5], B_t is the hidden state of the LSTM cell and also the searcher's belief state after t fixations. I_t is the input to the LSTM at time t, and it is defined as $I_t = M_t \odot H + (1 - M_t) \odot L$. Recall that M_t is the circular mask generated from the t^{th} fixation, L and H are the predicted maps from the Panoptic-FPN [30] for the 80 COCO objects and 54 "stuff" classes for low- and high-resolution input images, respectively.

Results. Fig. 5 shows the cumulative probability of gaze landing on the target after each of the first 6 fixations made by humans and the algorithms in our model comparison. First, note that even the most predictive models have a performance ceiling lower than that of humans, whose ceiling over this range is nearly 1. These lower ceilings likely reflect a proportion of trials in which the models search was largely unguided. Second, note the steep increase in target fixation probability after the first and second fixations. The slopes of these functions indicate strong target guidance. The target was fixated in the very first movement on about half of the images, with the IRL model replicating human search guidance slightly better than its nearest competitors: the Detector and BC-CNN models.

We quantify the patterns from Fig. 5 using several metrics. Two of these metrics follow directly from Fig. 5 and capture aggregate measures combining search guidance and accuracy. The first of these computes the area under the cu-

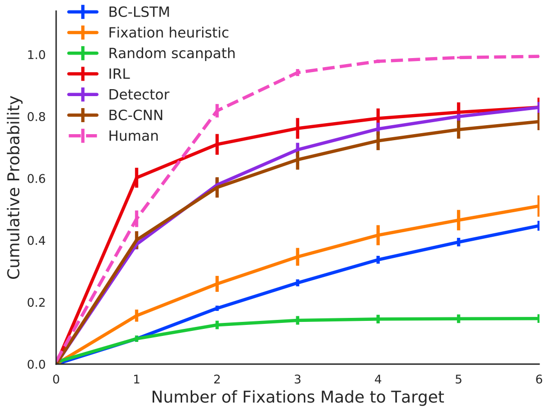

Figure 5: **Cumulative probability of fixating the target** for human searchers and all predictive methods. X-axis is the number of fixations until the fovea moves to the target object; Y-axis is the percentage of scanpaths that succeed in locating the target. Means and standard errors are first computed over target categories, and then over searchers.

mulative probability of the target fixation curve, a metric we refer to as **Target Fixation Probability AUC** or TFP-AUC. Second, we compute the sum of the absolute differences between the human and model cumulative probability of target fixation in a metric that we refer to as **Probability Mismatch**. We also report the **Scanpath Ratio**, which is a widely used metric for search efficiency. It is computed by the ratio of Euclidean distance between the initial fixation location and the center of the target to the summed Euclidean distances between fixations to the target [22]. Finally, we compute two metrics for scanpath prediction success, that both capture the scanpath similarity between fixation sequences generated by humans and sequences generated by the model. The first of these computes a **Sequence Score** by first converting a scanpath into a string of fixation cluster IDs and then use a string matching algorithm [40] to measure similarity between two strings. Finally, we use **MultiMatch** [3, 13] to measure the scanpath similarity at the pixel level. MultiMatch measures five aspects of scanpath similarity: shape, direction, length, position, and duration. We exclude the duration metric because the studied models do not predict fixation duration. Unless otherwise specified, each model generates 10 scanpaths of maximum length 6 (excluding the first fixation) for each testing image by sampling from the predicted action map at each fixation, with the results averaged over scanpaths.

As seen from Tab. 2, the IRL algorithm outperforms the other methods on all metrics. The performance of IRL is closest to *Human*, an oracle method where the scanpath of a subject is used to predict the scanpath of another subject for the same input image. Fig. 6 also shows that **reward maps** recovered by the IRL model depend greatly on the category of the search target. In the top row, higher reward was assigned to the laptop when searching for a mouse, while for the same image greater reward was expected from fixating on the monitor when searching for a tv. Similarly, the search for a car target in the bottom image resulted in the expectation of reward from the other cars on the road but almost not at all from the highly-salient stop sign, which becomes intensely prioritized when the stop sign is the target.

Implementation details. We resize each input image to 320×512 and obtain a low-resolution image by applying a Gaussian filter with standard deviation $\sigma = 2$. To compute the contextual beliefs, we use a Panoptic-FPN with backbone network ResNet-50-FPN pretrained on COCO2017 [30]. Panoptic-FPN outputs a feature map of 134 channels, corresponding to 80 object categories and 54 background classes in COCO, and it is resized to 20×32 spatially.

For IRL and BC-CNN, we use the same policy network architecture: a network composed of four convolutional (conv) layers and a softmax layer. IRL model has two additional components—critic network and discriminator network. The **critic network** has two convolutional layers and two fully-connected (fc) layers. The discriminator network shares the same sturcture with the IRL policy network except the last layer which is a sigmoid layer. Each conv layer and fc layer in BC-CNN and IRL is followed by a ReLU layer and a batch-norm layer [24]. BC-LSTM has the same policy network as the BC-CNN, with the difference being the use of a convolutional LSTM [5] to update the states. BC-CNN and BC-LSTM use the KL divergence between predicted spatial distribution and ground truth as loss. The prediction of both behavior cloning models and IRL is conditioned on the search target. We implement the target conditioning by introducing an additional bias term based on the search task to the input features at each layer [44]. The human visual system employs Inhibition-of-Return (IOR) to spatially tag previously attended locations with inhibition to discourage attention from returning to a region where information has already been depleted [51]. To capture this mechanism, we enforce IOR on the policy by setting the predicted probability map to 0 at each attended location using a 3×3 grid. See the supplementary for more detail.

4.2. Group Model vs Individual Model

The previous subsection described the IRL model's ability to predict a searcher's scanpath on unseen test images, but how well can this model predict the scanpaths of a new unseen searcher without training on that person's scanpaths? To answer this question, we perform ten leave-one-subject-out experiments, with each experiment corresponding to a test subject. For every subject we train two models: (1) a group model using the scanpaths of the 9 other subjects; and (2) an individual model using the scanpaths of the test subject on the training images. We evaluate the performance of these models on the scanpaths of each test subject on the unseen test images. Fig. 7 shows that both

	TFP-AUC ↑	Probability Mismatch ↓	Scanpath Ratio ↑	Sequence Score ↑	MultiMatch ↑			
					shape	direction	length	position
Human	5.200	-	0.862	0.490	0.903	0.736	0.880	0.910
Random scanpath	0.795	4.407	-	0.295	0.869	0.558	0.849	0.849
Detector	4.046	1.166	0.687	0.414	0.877	0.676	0.853	0.863
Fixation heuristic	2.154	3.046	0.545	0.342	0.873	0.614	**0.870**	0.850
BC-CNN	3.893	1.328	0.706	0.409	0.880	0.669	0.865	0.874
BC-LSTM	1.702	3.497	0.406	0.324	0.834	0.567	0.818	0.770
IRL(Ours)	**4.509**	**0.987**	**0.826**	**0.422**	**0.886**	**0.695**	0.866	**0.885**

Table 2: **Comparing scanpath prediction algorithms** (rows) using multiple scanpath metrics (columns) on the COCO-Search18 test dataset. In the case of Sequence Score and Multimatch, "Human" refers to an oracle method where one searcher's scanpath is used to predict another searcher's scanpath; "Human" for all other metrics refers to observed behavior.

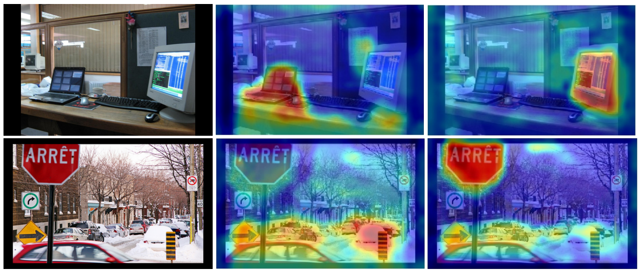

Figure 6: Initial **reward maps** learned by the IRL model for two different search targets in two test images. Top row: original image (left), mouse target (middle), and tv target (right). Bottom row: original image (left), car target (middle), stop sign target (right). Redder color indicates the expectation of higher reward for fixating a location.

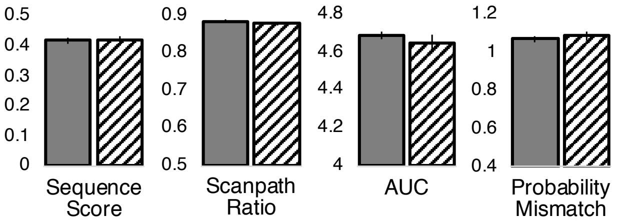

Figure 7: No significant differences were found between a group model (solid), trained with 9 subjects, and an individual model (striped), trained with one subject.

models perform well, with an insignificant performance gap between them. This suggests that there is good agreement between group and individual behaviors, and that a group model can generalize well to new searchers.

4.3. Context Effects

Search efficiency. With DCB we can ask how an object from category A affects the search for a target from category B. This effect can either increase (guidance) or decrease (distraction) search efficiency. To study this, we first zero out the belief map of category A in the DCB state representation and then measure the TFP-AUC (see Sec. 4.1) on test images for category B. We compute the difference between the TFP-AUC obtained with and without switching off the belief map for category A in DCB. A positive value indicates that an object in category A helps to guide search for a target in category B, while a negative value indicates the opposite (that the object is distracting). We did this for the 134 COCO objects and stuff non-target categories A and the 18 target categories B. Fig. 8 shows the six most guiding and distracting objects for the knife and car searches. Note that the fork was highly distracting when searching for a knife, likely because the two look similar in periphery vision, but that the cake facilitated the knife search. Similarly for the car search, pavement provided the strongest guidance whereas trucks were the most distracting.

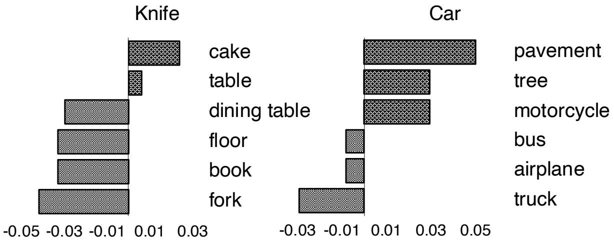

Figure 8: **Context effect**. The six most influential context objects (grey bars) for knife and car search tasks. The y-axis is the context object category and the x-axis is a measure of how much the belief map for a context object contributed to search efficiency, as measured by TFP-AUC. Larger positive values mean that the context object improved search guidance to the target, more negative values mean that the object distracted attention from the search.

Directional Prior. Can an object from category A serve as a directional spatial cue in the search for a target from category B? Suppose M is the probability map produced by the policy network of our IRL model, and let M' be the modified probability map from the policy network but with the belief map of category A in the DCB state representation being switched off. By computing the difference between M' and M which we call a *context map* (as depicted in the top row of Fig. 9), we can see the spatial relationship between the context object A and the target object B (see

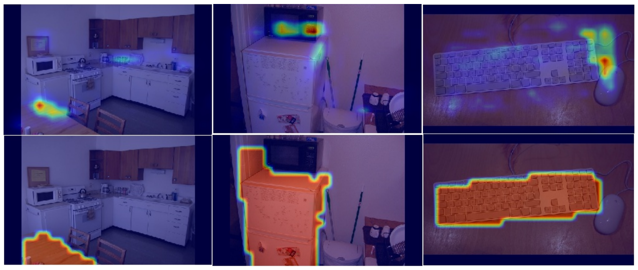

Figure 9: **Spatial relations** between context and target objects learned by the model. Top row shows individual context maps for a dining table (left) and a refrigerator (middle) in a microwave search, and a keyboard (right) in a mouse search. Bottom row are the belief maps of the corresponding context objects. Gaze is guided to the top of the dinning table and refrigerator when searching for a microwave, and to the right of the keyboard when searching for a mouse.

	Sequence Score ↑	Scanpath Ratio ↑	Prob. ↓ Mismatch
DCB-full	0.422	0.803	1.029
w/o history map	0.419	0.800	1.042
w/o saliency map	0.419	0.795	1.029
w/o stuff maps	0.407	0.777	1.248
w/o thing maps	0.331	0.487	3.152
w/o target map	0.338	0.519	2.926
DCB	0.422	0.826	0.987
CFI	0.402	0.619	1.797

Table 3: **Ablation study of the proposed state representation—dynamic contextual belief**. The full state (DCB-full) consists of 1 history map, 1 saliency map, 54 stuff maps, 79 context maps and 1 target map. We mask out one part by setting the map(s) to zeros at each time. See the supplementary for full results.

Fig. 9 for examples).

4.4. Ablation Study on State Representation

DCB is a rich representation that uses top-down, bottom-up, and history information. Specifically, it consists of 136 belief maps, divided into five factor groups: target object (1 map), context objects (79 maps), "stuff" (54 maps), saliency (1 map, extracted using DeepGaze2 [34]), and history (1 binary map for the locations of previous fixations). To understand the contribution of a factor group, we remove the group from the full state representation and measure the effect on performance. From the first block of Tab. 3, we can see that the most important factor groups were target and context objects, followed by stuff, whereas saliency and history weakly impacted model performance. In addition, an alternative state representation to DCB is the Cumulative Foveated Image (CFI) [56], but replacing DCB with CFI degrades the performance of IRL (as shown in the second block of Tab. 3).

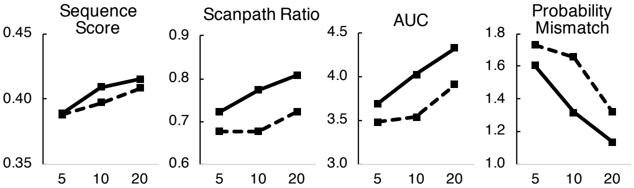

Figure 10: Performance of IRL (solid line) and BC-CNN (dashed line) as the number of training images per category increases from 5 to 20. IRL is more data efficient than BC-CNN, likely due to an adversarial data generator.

4.5. Data Efficiency

Fig. 10 shows IRL and BC-CNN performance as we vary the number of training images per object category. Both methods use DCB as the state representation. IRL is more data efficient than BC-CNN, achieving comparable or better results using less training data. A likely reason for this is that the GAIL-based [21] IRL method includes an adversarial component that generates augmented training data, leading to a less prone to overfitting policy network. Data efficiency is crucial for training for new categories, given the time and cost of collecting human fixations.

5. Conclusions

We proposed a new model for predicting search fixation scanpaths that uses IRL to jointly recover the reward function and policy used by people during visual search. The IRL model uses a novel and highly explainable state representation, *dynamic contextual beliefs (DCB)*, which updates beliefs about objects to obtain an object context that changes dynamically with each new fixation. To train and test this model we also introduced COCO-Search18, a large-scale dataset of images annotated with the fixations of people searching for target-object goals. Using COCO-Search18, we showed that the IRL model outperformed comparable models in predicting search scanpaths.

Better predicting human search behavior means better robotic search applications and human-computer systems that can interact with users at the level of their attention movements [43]. It may also be possible to use reward maps from the IRL model to annotate and index visual content based on what is likely to attract a person's attention. Finally, our work impacts the behavioral vision literature, where the visual features guiding human goal-directed attention are still poorly understood for real images [58].

Acknowledgements. This project is supported by US National Science Foundation Award IIS-1763981, the Partner University Fund, the SUNY2020 Infrastructure Transportation Security Center, and a gift from Adobe.

References

[1] Pieter Abbeel and Andrew Y. Ng. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the International Conference on Machine Learning*, 2004. 2

[2] Hossein Adeli and Gregory Zelinsky. Deep-bcn: Deep networks meet biased competition to create a brain-inspired model of attention control. In *CVPR Workshops*, 2018. 2

[3] Nicola C Anderson, Fraser Anderson, Alan Kingstone, and Walter F Bischof. A comparison of scanpath comparison methods. *Behavior research methods*, 47(4):1377–1392, 2015. 6

[4] Alper Aydemir, Kristoffer Sjöö, John Folkesson, Andrzej Pronobis, and Patric Jensfelt. Search in the real world: Active visual object search based on spatial relations. In *Proceedings of the IEEE Conference Robotics and Automation*, pages 2818–2824. IEEE, 2011. 2

[5] Nicolas Ballas, Li Yao, Chris Pal, and Aaron Courville. Delving deeper into convolutional networks for learning video representations. *arXiv:1511.06432*, 2015. 5, 6

[6] David J Berg, Susan E Boehnke, Robert A Marino, Douglas P Munoz, and Laurent Itti. Free viewing of dynamic stimuli by humans and monkeys. *Journal of vision*, 9(5): 19–19, 2009. 1

[7] Sage EP Boettcher, Dejan Draschkow, Eric Dienhart, and Melissa L-H Võ. Anchoring visual search in scenes: Assessing the role of anchor objects on eye movements during visual search. *Journal of vision*, 18(13):11–11, 2018. 3

[8] Ali Borji and Laurent Itti. State-of-the-art in visual attention modeling. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):185–207, 2013. 1

[9] Ali Borji, Ming-Ming Cheng, Huaizu Jiang, and Jia Li. Salient object detection: A benchmark. *IEEE transactions on image processing*, 24(12):5706–5722, 2015. 1

[10] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Cocostuff: Thing and stuff classes in context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1209–1218, 2018. 3

[11] Moran Cerf, Jonathan Harel, Wolfgang Einhäuser, and Christof Koch. Predicting human gaze using low-level saliency combined with face detection. In *Advances in neural information processing systems*, pages 241–248, 2008. 4

[12] Marcella Cornia, Lorenzo Baraldi, Giuseppe Serra, and Rita Cucchiara. Predicting human eye fixations via an lstm-based saliency attentive model. *IEEE Transactions on Image Processing*, 27(10):5142–5154, 2018. 1

[13] Richard Dewhurst, Marcus Nyström, Halszka Jarodzka, Tom Foulsham, Roger Johansson, and Kenneth Holmqvist. It depends on how you look at it: Scanpath comparison in multiple dimensions with multimatch, a vector-based approach. *Behavior research methods*, 44(4):1079–1100, 2012. 6

[14] Miguel P Eckstein. Visual search: A retrospective. *Journal of vision*, 11(5):14–14, 2011. 1

[15] Krista A Ehinger, Barbara Hidalgo-Sotelo, Antonio Torralba, and Aude Oliva. Modelling search for people in 900 scenes: A combined source model of eye guidance. *Visual cognition*, 17(6-7):945–978, 2009. 2

[16] James H Elder, Yuqian Hou, Ronen Goldstein, and Fadi Dornaika. Attentive panoramic visual sensor, October 31 2006. US Patent 7,130,490. 1

[17] Justin Fu, Katie Luo, and Sergey Levine. Learning robust rewards with adversarial inverse reinforcement learning. *arXiv:1710.11248*, 2017. 2

[18] Syed Omer Gilani, Ramanathan Subramanian, Yan Yan, David Melcher, Nicu Sebe, and Stefan Winkler. Pet: An eye-tracking dataset for animal-centric pascal object classes. In *2015 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2015. 2

[19] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*. 2014. 2

[20] John M Henderson, James R Brockmole, Monica S Castelhano, and Michael Mack. Visual saliency does not account for eye movements during visual search in real-world scenes. In *Eye movements*, pages 537–III. Elsevier, 2007. 1

[21] Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. In *Advances in Neural Information Processing Systems*, pages 4565–4573, 2016. 2, 4, 8

[22] Michael C Hout and Stephen D Goldinger. Target templates: The precision of mental representations affects attentional guidance and decision-making in visual search. *Attention, Perception, & Psychophysics*, 77(1):128–149, 2015. 6

[23] Xun Huang, Chengyao Shen, Xavier Boix, and Qi Zhao. Salicon: Reducing the semantic gap in saliency prediction by adapting deep neural networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 262–270, 2015. 1

[24] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015. 6

[25] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1254–1259, 1998. 1

[26] Saumya Jetley, Naila Murray, and Eleonora Vig. End-to-end saliency mapping via probability distribution prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5753–5761, 2016. 1

[27] Ming Jiang, Shengsheng Huang, Juanyong Duan, and Qi Zhao. Salicon: Saliency in context. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 1, 2

[28] Tilke Judd, Krista Ehinger, Frédo Durand, and Antonio Torralba. Learning to predict where humans look. In *2009 IEEE 12th international conference on computer vision*, pages 2106–2113. IEEE, 2009. 4

[29] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6399–6408, 2019. 3

[30] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and*

Pattern Recognition, pages 9404–9413, 2019. 3, 5, 6

[31] Kathryn Koehler, Fei Guo, Sheng Zhang, and Miguel P Eckstein. What do saliency models predict? *Journal of vision*, 14(3):14–14, 2014. 1

[32] Srinivas SS Kruthiventi, Kumar Ayush, and R Venkatesh Babu. Deepfix: A fully convolutional neural network for predicting human eye fixations. *IEEE Transactions on Image Processing*, 26(9):4446–4456, 2017. 1

[33] Matthias Kümmerer, Lucas Theis, and Matthias Bethge. Deep gaze i: Boosting saliency prediction with feature maps trained on imagenet. *arXiv:1411.1045*, 2014.

[34] Matthias Kummerer, Thomas SA Wallis, Leon A Gatys, and Matthias Bethge. Understanding low-and high-level contributions to fixation prediction. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4789–4798, 2017. 1, 2, 8

[35] 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. 3, 4

[36] Christopher Michael Masciocchi, Stefan Mihalas, Derrick Parkhurst, and Ernst Niebur. Everyone knows what is interesting: Salient locations which should be fixated. *Journal of vision*, 9(11):25–25, 2009. 1

[37] Stefan Mathe and Cristian Sminchisescu. Actions in the eye: Dynamic gaze datasets and learnt saliency models for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 37(7):1408–1424, 2014. 2

[38] Ken Nakayama and Paolo Martini. Situating visual search. *Vision research*, 51(13):1526–1537, 2011. 1

[39] Venkatraman Narayanan and Maxim Likhachev. Perch: Perception via search for multi-object recognition and localization. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5052–5059. IEEE, 2016. 1

[40] Saul B Needleman and Christian D Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48(3):443–453, 1970. 6

[41] Andrew Y. Ng and Stuart Russell. Algorithms for inverse reinforcement learning. In *Proceedings of the International Conference on Machine Learning*, 2000. 2

[42] Dim P Papadopoulos, Alasdair DF Clarke, Frank Keller, and Vittorio Ferrari. Training object class detectors from eye tracking data. In *European conference on computer vision*, pages 361–376. Springer, 2014. 2

[43] Sohee Park, Arani Bhattacharya, Zhibo Yang, Mallesham Dasari, Samir R Das, and Dimitris Samaras. Advancing user quality of experience in 360-degree video streaming. In *2019 IFIP Networking Conference (IFIP Networking)*, pages 1–9. IEEE, 2019. 8

[44] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018. 6

[45] Jeffrey S Perry and Wilson S Geisler. Gaze-contingent realtime simulation of arbitrary visual fields. In *Human vision and electronic imaging VII*, volume 4662, pages 57–70. International Society for Optics and Photonics, 2002. 3

[46] Kevin Roth, Aurelien Lucchi, Sebastian Nowozin, and Thomas Hofmann. Stabilizing training of generative adversarial networks through regularization. In *Advances in neural information processing systems*, pages 2018–2028, 2017. 4

[47] John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015. 4

[48] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv:1707.06347*, 2017. 4

[49] Antonio Torralba, Aude Oliva, Monica S Castelhano, and John M Henderson. Contextual guidance of eye movements and attention in real-world scenes: the role of global features in object search. *Psychological review*, 113(4):766, 2006. 2

[50] Melissa Le-Hoa Võ, Sage EP Boettcher, and Dejan Draschkow. Reading scenes: How scene grammar guides attention and aids perception in real-world environments. *Current opinion in psychology*, 2019. 3

[51] Zhiguo Wang and Raymond M Klein. Searching for inhibition of return in visual search: A review. *Vision research*, 50 (2):220–228, 2010. 6

[52] Zijun Wei, Hossein Adeli, Minh Hoai, Gregory Zelinsky, and Dimitris Samaras. Learned region sparsity and diversity also predict visual attention. In *Advances in Neural Information Processing Systems*, 2016. 2

[53] Jeremy M Wolfe. Guided search 2.0 a revised model of visual search. *Psychonomic bulletin & review*, 1(2):202–238, 1994. 2

[54] Jeremy M Wolfe and Todd S Horowitz. Five factors that guide attention in visual search. *Nature Human Behaviour*, 1(3):0058, 2017. 1

[55] Markus Wulfmeier, Peter Ondruska, and Ingmar Posner. Maximum entropy deep inverse reinforcement learning. *arXiv:1507.04888*, 2015. 2

[56] Gregory Zelinsky, Zhibo Yang, Lihan Huang, Yupei Chen, Seoyoung Ahn, Zijun Wei, Hossein Adeli, Dimitris Samaras, and Minh Hoai. Benchmarking gaze prediction for categorical visual search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 2, 3, 8

[57] Gregory J Zelinsky. A theory of eye movements during target acquisition. *Psychological review*, 115(4):787, 2008. 2

[58] Gregory J Zelinsky, Yifan Peng, Alexander C Berg, and Dimitris Samaras. Modeling guidance and recognition in categorical search: Bridging human and computer object detection. *Journal of Vision*, 13(3):30–30, 2013. 2, 8

[59] Mengmi Zhang, Jiashi Feng, Keng Teck Ma, Joo Hwee Lim, Qi Zhao, and Gabriel Kreiman. Finding any waldo with zeroshot invariant and efficient visual search. *Nature communications*, 9(1):3730, 2018. 2, 3

[60] Brian D. Ziebart, Andrew Maas, J. Andrew Bagnell, and Anind K. Dey. Maximum entropy inverse reinforcement learning. In *AAAI Conference on Artificial Intelligence*, 2008. 2, 4

Supplementary Material: Predicting Goal-directed Human Attention Using Inverse Reinforcement Learning

Zhibo Yang[1], Lihan Huang[1], Yupei Chen[1], Zijun Wei[2], Seoyoung Ahn[1],
Gregory Zelinsky[1], Dimitris Samaras[1], Minh Hoai[1]
[1]Stony Brook University, [2]Adobe Inc.

Abstract

This document provides further details about the COCO-Search18 dataset (Sec. 1), Dynamic Contextual Beliefs (Sec. 2), and implementation (Sec. 3). We also include additional results from experiments and ablation studies, and interpretation (Sec. 4).

1. Details about COCO-Search18 Dataset

Data source: The COCO-Search18 dataset annotates COCO [6] with human gaze fixations made during a standard target-present (TP) or target-absent (TA) search task, where on each trial the search image either depicted the target (TP) or it did not (TA). All of the images were selected from the *trainval* set, and detailed descriptions of TP and TA image selection and gaze collection methods are provided below.

Target present image selection:

In addition to the exclusion criteria described in the main text, we also excluded images in which the target was highly occluded or otherwise difficult to recognize. Specifically, we only selected images in which the cropped target-object patch had a classification confidence >.99. To train this classifier, we cropped the target in each image (by bounding box) and used these image patches as positive samples. Same-sized image patches of non-target objects were used as negative samples. Negative samples were constrained to intersect with the target by 25% (area of intersection divided by area of target) so that they could serve as hard negatives for specific targets. More than 1 million cropped patched were collected and resized to 224x224 pixels, while keeping the original aspect ratio by padding. The classifier is fine-tuned from an ImageNet-pretrained ResNet-50 model with the last fully connected layer changed from 1000 outputs to 33 (32+“Negative”). Images with a classification score for the cropped target patch that was <.99 were excluded. This resulted in 18 categories with at least 100 images in each category, and 3131 images in total. As described in the main text, we conducted a final manual checking of the dataset to exclude images depicting digital clocks (5 images), so as to make the clock target category specific to analog clocks, and to remove images depicting content that participants might find objectionable. This latter criterion resulting in the exclusion of 30 images, 22 of which were from the toilet category.

Category	TP images	ACC	TA images	ACC
bottle	166	0.84	166	0.92
bowl	141	0.80	141	0.90
car	104	0.89	104	0.91
chair	253	0.89	253	0.64
clock	119	0.99	119	0.97
cup	276	0.92	276	0.76
fork	230	0.96	230	0.98
keyboard	184	0.92	184	0.98
knife	141	0.89	141	0.97
laptop	123	0.95	123	0.95
microwave	156	0.97	156	0.95
mouse	109	0.97	109	0.97
oven	101	0.91	101	0.93
potted plant	154	0.84	154	0.95
sink	279	0.97	279	0.94
stop sigh	126	0.95	126	0.99
toilet	158	0.99	158	1.00
tv	281	0.96	281	0.93
total/mean	3101	0.92	3101	0.92

Table 1: Number of images and response accuracy (ACC) for TP and TA images grouped by target category.

After implemented all exclusion criteria, we selected 3101 target-present images from 18 categories: bottle, bowl, car, chair, clock, cup, fork, keyboard, knife, laptop, microwave, mouse, oven, potted plant, sink, stop sign, toilet, tv. See Table 1 for the specific number of images in each category and the average response accuracy (ACC).

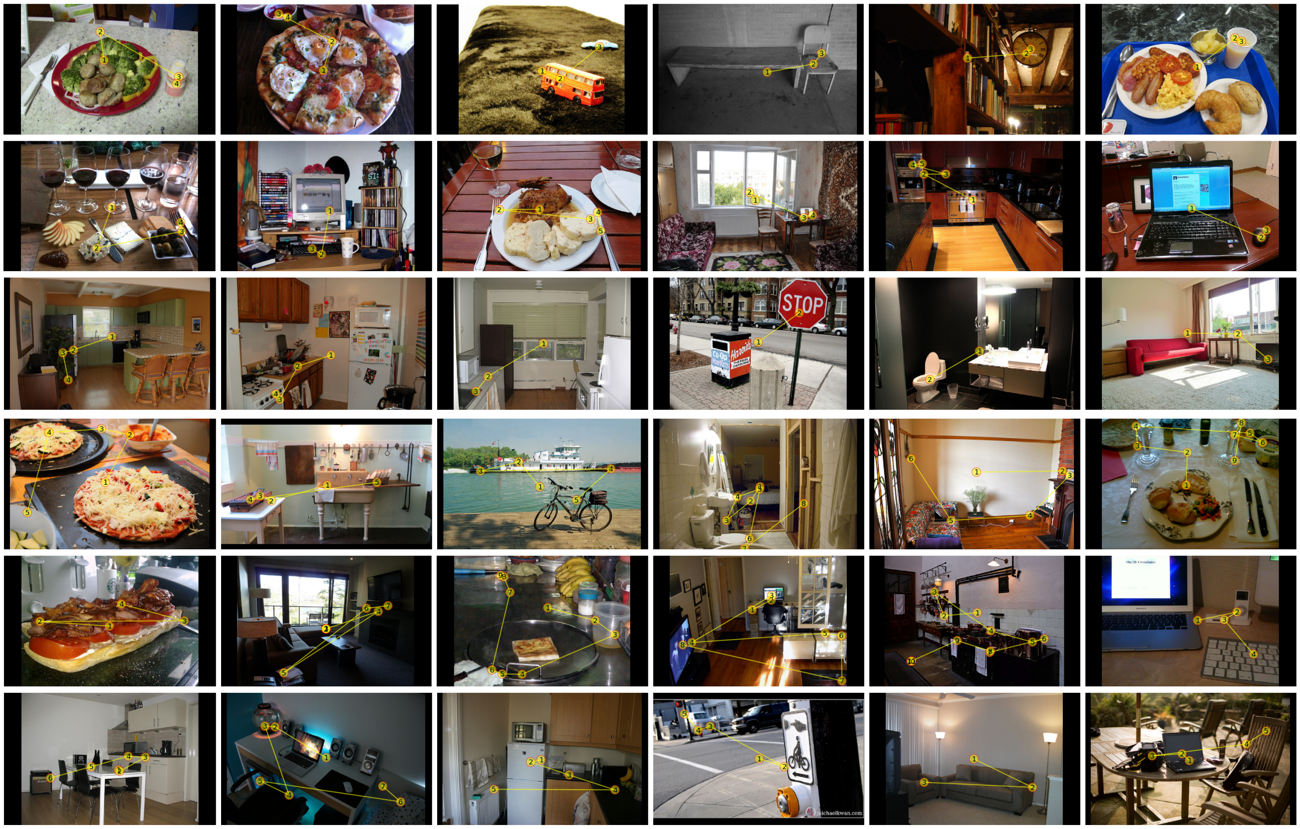

Figure 1: Examples of human scanpaths during target-present (top 3 rows) and target-absent (bottom 3 rows) visual search. From left to right and top to bottom, the 18 target categories are: bottle, bowl, car, chair, clock, cup, fork, keyboard, knife, laptop, microwave, mouse, oven, potted plant, sink, stop sign, toilet, and tv. Each yellow line represents the scanpath of one behavioral searcher, with numbers indicating fixation order.

There were an equal number of TA images (for a total of 6202 images), which were all resized and padded to fit the 1050×1680 resolution of the display monitor.

Gaze data collection procedure: Ten university undergraduate and graduate students (6 males, age range 18–30) with normal or corrected to normal vision participated in this study, which was approved by the Institutional Review Board. They were naive with respect to experimental question and design, and were compensated with course credits or money for their participation. Informed consent was obtained at the beginning of the experiment, and every participant read and understood the consent form before signing it.

The 6202 images were divided into six days of experiment sessions with each session consisting of $\sim$500 TP images and the same number of TA images, randomly interleaved. Images for a given target category were grouped and presented sequentially in an experiment block (i.e., target type was blocked). Preceding each block was a calibration procedure needed to map eye position obtained from the eye-tracker to screen coordinates, and a calibration was not accepted until the average calibration error was $\leq$.51 and the maximal error was $\leq$.94. Each trial began with a fixation dot appearing at the center of the screen. To start a trial, the subject should press the “X” button on a gamepad while carefully looking at the fixation dot. A scene would then be displayed and their task was to answer “yes” or “no” whether an exemplar of the target category for that block appears in the displayed scene. The subject registered a “yes” target judgment by pressing the right rigger of the gamepad, and a “no” judgment by pressing the left trigger. They were told that there were equal number of target present and absent trials, and that they should respond as quickly as possible while remaining accurate. Participants were allowed to take multiple breaks between and within each block.

Image presentation and data collection was controlled by Experiment Builder (SR research Ltd., Ottawa, Ontario, Canada). Images were presented on a 22-inch LCD monitor (resolution: 1050x1680), and subjects viewed these stimuli in a distance of 47cm from the monitor, enforced by both chin rest and head rest. Eye movements were recorded using an EyeLink 1000 eye tracker in tower-mount configuration (SR research Ltd., Ottawa, Ontario, Canada). The experiment was conducted in a quiet and dimmed laboratory room. Fig. 1 shows some TP and TA images from the 18 object categories, with overlaid human scanpaths.

2. Detailed Description of DCB

DCB: An input image is resized to 320×512 for computational efficiency (the original image is 1050×1680), while the blurred image is obtained by applying a Gaussian filter on the original image with the standard deviation $\sigma = 2$. Both images are passed through a Panoptic-FPN with backbone network ResNet-50-FPN pretrained on COCO2017 [4]. The output of the Panoptic-FPN has 134 feature maps, consisting of 80 "thing" categories (objects) and 54 "stuff" categories (background) in COCO. Feature maps are then resized to 20×32 spatially, same as the discretization of fixation history. At a given time step t, feature maps H for the original image and feature maps L for the blurred image are combined for DCB:

$$B_t = M_t \odot H + (1 - M_t) \odot L \tag{1}$$

where $\odot$ is element-wise product and M_t is the mask generated from fixation history and repeated over feature channels (see Fig. 2). Note that the above equation is equivalent to Eq. (1) in the main paper which is written in a recurrent form.

Encoding the target object category: The task embedding used in our model is the one-hot encoding maps which spatially repeat the one-hot vector. To make predictions conditioned on the task, inputs of each convolutional layer are concatenated with this embedding. This is equivalent to adding a task-dependent bias term for every convolutional layer.

3. Implementation details

Action Space. Our goal is to predict the pixel location where the person is looking in the image during visual search. To reduce the complexity of prediction, we discretize the image into a 20×32 grid, with each patch corresponding to 16×16 pixels in the original image coordinates. This descretized grid defines the action space for all models tested in this paper. At each step, the policy chooses one out of 640 patches and the center location of that selected patch in the original image coordinates is used as an action. The maximum approximation error due to this discretization procedure is 1.75 degrees visual angle.

IRL. The IRL model is composed of three components—the policy network, the critic network and the discriminator network. The **policy network** consists of four convolutional layers whose kernel sizes are 5, 3, 3, 1 with padding 2, 1, 1, 0 and output channels are 128, 64, 32 and 1, and a softmax layer to output a final probability map. The **critic network** has two convolutional layers of kernel size 3 and two fully-connected (fc) layers whose output sizes are 64 and 1. The convolutional layers have output sizes 128 and 256, respectively, and each is followed by a max pooling layer of kernel size 2 to compress the feature maps into a vector. Then this feature vector is regressed to predict the value of the state through two fc layers . The **discriminator network** shares the same structure with the IRL policy network except that the last layer is a sigmoid layer. Note that all convolutional layers and fully-connected layers are followed by a ReLU layer and a batch normalization layer [2] except the output layer.

The critic network is jointly trained with the policy network to estimate the value of a state (i.e., expected return) using smoothed L_1 loss. The estimated value is used to compute the advantage $A(S, a)$ (note that the state S is represented by the proposed DCB in our approach) in Eq. (4) of the main paper using the Generalized Advantage Estimation (GAE) algorithm [8]. At each iteration, the policy network first generates two scanpaths by sampling fixations from the current policy outputs for each image in a batch. Second, we break the generated scanpaths into state-action pairs and sample the same number of state-action pairs from ground-truth human fixations to train the discriminator network which discriminates the generated fixation from behavioral fixations. Lastly, we update the policy and critic network jointly using the PPO algorithm [9] by maximizing the total expected rewards which are given by the discriminator (see Eq. (3) of the main paper).

Training: The IRL model was trained for 20 epochs with an image batch size of 128. The batch sizes used for training the discriminator and policy networks were 64. For the PPO algorithm, the reward discount factor, the clip ratio and number of epochs were set to 0.99, 0.2, and 10, respectively. The extra discount factor in the GAE algorithm was set to 0.96. Both the policy network and the discriminator network were trained with a learning rate of 0.0005. It took approximately 40 minutes to train the proposed IRL model (for 20 epochs) on a single NVIDIA Tesla V100 GPU. The training procedure consumed about 5.6GB GPU memory. Note that the segmentation maps used to construct the DCB state representation had been computed beforehand.

Additional details on two baseline methods. **Detector**: The detector network consists of a feature pyramid network (FPN) for feature extraction (1024 channels) with a ResNet50 pretrained on ImageNet as the backbone and a convolution layer for detection of 18 different targets. The detector network predicts a 2D spatial probability map of the target from the image input and is trained using the

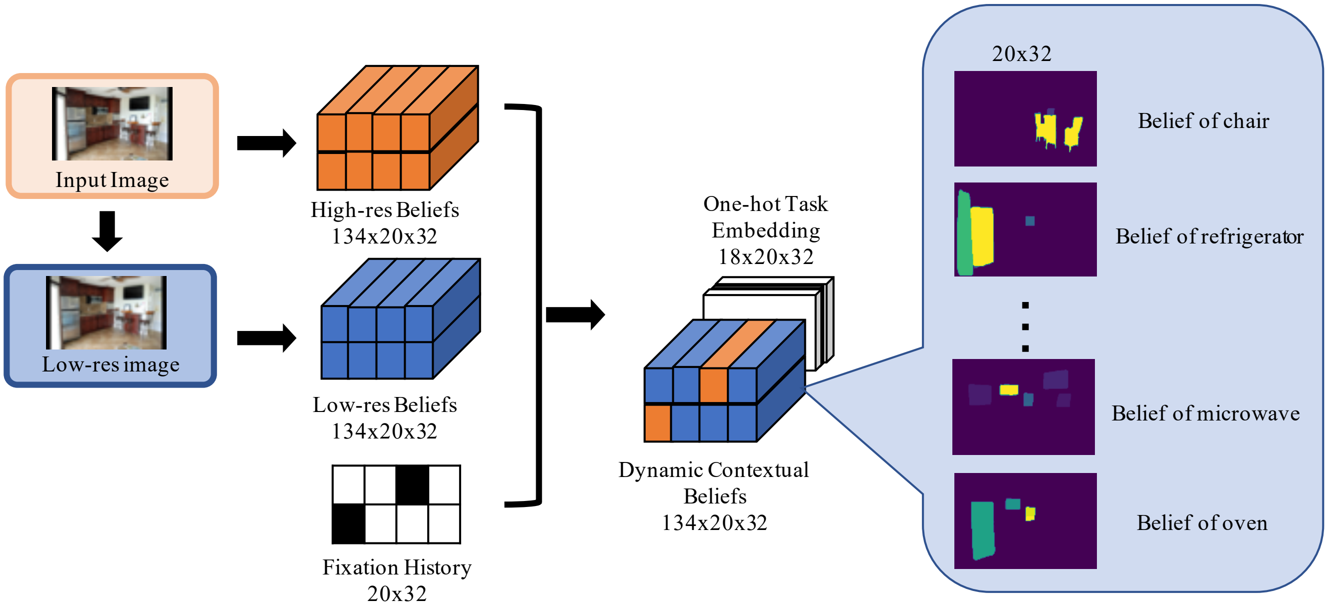

Figure 2: **Detailed illustration of Dynamic-Contextual-Belief**. First, an input image and its low-res image counterpart are converted into high-res beliefs and low-res beliefs. At each fixation, which is discretized into a binary fixation history map with 1's around the fixation location and 0's elsewhere, a new state is generated by concatenating the output of Eq. (1) with a one-hot task embedding (best viewed in color).

ground-truth location of the target. Another similar baseline is **Fixation Heuristics**. This network shares exactly the same network architecture with the detector baseline but it is trained with behavioral fixations in the form of spatial fixation density map (FDM), which is generated from 10 subjects on the training images.

Scanpath Generation. When generating scanpaths, a fixation location is sampled from the probability map that the models have produced and Inhibition-of-Return is applied to prevent revisiting previously attended locations. All predictive methods including IRL, behavior cloning, and heuristic methods, generate a new spatial probability map at every step, while the predicted probability map is fixed over all steps for the Detector and Fixation Heuristic baselines.

4. Additional Experiment Results

Cumulative distribution of sequence scores. In the main paper we reported the *average* Sequence Score of 0.422 for the scanpaths generated by the IRL model. To put this in perspective, Fig. 3 plots the cumulative distribution of the sequence scores and shows four qualitative examples that have sequence scores of 0.33, 0.40, 0.50, and 0.75, respectively.

Comparing different state representations. To evaluate the benefits of having DCB as the state representation, we compared its predictive performance with the Cumulative Foveated Image (CFI) [11] under the same IRL framework. CFI is created by extracting CNN feature maps on the retina-transformed images which are progressively more blurred based on the distance away from the currently fixated location. On the other hand, the DCB is created by extracting panoptic segmentations [3] on uniform-blur images which are uniformly blurred except around the fixated region (the level of blurriness applied in DCB is close to the middle level in the blur pyramid of CFI [1, 7, 11]). For a fair comparison, we extract features for CFI using the backbone ResNet-50-FPN network from the Panoptic-FPN [3] that was used in DCB. Both DCB and CFI have the same spatial resolution of 20×32. As shown in Tab. 2, the IRL model with DCB achieves significantly higher search efficiency and scanpath similarity than when using CFI as state representation. Specifically, DCB reduces the search gap by approximately 45% and improves the scanpath ratio from 61.9% to 82.6%, much closer to the human behavioral ratio of 86.2%. This result is even more impressive considering the size differences between the policy network used with DCB and CFI: DCB is trained with a smaller policy network, since it is comprised of 134 channels, nearly 8x smaller than CFI of 1024 channels. In our experiment, the policy network with CFI state representation has 29.6M parameters, while the policy network with DCB state representation only has 0.3M parameters. Relatedly, another benefit of having DCB as state representation is that it is memory and operation efficient. Creating DCB requires a smaller computational cost than creating CFI, since there's only a single level of blurriness in DCB

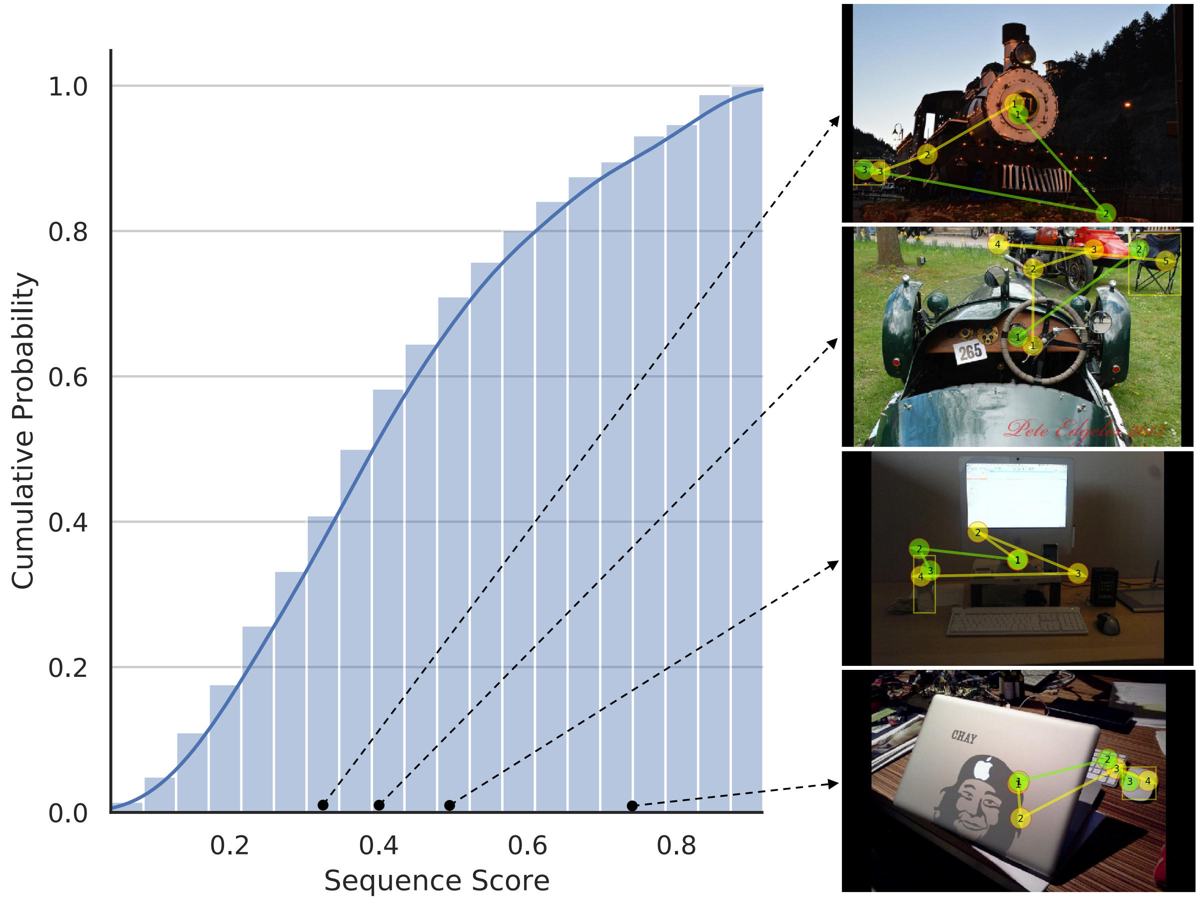

Figure 3: Left: cumulative distribution of the sequence scores of the proposed IRL scanpath prediction method. Right: Four qualitative examples. Human scanpaths are colored in yellow, and the IRL-generated scanpaths are in green. The sequence score for the generated scanpaths are 0.33, 0.40, 0.50, and 0.75, from top to bottom.

and extracted panoptic segmentation maps are smaller by an order of magnitude than the feature maps extracted for CFI. Given that IRL models are particularly difficult to train in high dimensional environments [10], having an efficient representation like DCB can be very helpful.

State Ablation. DCB is a rich representation that incorporates top-down, bottom-up, and history information. The full representation consists of 136 belief maps, which can be divided into five groups: target object (1 map), "thing" (object, 79 maps), "stuff" (background classes, 54 maps), saliency (1 map, extracted using DeepGaze2 [5]), and history (1 binary map for the locations of previous fixations). To understand the contribution of each factor, we removed the maps of each group one at a time and compared the resulting model's performance. As shown in Tab. 3, target object and "thing" maps are the most critical for generating human-like scanpaths, followed by "stuff" maps, whereas saliency and history do not have strong impact to the model performance.

Greedy vs. Non-greedy search behavior. How does human search behavior compare to generated scanpaths reflecting either Greedy or Non-greedy reward policies? Under the greedy policy, the selection of each location to fixate during search reflects a maximization of immediate reward. But the greedy policy is highly short-sighted – it only seeks reward in the short term. Non-greedy reward seeks to maximize the total reward that would be acquired over the sequence of fixations comprising a scanpath. This policy therefore does not maximize reward in the near term, but rather allows more exploration that leads to higher total reward. As shown in Tab. 4, we generated greedy and non-greedy policies from our IRL model and compared their predictive performance on human scanpaths. The results show that 1) models using greedy vs. non-greedy policy produce different search behaviors, with the model using non-greedy policy generating more human-like scanpaths by all tested metrics. This is an interesting finding. Despite the high efficiency of human search in our study (1-2 sec), the search process was strategic in that the fixations maxi-

State	Sequence	Scanpath	TFP-	Probability	MultiMatch ↑			
Representation	Score ↑	Ratio ↑	AUC ↑	Mismatch ↓	shape	direction	length	position
DCB	**0.422**	**0.826**	**4.509**	**0.987**	**0.886**	**0.695**	0.866	**0.885**
CFI	0.402	0.619	3.412	1.797	0.875	0.666	0.864	0.857

Table 2: **Dynamic contextual belief (DCB) vs. cumulative foveated image (CFI)** under the framework of IRL.

State	Sequence	Scanpath	TFP-	Probability	MultiMatch ↑			
Representation	Score ↑	Ratio ↑	AUC ↑	Mismatch ↓	shape	direction	length	position
DCB with all components	0.422	0.803	4.423	1.029	0.880	0.676	0.841	0.888
w/o history map	0.419	0.800	4.397	1.042	0.882	0.672	0.844	0.887
w/o saliency map	0.419	0.795	4.403	1.029	0.880	0.675	0.840	0.887
w/o stuff maps	0.407	0.777	4.111	1.248	0.876	0.662	0.836	0.875
w/o thing maps	0.331	0.487	2.047	3.152	0.855	0.605	0.852	0.818
w/o target map	0.338	0.519	2.274	2.926	0.866	0.613	0.837	0.820

Table 3: **Ablation study of the proposed state representation—dynamic contextual belief**. The full state consists of 1 history map, 1 saliency map, 54 stuff maps, 79 context maps and 1 target map. We mask out one part by setting the map(s) to zeros at each time.

Scanpath	Sequence	Scanpath	TFP-	Probability	MultiMatch ↑			
generation policy	Score ↑	Ratio ↑	AUC ↑	Mismatch ↓	shape	direction	length	position
Based on total reward	**0.422**	**0.826**	**4.509**	**0.987**	**0.886**	**0.695**	0.866	**0.885**
Based on immediate reward	0.375	0.704	3.893	2.143	0.886	0.648	**0.873**	0.852

Table 4: **IRL model predictions using Greedy (immediate reward) and Non-greedy (total reward) policy**.

	Sequence	Scanpath	TFP-	Probability	MultiMatch ↑			
	Score ↑	Ratio ↑	AUC ↑	Mismatch ↓	shape	direction	length	position
IRL, 20 ipc	0.415	0.808	4.324	1.140	0.875	0.672	0.832	0.879
CNN, 20 ipc	0.408	0.723	3.906	1.325	0.884	0.664	0.849	0.878
IRL, 10 ipc	0.409	0.774	4.029	1.318	0.881	0.591	0.851	0.819
CNN, 10 ipc	0.397	0.678	3.542	1.657	0.877	0.594	0.847	0.821
IRL, 5 ipc	0.389	0.723	3.696	1.603	0.876	0.588	0.844	0.813
CNN, 5 ipc	0.388	0.678	3.484	1.731	0.886	0.594	0.862	0.828

Table 5: **Data efficiency of IRL and CNN**. "ipc" means images per category used for training. For exmaple, IRL 10 ipc means we train the IRL model using 10 images from each category which are randomly selected from the training data. CNN and IRL are trained and tested on the same images for fair comparison.

mized total reward, even over that short period of time.

Data Efficiency. Table 5 shows the full results of IRL and BC-CNN given different numbers of training images across different metrics. Both use DCB as the state representation. The results are consistent with the results presented in the main paper and suggest that IRL is more data-efficient when compared to the CNN – IRL achieved comparable or better results than the CNN using less training data.

References

[1] Xun Huang, Chengyao Shen, Xavier Boix, and Qi Zhao. Salicon: Reducing the semantic gap in saliency prediction by adapting deep neural networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 262–270, 2015. 4

[2] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015. 3

[3] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6399–6408, 2019. 4

[4] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9404–9413, 2019. 3

[5] Matthias Kummerer, Thomas SA Wallis, Leon A Gatys, and Matthias Bethge. Understanding low-and high-level contri-

butions to fixation prediction. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4789–4798, 2017. 5

[6] 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. 1

[7] Jeffrey S Perry and Wilson S Geisler. Gaze-contingent real-time simulation of arbitrary visual fields. In *Human vision and electronic imaging VII*, volume 4662, pages 57–70. International Society for Optics and Photonics, 2002. 4

[8] John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015. 3

[9] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 3

[10] Aaron Tucker, Adam Gleave, and Stuart Russell. Inverse reinforcement learning for video games. *arXiv preprint arXiv:1810.10593*, 2018. 5

[11] Gregory Zelinsky, Zhibo Yang, Lihan Huang, Yupei Chen, Seoyoung Ahn, Zijun Wei, Hossein Adeli, Dimitris Samaras, and Minh Hoai. Benchmarking gaze prediction for categorical visual search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 4