CENTER FOR
Brains
Minds+
Machines

CBMM Memo No. 096 **July 31, 2018**

What am I searching for?

Mengmi Zhang[1,2,3], Jiashi Feng[2], Joo Hwee Lim[3], Qi Zhao[4] and Gabriel Kreiman[1]

1: Children's Hospital, Harvard Medical School
2: National University of Singapore
3: A*STAR, Singapore
4: University of Minnesota

Abstract

Can we infer intentions and goals from a person's actions? As an example of this family of problems, we consider here whether it is possible to decipher what a person is searching for by decoding their eye movement behavior. We conducted two human psychophysics experiments on object arrays and natural images where we monitored subjects' eye movements while they were looking for a target object. Using as input the pattern of "error" fixations on non-target objects before the target was found, we developed a model (InferNet) whose goal was to infer what the target was. "Error" fixations share similar features with the sought target. The Infernet model uses a pre-trained 2D convolutional architecture to extract features from the error fixations and computes a 2D similarity map between the error fixation and all locations across the search image by modulating the search image via convolution across layers. InferNet consolidates the modulated response maps across layers via max pooling to keep track of the sub-patterns highly similar to features at error fixations and integrates these maps across all error fixations. InferNet successfully identifies the subject's goal and outperforms all the competitive null models, even without any object-specific training on the inference task.

This material is based upon work supported by the Center for Brains, Minds and Machines (CBMM), funded by NSF STC award CCF-1231216.

What am I searching for?

Mengmi Zhang
Children's Hospital, Harvard Medical School
National University of Singapore, A*STAR, Singapore

Jiashi Feng
National University of Singapore

Joo Hwee Lim
A*STAR, Singapore

Qi Zhao
University of Minnesota

Gabriel Kreiman
Children's Hospital
Harvard Medical School *

Abstract

Can we infer intentions and goals from a person's actions? As an example of this family of problems, we consider here whether it is possible to decipher what a person is searching for by decoding their eye movement behavior. We conducted two human psychophysics experiments on object arrays and natural images where we monitored subjects' eye movements while they were looking for a target object. Using as input the pattern of "error" fixations on non-target objects before the target was found, we developed a model (InferNet) whose goal was to infer what the target was. "Error" fixations share similar features with the sought target. The Infernet model uses a pre-trained 2D convolutional architecture to extract features from the error fixations and computes a 2D similarity map between the error fixation and all locations across the search image by modulating the search image via convolution across layers. InferNet consolidates the modulated response maps across layers via max pooling to keep track of the sub-patterns highly similar to features at error fixations and integrates these maps across all error fixations. InferNet successfully identifies the subject's goal and outperforms all the competitive null models, even without any object-specific training on the inference task.

1 Introduction

Eye movements reflect rich information about the complex cognitive states of the brain, including thought processes and goals [8, 7, 24, 15, 3, 4, 16, 33]. Additionally, with advanced eye-tracking technologies, it is now possible to monitor eye movements at high spatial and temporal resolution while controlling the task and visual environment. Therefore, eye movements provide a suitable arena to investigate how to infer a person's goals from their actions.

Our work addresses the challenging problem of inferring what the subject is looking for in the context of a visual search task by decoding their error fixations. We define *"error" fixations* as the non-target fixations before the target was found. Given these error fixations, the goal is to decode what the target is (Figure 1). Several studies have shown that the error fixations during visual search are not random: those fixations are more likely to be on objects and locations that are similar to the target [11, 1, 32].

With the advancement of eye-tracking technology in wearable devices, computational models to infer the search target from human eye movements have several important application domains, such as health care, interactive user interfaces, and virtual reality (VR). For example, gaining information of the sought object of interest would be invaluable for VR processors to provide timely feedback to players. As another example, compared with neural decoding methods based on electrode recordings

*To whom correspondence should be addressed: gabriel.kreiman@tch.harvard.edu

Preprint. Work in progress.

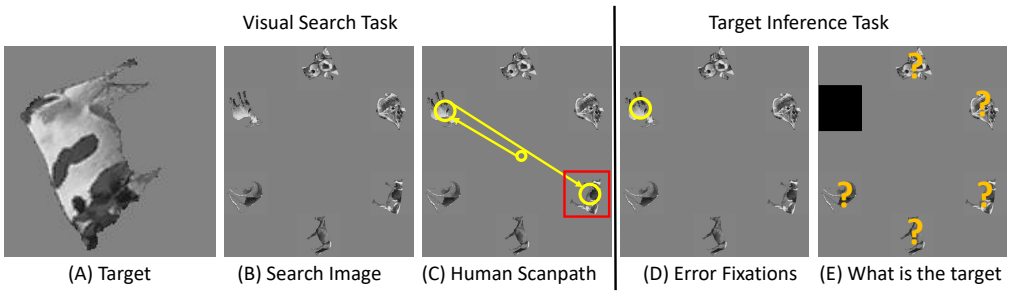


Figure 1: Illustration of the target inference problem. Human subjects were instructed to move their eyes to search for a given target (A) in the search image (B) irrespective of changes in size, rotation angles, or other format changes. The visual search task resulted in a sequence of fixations (C, yellow circles with the arrows). The red bounding box refers to the ground truth target location in the search image (not shown in the actual experiment). In this example, the subject required 2 fixations to find the target. We defined the fixations falling on the *non-target* objects as "error fixations". In the target inference task, given the error fixations recorded from the psychophysics visual search task (D, yellow circle), the model is asked to infer what target object the subject was searching for out of the remaining possible objects (E, question marks in orange color, the question marks are not shown to the computational model). In this example, there is only 1 error fixation, in general, there could be anywhere from 1 to 4 error fixations in these experiments with arrays of 6 objects.

inside human brains, decoding intentions in physically-disabled patients from eye movements is less invasive, has lower cost and significantly fewer potential complications.

To the best of our knowledge, there have been few attempts to build computational models that use eye fixation information for inferring what the search target is on complex natural images. To tackle this challenging problem, we proposed a zero-shot deep network, the Inference Network (InferNet). InferNet applies knowledge from an object recognition task on a target inference problem *without any retraining*. A likelihood map is computed based on feature similarity between the sub-patterns at the error fixations and the local patterns on the search image. InferNet then updates the belief of where the target of interest is across error fixations by cumulative addition of feature similarity maps modulated at each error fixation. We designed two sets of visual search experiments with object arrays and natural images, respectively, collected human eye movement data, and evaluated InferNet on these two datasets given the human error fixations in the search tasks. InferNet could successfully decode what the target was without any prior training on the inference task.

2 Related Works

Transfer learning. There is extensive work on networks that can leverage knowledge from one domain to a related task [25]. Examples of transfer learning include between-class transfer in the same task [2, 18, 31]; between task transfer, such as from classification to object detection [28, 27, 20] and image classification to semantic segmentation [21]. Our work focuses on task transfer by taking a network pre-trained for image classification and applying those weights on the target inference task *without any fine-tuning on this new task*.

Target decoding from fixations. Although information about a target is available in the fixation behavior during visual search, this does not imply that subjects are able to extract this information and use it to infer a search target [11, 1, 32]. Whether humans can infer the target information from other people's fixation behavior or not remains controversial. Some researchers have reported that it is possible to decode task information from eye movements [4, 14, 9, 23, 6, 26] while others have argued against otherwise [15, 13].

The focus in the current study is on designing a computational model capable of inferring what the subject's target is. There are a few studies on decoding target information in the context of visual search [5, 34, 26], but current methods are limited in using elementary search statistics [26] and handcrafted features [5, 34]. Moreover, existing approaches have only been tested with pre-defined object classes with constrained object set sizes. These computational models do not generalize to infer any target from arbitrary classes. In contrast, the InferNet model is capable of inferring any target on complex natural images.

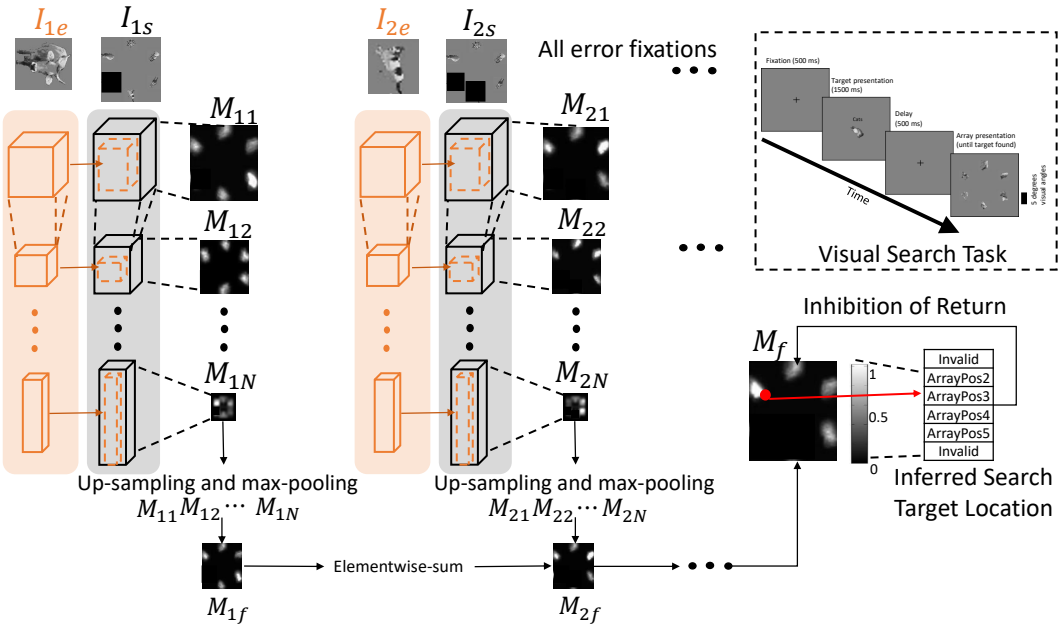


Figure 2: Architecture of InferNet. At each error fixation i, InferNet takes two inputs: the object I_{ie} at the error fixation and the search image I_{is} with the object at the error fixation inhibited with a black mask. The model consists a pre-trained deep convolutional network that processes the objects at the error fixations (**Prior Network** (orange shade)) and also processes the search image (**Likelihood Network** (gray shade)). The weights used to process the error fixations and the search images are identical and are pre-trained for image classification (see text). The **Prior Network** generates feature maps in each layer from the object at error fixations I_{ie} whereas the **Likelihood Network** generates feature maps in each layer for the search array image I_{is} via a 2D convolution neural network. Conditioned on the **Prior Network**, the **Likelihood Network** modulates the prior response maps by convolving the error fixation representation of I_{ie} with the feature maps from I_{is} at multiple layers, generating feature similarity maps M_{i1}, M_{i2}, ..., M_{iN}. These feature similarity maps are then resized, normalized and concatenated. We perform max-pooling across these maps to generate the consolidated feature similarity map M_{if}. This process is repeated for each error fixation i. The final probabilistic map M_f is the sum of all the individual error fixation maps. InferNet makes a decision on where the target is possibly located based on the maximum activation on M_f (red dot). An inhibition of return mechanism is applied if the target is not found at the current inferred location and the next maximum on M_f is selected. The error fixations are recorded from human subjects in the visual search task (Figure 1A-C). A schematic of the human psychophysics experiment in the visual search task is shown in the dash black box on the top right.

3 InferNet

We provide an overview of the model, followed by a more detailed description of our proposed zero-shot deep network (InferNet, Figure 2).

3.1 Overview

Error fixations share more visual feature similarities with the target than with distractors [11, 1, 32] (see also Supplementary Material for feature similarity comparison between pairs of targets and error fixations versus pairs of targets and random fixations). Thus, our model is based on the idea that the location with more feature similarities for all error fixations is more likely to be the search target location. We approximate the target inference problem in feature similarity space among targets and distractors: given T error fixations with coordinates (x_i, y_i) where $1 \leq i \leq T$, the task is to predict a 2D probabilistic map M_f of where the search target is most likely to be (Figure 2). We take the maximum on M_f as the current guess location. If the cropped area centered at the current guess location overlaps with the ground truth bounding box encompassing the whole target object, the inference is deemed successful; otherwise, after each incorrect guess, the map is updated by removing the erroneous inference location on M_f.

The model is based on a pre-trained deep convolutional network that is applied to the error fixations (**Prior Nework (PN)**) and to the search image (**Likelihood Network (LN)**). **PN** takes the cropped area I_{ie} of size 28×28 pixels centered at error fixation i as input and outputs feature maps across layers. We define I_{is} as the search image which has the objects at all past error fixations $1, ..., i$ inhibited with a black mask. **LN** modulates the feature maps from I_{is}, generating a series of likelihood maps (M_{i1}, M_{i2}, ..., M_{ij}, ...,M_{iN}) across different layers where j denotes the index of the jth attention map M_{ij} for error fixation i. These maps are concatenated and max-pooled to produce the final likelihood map M_{if} for error fixation i which tracks the parts of the image that are most similar between I_{ie} and I_{is}. InferNet integrates these likelihood maps M_{if} across all T error fixations via elementwise-sum by assuming all the error fixations play equally important roles in contributing to the final inference map M_f.

3.2 Prior Network

We used a deep feed-forward network, implemented in VGG16 [30], and pre-trained for image classification on the ImageNet dataset [29]. We show that the invariant features from VGG16 can be directly used for target inference task without any additional training. Given I_{ie} at error fixation i, the network weights W learnt from image classification extract feature maps $\varphi_j^{PN}(I_{ie}, W)$ at layer j (orange boxes in Figure 2).

3.3 Likelihood Network

Given I_{is}, **LN** has the same network parameters W as **PN** and extracts the feature representation of I_{is} at layer j, $\varphi_j^{LN}(I_{is}, W)$ (gray boxes in Figure 2). The weights are shared between **PN** and **LN**, and both are pre-trained for image classification, not for target inference. The weights W do *not* depend on I_{is} or I_{ie}. The InferNet network has no prior training with the objects or images in this study. The locations of the error fixations in I_{is} are blacked out (so that the model does not indicate that the most similar location to an error fixation is the error fixation itself). The input to **PN** is smaller than the input to **LN**, hence the output $\varphi_j^{PN}(I_{ie}, W)$ is smaller than $\varphi_j^{LN}(I_{is}, W)$. The activity of the units in **LN** in response to the search image is modulated by those in **PN**, which contain features more similar to the visual search target than distractors.

The modulation in the activation map is achieved by convolving the representation of the error fixation with the representation of the search image at multiple scales:

$$M_{ij} = m(\varphi_j^{SN}(I_{is}, W), \varphi_j^{PN}(I_{ie}, W)) \tag{1}$$

where $m(\cdot)$ is the error fixation modulation function defined as a 2D convolution operation with kernel $\varphi_j^{PN}(I_{ie}, W)$ on the search feature map $\varphi_j^{LN}(I_{is}, W)$ where j denotes the index of the jth feature similarity map M_{ij} for error fixation i.

Inspired by neurophysiological recordings during visual search and attentional modulation in visual cortex [10, 12] (see also discussion in [22]), and with the goal of capturing target properties at multiple scales and with different features, modulation is applied across multiple layers. Intuitively, if the target object shares more similarities with the error fixations in low-level features, such as similar orientations, error fixation modulation on M_{ij} may be sufficient; however, if high-level features are shared between the target and the error fixations, such as surface texture, feature similarity maps at higher levels may be required. We empirically selected $N = 7$ feature similarity maps (see details in Supplementary Material). In general, it is possible to select other layers based on specific applications, or even learn which layers to select for specific problems).

Each of these feature similarity maps is up-sampled to 224×224 pixels and the final feature similarity map is max pooled at each location (x, y) on M_{ij} over all the N intermediate maps (Table 1 reports performance separately for each feature similarity map). The model thus keeps track of all the locations which share similar sub-patterns including both low-level and high-level feature descriptors:

$$M_{if}(x, y) = \max_{j=1}^{N} M_{ij}(x, y) \tag{2}$$

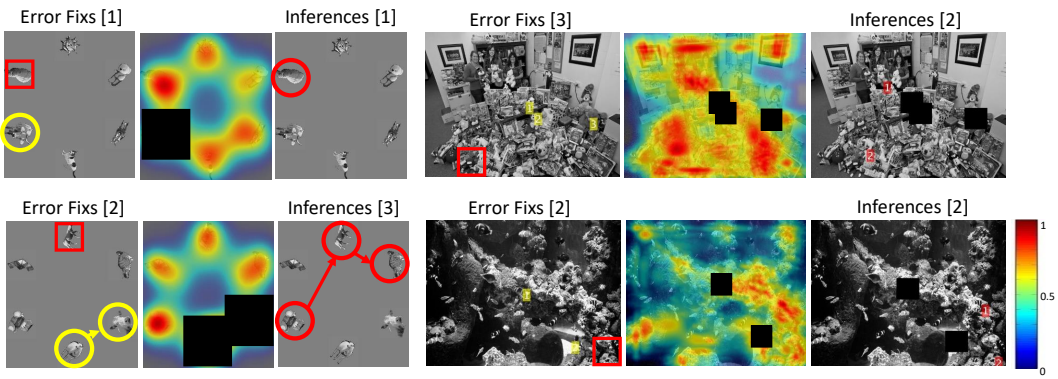


Figure 3: Two example results of target inference in object arrays (first 3 columns) and two examples in natural images (last 3 columns). Given the "error fixations" (yellow circles, column 1 and 4), the InferNet model predicts the 2D probabilistic map M_f overlaid on the stimuli (Columns 2 and 5, scale on the right). The red bounding box (Column 1, 4) denotes the ground truth area encompassing the search target. The red circles in Column 3 and black boxes in Column 6 show the successive maxima of the final inference map. InferNet correctly determined the target at the 1st and 3rd guess (Column 3) and in the second guess (Column 6).

3.4 Combination of maps and target inference

The feature similarity maps M_{if} are summed over all T error fixations:

$$M_f(x, y) = \sum_{i=1}^{T} M_{if}(x, y) \tag{3}$$

We assume all error fixations play equally important roles in inferring the search target. In general, it is possible to use a weighted summation where some error fixations are more important than the rest depending on the applications. InferNet selects the maximum of the M_f map. If the cropped area centered at the current guess location overlaps with the ground truth bounding box encompassing the whole target object, the inference is deemed successful and the inference stops. Otherwise, that location is inhibited and the next maximum is selected.

3.5 Evaluation

To evaluate performance of InferNet, we computed the average number of guesses required over all the trials with different images as a function of the number T of error fixations. The less number of guesses required, the more effective the inference process is. However, since the target inference difficulty varies, we report the relative performance P_r defined as the average number of guesses required by the computational model $A_m(T)$ relative to the average number of guesses required by a chance model $A_c(T)$ on the same image and task (see Section 4.2 for the chance model description):

$$P_r(T) = \frac{A_c(T) - A_m(T)}{A_c(T)} \times 100 \tag{4}$$

If the computational model requires less number of guesses on average, $P_r(T)$ is greater than zero. The larger $P_r(T)$, the more efficient the inference process is.

4 Experiments

We tested InferNet on images containing object arrays and also in natural images by evaluating the number of guesses required to correctly infer the sought target, $P_r(T)$. As benchmarks, we compared our model with other alternative null models, defined below. All the data (images, eye movements in visual search, source code) is publicly available: `https://github.com/kreimanlab/HumanIntentionInferenceZeroShot.git`.

4.1 Datasets

We designed two sets of psychophysics visual search tasks: object arrays and natural images. Ten subjects (5 in each task) were first presented with the exemplar target followed by the search image (see Figure 2 for schematic illustration of our psychophysics experiment). The target was always present for all trials. We used an EyeLink D1000 eyetracker (SR Research, Canada) to record eye movements during the visual search tasks. In the target inference task, we filtered out those fixations on targets and only used error fixations obtained prior to subjects locating the target in each trial. The appearance of the target object in the search image was different from that in the target image.

Object Arrays We selected segmented objects without occlusion from natural images in the MSCOCO dataset [19] from 6 categories: sheep, cattle, cats, horses, teddy bears and kites. Due to the uncontrolled and diverse nature of these stimuli, they may differ in low-level properties that could contribute to visual search performance. To minimize such contributions, we took several steps to normalize their low-level features (see Supplementary Material for details). Six objects (one per category) were uniformly arranged in a circle. There were 300 trials in total.

Natural Images To evaluate whether our model could generalize to infer the sought target in complex natural images, we collected 240 natural images from common object categories, such as animals (clownfish) and daily objects (alarm clock). In contrast to the object arrays experiment, here the objects were immersed in natural background and clutter and the object classes were not restricted to 6 categories. None of the images in the data set were taken from ImageNet, the dataset used to train VGG16. Moreover, there were 140 images out of the selected 240 images containing target objects whose categories are not part of ImageNet. In other words, these objects are novel to InferNet. The target object as rendered in the target image differed from the one rendered in the search image in terms of size, pose and rotation.

4.2 Comparative Null Models

We compared our model with several alternative null models. In all cases, the alternative models proposed an inference map and the procedure to select a target was the same as with InferNet, including infinite inhibition-of-return (i.e. never selecting the same location twice).

Chance. We considered a model where the target location was chosen at random. For object arrays, we randomly chose one out of the remaining possible locations. For the natural images dataset, a random location was selected for each guess. This random process was repeated 20 times.

Template Matching. To evaluate whether pixel-level features of the error fixations were sufficient for guiding inference, we introduced a pixel-level template matching model where the inference map was generated by sliding the canonical target of size 28×28 pixels over the whole search image of size 224×224 pixels. Compared to the classical sliding window models in computer vision, this can be interpreted as an "attentional" sliding window.

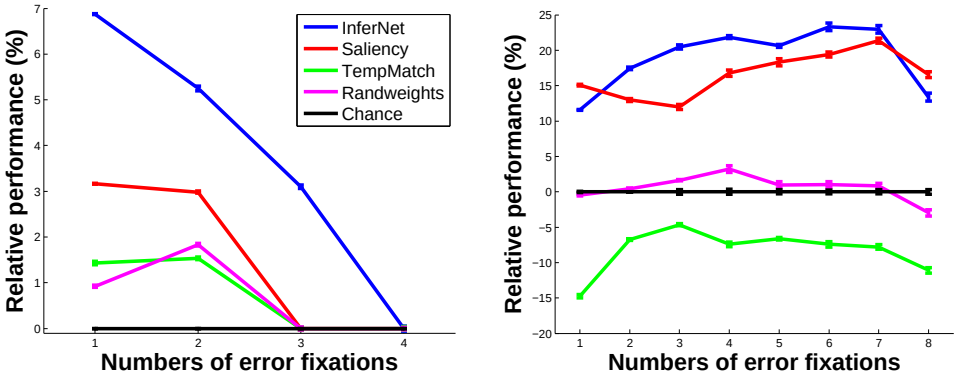
IttiKoch. We considered a pure bottom-up saliency model that has no information about the error fixations [17].

RanWeight. Instead of using VGG16 [30] pre-trained for image classification, we randomly picked weights W from a gaussian distribution with mean 0 and standard deviation 1000. The network was otherwise identical to InferNet. The random selection of weights was repeated 100 times.

4.3 Object arrays

Figure 3 shows examples illustrating how the model efficiently inferred the target location given only one or two fixations on object arrays. In the first example (Column 1-3, Row 1), a subject made one error fixation on the cow which looks visually similar to the sheep before finding the sheep. Given this single error fixation, InferNet determined that the subject was probably looking for a sheep among all the five remaining distractors (red circle, Column 3, Row 1). In the second example (Column 1-3, Row 2), a subject made 2 error fixations before finding the target (horse). In this case, InferNet correctly determined the target at the 3rd guess (Column 3, Row 2).

InferNet showed an overall improvement of $3.8 \pm 3\%$ with respect to the chance model over all error fixations (Figure 4a, blue). Even with a single error fixation as input data, InferNet could infer the



(a) Object arrays (b) Natural images

Figure 4: Evaluation of model inference performance for object arrays (a) and natural images (b). Relative performance improvement for the computational model relative to the chance model as a function of the number of error fixations. The smaller the number of guesses, the better the inference algorithm is and the higher the relative performance improvement. The different colors denote different models: InferNet model (blue), bottom-up IttiKoch saliency (red), template matching (green), RanWeight (magenta), Chance (black). See Section 4.2. Error bars are standard error of the mean for all trials.

target 6.87% faster than the chance model. That is, while random guessing would correctly land on the target within 3 guesses, InferNet only required 2.80 ± 0.01 guesses on object arrays.

In Figure 4a, none of the null models reached the level of relative performance improvement shown by InferNet ($P < 4.6 \times 10^{-20}$, two-tailed t-test, $t = -9.2$, $df = 12128$) for all the numbers of error fixations except for the case of 4 error fixations where none of the models were above chance. Performance for the bottom-up saliency model (IttiKoch) is better than the chance model but still below InferNet which suggests that the target information embedded in error fixations is useful for target inference. The model with random weights (RanWeight) and the model with template matching (TempMatch) on pixel levels show minimal improvements from selecting random locations (Figure 4a), suggesting the discriminative features learnt from a hierarchical network for image classification are important for target inference.

4.4 Natural scenes

The experiment reported so far focused on images consisting of segmented objects at discrete locations, presented on a uniform background, at fixed positions equidistant from the center of the image. In the real world, visual search happens most of the time in cluttered environments involving non-segmented objects amidst a complex background. As the inference space becomes continuous (the target object could be anywhere on the search image), the inference problem becomes more challenging and hence, there is higher demand for computational models to assist in target inference in these scenarios. To evaluate whether our model could generalize to complex natural scenes, we extended the previous results by evaluating the relative performance of InferNet in the natural images (Figure 3 and Figure 4b).

Figure 3 shows two examples where InferNet successfully determined the target in natural images. The appearance of the target in the search image is notably different from that in the target image due to changes in size and 3D rotation. Yet, the examples in Figure 3 show that InferNet can still effectively use features from error fixations to infer what the target is. For example, in Row 1, column 4, the error fixations fall on plush toys, such as teddy bears. Based on the characteristics of all plush toys, InferNet outputs an inference map with high activations around all the plush toys regions. In this example, InterNet correctly inferred the target within 2 guesses. In another example (Row 2, column 4), all the high activations on M_f focused on ground regions, such as the surface of coral reefs. InferNet can extract the essential texture information of ground surface under the sea and consider the features shared across all error fixations into account.

Figure 4b shows that InferNet was successful at inferring the target in natural images with significant improvements of $19 \pm 4\%$ compared with the chance model. In general, InferNet required an average

	Object Arrays				Natural Images							
#Error Fixations	1	2	3	4	1	2	3	4	5	6	7	8
InferNet (our model)	6.87	5.25	3.10	-	12.83	19.67	22.48	24.20	24.35	25.59	28.28	18.14
Layer 5	1.88	3.51	1.58	-	9.35	15.91	17.11	14.70	17.24	13.18	20.91	9.56
Layer 10	3.98	4.07	0.67	-	14.69	21.26	24.82	23.18	25.16	23.82	26.97	15.98
Layer 17	5.96	5.64	1.99	-	16.50	22.51	19.28	23.50	22.42	19.17	26.43	14.38
Layer 23	7.46	6.13	0.01	-	13.32	22.44	24.72	22.33	28.07	25.00	23.56	16.93
Layer 24	6.60	6.74	3.28	-	18.53	25.73	28.04	28.10	30.59	28.37	30.42	27.61
Layer 30	8.21	5.77	3.08	-	-	7.04	4.45	0.51	6.03	0.02	3.36	-
Layer 31	7.56	3.78	2.34	-	-	6.15	4.60	-	5.00	2.26	3.93	-
Max + Max	6.87	3.99	1.13	-	12.84	19.40	21.11	22.13	22.96	21.75	24.49	20.01
Mean + Max	7.01	4.48	2.63	-	8.67	11.60	11.97	12.66	14.22	11.87	16.05	7.92
Mean + Mean	7.01	6.24	3.68	-	8.67	10.60	9.68	9.78	10.61	8.71	13.31	6.30

Table 1: Target inference relative performance (%) of ablated models compared with the chance model in object arrays and natural images given T error fixations (the larger, the better). (-) denotes performance not significantly better than chance.

of 16.2 ± 0.07 guesses given only one error fixation and 15 ± 0.6 guesses given 8 error fixations (blue) while the chance model required 18.2 guesses given only one error fixation and 17.3 guesses given 8 error fixations. As we observed in Figure 4b, InferNet outperformed all the alternative null models ($P < 4 \times 10^{-27}$, two-tailed t-test, $t = -10.8$, $df = 140422$). Performances for the bottom-up saliency model (IttiKoch) was relatively high among all the null models because target objects were typically salient and they occupied a large percentage of the image.

We also observed that given more error fixations, the average number of guesses required to infer the target of interest was reduced. This effect can be ascribed to two factors: (i) the hypothesis space, *i.e.* number of location choices on the search image, is reduced with more error fixations, and (ii) more error fixations provide richer information that is useful for target inference.

4.5 Ablation study

To evaluate the contribution of different layers of InferNet, we tested each individual feature similarity map M_j and their different combinations in object arrays and natural images. Table 1 shows our ablated models' relative performance compared with the chance model using feature similarity maps (M_j) at different layers j for T error fixations. The layer number refers to the index in the VGG16 network [30]. The first row M_f corresponds to our full model considering all feature similarity maps across layers whereas the other rows show the predictions using either only one feature similarity map from M_{i1} to M_{i7} in Figure 2 or their combinations.

From Table 1, we have several observations: (1) Compared to the individual maps, target inference performance was generally more effective using the feature similarity maps M_j in higher layers which implies that high-level features extracted at error fixations are more reliable for target inference. (2) We are also interested in exploring how the compositionality of feature similarity maps across layers reveals the identity of the target. InferNet takes max-pooling of M_{ij} for error fixation i and averages M_{if} for all T error fixations. Instead of max-pooling across layers, we also evaluated ablated models where the max-pooling across N layers is replaced by averaging and vice versa. We did not observe any significant improvements in object arrays but different combination methods of feature similarity maps contribute dramatically differently in natural images. Our InferNet model outperforms the rest which suggests error fixations seem not to be guided by the overall target features as a whole (taking average across N layers) but by sub-patterns of the search target (max-pooling across N layers) which aligns with [26]. (3) Our InferNet model treats all error fixations equally and only utilizes the visual feature information at the error fixations. In the last ablated model, we study the role of the locations and the sequence order of error fixations in target inference (see Supplementary Material). It is surprising that the experimental result seems to suggest the location and order information of error fixations do not matter much in target inference task.

5 Conclusion

We proposed a computational model to infer intentions from behaviors in the context of a visual search task. The InferNet model can determine what the sought target is, in object array images as well as in natural images, by using the prior set of non-target fixations. InferNet is based on transfer-learning in that it uses weights learnt for a different task. InferNet is a "zero-shot" architecture: there is no

training with the specific objects or images that the model analyzes during the inference process. Leveraging on the idea that error fixations share feature similarities with the targets, InferNet builds an implicit relationship between the inference problem and the feature similarity problem. The experimental results show that InferNet significantly outperforms the comparative null models.

There are many areas where the model could be improved. Most notably, inference could be enhanced by incorporating intuitive semantics in the real world (*e.g.* if the error fixations are mostly distributed on the ground, one could deduce that the target of interest would most likely not be the airplanes in the sky). Problem-specific training (e.g. weights for each layer, or weights for each error fixation) could also improve performance. The proof-of-principle demonstration in this study provides a possible inference solution to effectively guess what the subject is searching for in complex images and suggests that computational models can make reasonable conjectures to read the subject's mind purely based on behavioral data.

References

[1] R. G. Alexander and G. J. Zelinsky. Visual similarity effects in categorical search. *Journal of Vision*, 11(8):9–9, 2011.

[2] Y. Aytar and A. Zisserman. Tabula rasa: Model transfer for object category detection. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2252–2259. IEEE, 2011.

[3] T. Betz, T. C. Kietzmann, N. Wilming, and P. König. Investigating task-dependent top-down effects on overt visual attention. *Journal of vision*, 10(3):15–15, 2010.

[4] A. Borji and L. Itti. Defending yarbus: Eye movements reveal observers' task. *Journal of vision*, 14(3):29–29, 2014.

[5] A. Borji, A. Lennartz, and M. Pomplun. What do eyes reveal about the mind?: Algorithmic inference of search targets from fixations. *Neurocomputing*, 149:788–799, 2015.

[6] A. Borji, H. R. Tavakoli, D. N. Sihite, and L. Itti. Analysis of scores, datasets, and models in visual saliency prediction. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 921–928. IEEE, 2013.

[7] G. Busswell. How people look at pictures: A study of the psychology of perception in art, 1935.

[8] M. S. Castelhano, M. L. Mack, and J. M. Henderson. Viewing task influences eye movement control during active scene perception. *Journal of vision*, 9(3):6–6, 2009.

[9] M. Cerf, J. Harel, A. Huth, W. Einhäuser, and C. Koch. Decoding what people see from where they look: Predicting visual stimuli from scanpaths. In *International Workshop on Attention in Cognitive Systems*, pages 15–26. Springer, 2008.

[10] R. Desimone and J. Duncan. Neural mechanisms of selective visual attention. *Annual review of neuroscience*, 18(1):193–222, 1995.

[11] M. P. Eckstein, B. R. Beutter, B. T. Pham, S. S. Shimozaki, and L. S. Stone. Similar neural representations of the target for saccades and perception during search. *Journal of Neuroscience*, 27(6):1266–1270, 2007.

[12] C. D. Gilbert and W. Li. Top-down influences on visual processing. *Nature Reviews Neuroscience*, 14(5):350–363, 2013.

[13] M. R. Greene, T. Liu, and J. M. Wolfe. Reconsidering yarbus: A failure to predict observers' task from eye movement patterns. *Vision research*, 62:1–8, 2012.

[14] A. Haji-Abolhassani and J. J. Clark. A computational model for task inference in visual search. *Journal of vision*, 13(3):29–29, 2013.

[15] J. M. Henderson, S. V. Shinkareva, J. Wang, S. G. Luke, and J. Olejarczyk. Predicting cognitive state from eye movements. *PloS one*, 8(5):e64937, 2013.

[16] S. T. Iqbal and B. P. Bailey. Using eye gaze patterns to identify user tasks. In *The Grace Hopper Celebration of Women in Computing*, pages 5–10, 2004.

[17] 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.

[18] J. J. Lim, R. R. Salakhutdinov, and A. Torralba. Transfer learning by borrowing examples for multiclass object detection. In *Advances in neural information processing systems*, pages 118–126, 2011.

[19] 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 *European conference on computer vision*, pages 740–755. Springer, 2014.

[20] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[21] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[22] T. Miconi, L. Groomes, and G. Kreiman. There's waldo! a normalization model of visual search predicts single-trial human fixations in an object search task. *Cerebral Cortex*, 26(7):3064–3082, 2015.

[23] T. O'Connell and D. Walther. Fixation patterns predict scene category. *Journal of Vision*, 12(9):801–801, 2012.

[24] A. Oliva, A. Torralba, M. S. Castelhano, and J. M. Henderson. Top-down control of visual attention in object detection. In *Image processing, 2003. icip 2003. proceedings. 2003 international conference on*, volume 1, pages I–253. IEEE, 2003.

[25] S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2010.

[26] U. Rajashekar, A. C. Bovik, and L. K. Cormack. Visual search in noise: Revealing the influence of structural cues by gaze-contingent classification image analysis. *Journal of Vision*, 6(4):7–7, 2006.

[27] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[28] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[29] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[30] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[31] T. Tommasi, F. Orabona, and B. Caputo. Safety in numbers: Learning categories from few examples with multi model knowledge transfer. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3081–3088. IEEE, 2010.

[32] J. M. Wolfe. Guided search 4.0. *Integrated models of cognitive systems*, pages 99–119, 2007.

[33] A. Yarbus. Eye movements and vision. 1967. *New York*, 1967.

[34] G. J. Zelinsky, Y. Peng, and D. Samaras. Eye can read your mind: Decoding gaze fixations to reveal categorical search targets. *Journal of vision*, 13(14):10–10, 2013.