University of California
Santa Barbara

Incorporating human visual properties into neural network models

A dissertation submitted in partial satisfaction
of the requirements for the degree

Doctor of Philosophy
in
Electrical and Computer Engineering

by

Aditya Jonnalagadda

Committee in charge:

Professor Miguel P. Eckstein, Chair
Professor Pradeep Sen
Professor William Yang Wang
Professor B. S. Manjunath

March 2023

The Dissertation of Aditya Jonnalagadda is approved.

Professor Pradeep Sen

Professor William Yang Wang

Professor B. S. Manjunath

Professor Miguel P. Eckstein, Committee Chair

March 2023

Incorporating human visual properties into neural network models

by

Aditya Jonnalagadda

Dedicated to all my teachers

Acknowledgements

I would like to take the opportunity to thank everyone who have contributed directly or indirectly to this work,

- Past and present lab members for their wonderful company,
- Undergraduate research assistants for their tireless efforts,
- Co-authors for their invaluable collaborations and
- Prof. Eckstein for greatly contributing to my overall growth.

Curriculum Vitæ

Aditya Jonnalagadda

Education

2023	Ph.D. in Electrical and Computer Engineering (Expected), University of California, Santa Barbara.
2017	M.S. in Electrical and Computer Engineering, University of California, Santa Barbara.
2012	B.Tech. in Electronics and Electrical Communication engineering, Indian Institute of Technology, Kharagpur.

Publications

Aditya Jonnalagadda, Miguel A. Lago, Craig Abbey, Miguel P. Eckstein Convolutional Neural Network Model Observers Discount Signal-like Anatomical Structures During Search in Virtual Phantoms (to be Submitted at IEEE-TMI) 2023

Aditya Jonnalagadda, William Yang Wang, B.S. Manjunath, Miguel P. Eckstein FoveaTer: Foveated Transformer for Image Classification (to be Submitted) 2023

Aditya Jonnalagadda, Miguel P. Eckstein A Foveated Vision-Transformer Model for Scene Classification, Vision Sciences Society (VSS) 2022

Aditya Jonnalagadda, Iuri Frosio, Seth Schneider, Morgan McGuire, and Joohwan Kim Robust Vision-Based Cheat Detection in Competitive Gaming (i3D 2021, ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games) 2021

Miguel A. Lago, **Aditya Jonnalagadda**, et al, Under-exploration of Three-Dimensional Images Leads to Search Errors for Small Salient Targets (Current Biology) 2021

Lauren E. Welbourne, **Aditya Jonnalagadda**, Barry Giesbrecht, and Miguel P. Eckstein The traverse occipital sulcus and intraparietal sulcus show neural selectivity to object-scene size relationships (Nature, Communications Biology) 2021

Aditya Jonnalagadda*, Miguel A. Lago*, et al, Evaluation of Convolutional Neural Networks for Search in $1/f^{2.8}$ Filtered Noise and Digital Breast Tomosynthesis Phantoms. Society of Photo-Optical Instrumentation Engineers (SPIE) 2020

Arturo Deza, **Aditya Jonnalagadda**, Miguel P. Eckstein Towards Metamerism via Foveated Style Transfer, International Conference on Learning Representations (ICLR) 2019

Aditya Jonnalagadda, Arturo Deza, Miguel P. Eckstein A foveated object detector that misses giant and misplaced targets in scenes, Vision Sciences Society (VSS) 2018

Patent

Aditya Jonnalagadda, Iuri Frosio, Joohwan Kim, Seth Schneider Graphics processing units for detection of cheating using neural networks. US20220180173A1, Filed December 7, 2020, Published June 9, 2022. *Patent pending* 2020

Honors and Awards

Jan 2023	Nominated for Outstanding Teaching Assistant Award by Academic Senate.
June 2022	Received ECE Dissertation Fellowship of $\$10,000$
Feb 2020	Received honourable mention for my poster at SPIE conference
May 2018	Received Outstanding ECE TA award for being one of the best Teaching Assistants and it included a cash prize of \$6000
2012 – 15	'Qualstar' Award – Award for excellence at Qualcomm on multiple occasions
Dec 2013	Promoted from associate engineer to Engineer within just 15 months
Oct 2011	Received pre-placement offer from Qualcomm
2009 - 12	Recipient of Means Cum Merit scholarship for three successive academic years (IIT Kharagpur)
June 2008	Secured All India Rank 562 out of 0.32 million in IIT-Joint Entrance Examination in 2008 (Percentile: 99.82)
June 2008	Secured State Rank 357 out of 0.325 million in EAMCET (Engineering and Medical Common Entrance Test) conducted by Andhra Pradesh Government (Percentile: 99.89)
June 2008	Secured All India rank 92 in B.Architecture exam in 2008

Abstract

Incorporating human visual properties into neural network models

by

Aditya Jonnalagadda

Many animals and humans process the visual field with varying spatial resolution (foveated vision) and use peripheral processing to make intelligent eye movements and point the fovea to acquire high-resolution information about objects of interest. A foveated architecture results in computationally efficient rapid scene exploration and can result in energy savings for computer vision. A foveated model can also serve as a proxy to identify circumstances in which humans might make an error and as a tool to understand human vision. Foveated architectures have been implemented into previous computer vision models. However, they have not been explored with the latest computer vision architecture transformer networks, which result in better robustness against adversarial attacks and better representation of spatial relationships across the entire image.

We propose foveated computational module for object classification (FoveaTer) and object detection (FST) integrated into the vision transformer architecture. We evaluate FoveaTer's computational savings and gains in robustness to adversarial attacks relative to a full-resolution model. We used the self-attention weights to optimize the guidance of the model eye movements. We have also investigated using FoveaTer to predict the various behavioral effects of humans. We performed a psychophysics experiment for the scene categorization task and predicted the human categorization performance using the FoveaTer model. Using two additional psychophysics experiments, a forced-fixation mouse recognition experiment to detect mouse in the visual periphery and a visual search

experiment to detect mouse using a limited number of fixations, We have also evaluated how the FST model uses contextual information to guide eye movements like humans.

In addition, we trained anthropomorphic CNN models to detect simulated tumors in simulated 3D Digital Breast Tomosynthesis phantoms and compare their performance and errors against that of radiologists. We provide preliminary results on extending the FST model for tumor search in virtual mammograms.

Thus, the contributions of the dissertation are to further the implementation of computational cost savings for computer vision, to predict perceptual errors of humans, and provide a computational tool to study human vision/cognitive science in the wild.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

Eye movements are a fundamental aspect of human vision and cognition. Humans make about three eye movements per second to explore the visual scene and follow up with decisions and/or actions, including identifying a face, localizing, and grabbing an object. Computational models have been fundamental to elucidating the visual properties and computations guiding eye movements and understanding human vision [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11], cognition [12], and visual and cognitive dysfunctions [13, 14, 15, 16, 17, 18]. However, even with the 20 years of the development of computational models of eye movements [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11], we still do not have adequate models of human eye movements that encapsulate all fundamental aspects of human vision and cognition during natural tasks. We do not have models that could generate human-like eye movements and perceptual decisions on a real-world image ("never-seen" by the model). To develop such a model, we must incorporate the varying fidelity of vision across the visual field (foveated visual system), the task the observer is engaged in, the representations of the visually relevant information for the task, the

contextual relationship among objects in scenes, and an eye movement algorithm that determines task-relevant fixations that maximize task performance. Furthermore, the model should apply to real-world images never seen by the model (image-computable) and output both a fixation sequence and a task-relevant decision after each fixation.

Such a model does not currently exist. Simple models that process bottom-up saliency or extensions of saliency [9, 11, 19, 20] do not take into account foveated vision [10, 11, 19, 20], the observer's task, contextual information, do not make task-relevant decisions, and have shown to be bad predictors of human fixations during tasks [1, 21, 22], actions [1, 2], and even free viewing [22]. Theoretically, powerful Bayesian foveated ideal models that predict eye movements that are optimally programmed to result in maximal perceptual performance, such as search [6, 7, 23, 24, 25, 26, 27] or face identification [28], have been instrumental in advancing the understanding of eye movements but are only computable for synthetic stimuli. They are simply not applicable to real-world images for which the image statistics are not known or not specified with simple probability density functions. Other computational models of eye movements have incorporated foveation [4, 29] but do not have the capabilities to be applied to "never-seen" real images, use heuristic eye movement algorithms rather than learned task-relevant eye movements, and have no understanding of object relationships and scene structure [4, 29].

Computer vision provides a powerful framework to develop the desired model, but until recently, the two reigning computer object detector frameworks (convolutional neural networks [30, 31, 32], CNNs, and the pre-CNN Deformable Parts Models [33, 34]) could not learn contextual information such as the relationship among objects [32] which is a crucial component of human scene processing and eye movements [10, 35, 36, 37, 38, 39, 40, 41]. There are also CNN-based models, the most well-known DeepGaze [42, 43], that are trained directly on datasets of images and associated human fixations and can generate fixation predictions for "never-seen" images. Such models might be good for

modeling free viewing but do not incorporate the task, foveation, or perceptual task performance. Perhaps the closest to what we envision was proposed by combining a Recurrent Neural Network (RNN) and reinforcement learning algorithm to optimize eye movements with digit recognition. However, the approach did not include a biologically plausible foveation, was not applied to real-world images, nor was integrated into an architecture that can learn scene context.

Finally, to our knowledge, none of the previous efforts has the capability to integrate foveated architecture, then execute eye movements and make perceptual decisions on natural images. We present the foveated vision transformer model that takes into account the foveated properties of human vision. The model understands scene structure/object relationships, executes task-information gathering eye movements, and makes perceptual decisions. The proposed framework could potentially allow making computational theoretical predictions of the functional impact of disruptions in eye movement strategies, visual deficits in early vision and cognitive deficits.

1.2 Summary of Contributions

There dissertation presents five significant contributions. First, we propose a novel Foveated Vision Transformer (FoveaTer), which performs the image classification task by using pooling regions and performing sequential fixations, whose locations are generated using the -model's internal self-attention weights. Second, we show the improved adversarial robustness of a foveated vision transformer compared to a baseline high-resolution (non-foveated) architecture. Third, we perform three psychophysics experiments to acquire human perceptual behavior for scene classification, mouse detection in the visual periphery, and visual search for the mouse under limited eye movements. Fourt, we propose Foveated Search Transformer (FST), an extension to the FoveaTer model for

visual search. We use it to simulate human perceptual behavior observed in the three psychophysics experiments and replicate the contextual effects related to the spatial location and scale in human subjects. Fifth, we train a CNN-based model observer for the virtual DBT phantoms for the case where the signal and background statistics are unknown and show that the CNN-based model observer is a better anthropomorphic linear model observers that are typically used in the medical imaging literature. Finally, we show how the FST model can be extended to virtual mammograms in 1/f noise using some preliminary results.

1.3 Dissertation Organization

In Chapter 2, We propose Foveated Transformer (FoveaTer) model, which uses pooling regions and eye movements to perform object classification tasks using a Vision Transformer architecture. Using square pooling regions or biologically-inspired radial-polar pooling regions, our proposed model pools the image features from the convolution backbone and uses the pooled features as an input to transformer layers. It decides on subsequent fixation locations based on the model's attention to various locations from past and present fixations. The model uses a confidence threshold to stop scene exploration. It dynamically allocates more fixation/computational resources to more challenging images before deciding the final image category. We then show an ensemble model combining the FoveaTer and the baseline non-foveated models. In the ensemble model, we use the non-foveated model when the FoveaTer model does not achieve the required confidence after making a pre-defined number of fixations, improving the model's accuracy. We perform a human psychophysics scene categorization task and use the experimental data to find a suitable radial-polar pooling region combination that predicts human results. We also show that the Foveated model better explains the human decisions in a scene

categorization task than the baseline non-foveated model. We show that integrating the foveated architecture and allowing the model to perform sequential data processing makes the model acquire better adversarial robustness against the Projected Gradient Descent (PGD) attack.

In chapter 3, we extend the previously proposed classification model, FoveaTer, for the task of visual search in order to explore the model's ability to simulate human perceptual behavior. To this end, we perform two psychophysics experiments to measure human subjects' detection performance for a computer mouse object. In the first forced-fixation experiment, we measured the decrease in human detection performance as a function of eccentricity, i.e., we measured the recognition performance by displaying the mouse at different distances from the fovea. In the second visual search experiment, the computer mouse could be present or absent at different parts of a scene. The participants were allowed to make eye movements, and the display was interrupted after one, two, or three saccades. We created the dataset used in the second experiment that manipulated the location of the mouse at expected & unexpected spatial locations and its size relative to the surrounding object (i.e., the mouse at expected & unexpected scales relative to the monitor and keyboard in the scene). For the psychophysics experiments, the human subjects' task was to decide on the presence or absence of the computer mouse, which is equivalent to an image-level classification decision. We modified the FoveaTer model to have two outputs corresponding to the computer mouse's presence or absence and trained the model's attention weights to adapt to the new task. We also used the biologically inspired radial-polar pooling regions and a larger input to the model than the FoveaTer model. We also reduced the downscaling image factor in the convolution backbone, thereby preserving a larger spatial size at the input of the transformer layers. We demonstrate the model's ability to approximate human behavior in terms of eccentricity performance, the effectiveness of the fixations made by the model as compared

to guidance by existing saliency models, and the display of spatial and scale contextual effects by the model similar to that of the humans.

In chapter 4, we switch from natural images to virtual phantoms of x-ray breast images (mammograms). For detecting a lesion in one of a few locations, linear model observers (e.g., Channelized Hotelling Observer) have been shown to predict human performance. Here we compare the detection accuracy of the Channelized Hotelling Observer (CHO) and a Convolution Neural Network (CNN) against the radiologists' performance for two types of signals embedded in 2D/3D breast tomosynthesis phantoms (DBT. In this work, we first trained three model observers and compared their performance against the radiologists' performance for two types of signals of clinical relevance, small microcalcification (CALC) and larger masses (MASS). The CNN's performance is comparable to or better than that of radiologists, while the linear model observer is worse than radiologists. An analysis of the eye position of radiologists showed that they fixated more often and longer times at the locations corresponding to CNN false positives than those of the linear model observer. False positives related to the phantom's anatomy from the linear model observer are not good predictors of radiologist fixation locations. In conclusion, we showed that CNN could be used as an anthropomorphic model observer for the search task where the traditional model observers fail due to complex backgrounds.

In chapter 5, as part of future work, we show preliminary results of applying the FST model on digital breast tomosynthesis phantoms (DBT) and virtual mammograms generated using 1/f noise.

Chapter 2

Foveated Transformer for Image Classification (FoveaTer)

2.1 Introduction

Many mammals, including humans, have evolved a locus (the fovea) in the visual sensory array with increased spatial fidelity and use head and eye movements [44, 45] to orient such locus to regions and objects of interest. The system design allows visual-sensing organisms to accomplish two objectives: fast target detection crucial for survival and savings in computational cost. Computational savings are accomplished by limiting the number of units with high computational costs (i.e., higher spatial resolution processing) to the fovea's small spatial region. Fast target detection is achieved by distributing the remaining computational power across a much larger area in the periphery, with a lower spatial resolution with increasing distance from the fovea. Critical to the design is an efficient algorithm to guide through eye movements the high-resolution fovea to regions of interest using the low-resolution periphery [46, 47, 48] and allow optimizing the target detection and scene classification. Various computational models were proposed

to model the search using foveated visual system [49, 50].

Computer vision has evolved from hand-crafted features to data-driven features in modern CNNs. Due to their computational limitations, the objectives of the computer vision systems align well with those of human visual system: to optimize visual detection and recognition with an efficient computational and metabolic footprint. Approaches toward saving computational power can be seen; for example, computer vision systems evolved from using sliding windows to RCNN's [51] use of selective search and Faster-RCNN's [52] use of Region Proposal Network (RPN).

A system that mimics human vision by processing the scene with a foveated system and rational eye movements has also been proposed. This approach to exploring the scene can be seen in models like RAM [53] for recognizing handwritten single-digits or detecting objects [54] where they sequentially process the image and decide what to process next by using the peripheral information. These foveated models approach that of full-resolution models but using a fraction of the computations. Foveated systems have also shown to result in more robustness [55, 56, 57, 58] against adversarial attacks.

There has been a recent innovation in computer vision using Transformers [59, 60] for object classification tasks that depart from the traditional over-reliance on convolutions. Even after replacing the convolutions with attention modules and multilayer perceptrons, Vision Transformers [60, 59] achieve close to state-of-the-art performance on the ImageNet dataset and provide better robustness against adversarial attacks [61].

Due to the flattened architecture of the transformers, it is easier for multi-resolution features to share the same feature channels. Transformers [62] have the added benefit of self-attention, which facilitates the interaction of various parts of the image irrespective of distance. No papers have evaluated the additional potential gains of incorporating a foveated architecture into Vision Transformers for the task of ImageNet classification.

Here, we evaluate the effect of a foveated architecture and sequential eye movements

on a state-of-the-art transformer architecture. We compare the Foveated transformer relative to the Baseline model in terms of classification accuracy and robustness to adversarial attacks. We perform a psychophysics experiment for a scene classification task and evaluate the Foveated model agreement with the human decision against that of the Baseline model. We first perform an object classification task using multiple fixations, moving foveal attention across different parts of the image, and using only a limited portion of the image information at each fixation, thereby reducing the input to the transformer by many folds. The model decides on subsequent fixation location using the self-attention weights accumulated from the previous fixations until the current step. Finally, the model makes the final classification decision.

2.2 Related work

Transformers have achieved great success in Natural Language Processing since their introduction by [62] for machine translation. Recently, the application of Transformer models in Computer Vision has seen tremendous success. Vision Transformer (ViT) model introduced by [60] achieved remarkable performance on ImageNet [63] by using additional data from JFT 300M [64] private dataset. Subsequently, the DeiT model [59] introduced knowledge transfer concepts in transformers to leverage the learning from existing models. Using augmentation and knowledge transfer, the DeiT model achieved close to state-of-the-art performance using training data from the ImageNet dataset alone.

Sequential processing provides three main advantages in computer vision. [65] proposed a model based on the Boltzmann machine that uses foveal glimpses and can make eye movements. First, it can limit the amount of information processed at a given instant to be constant, i.e., the ability to keep computations constant irrespective of the input image size. Second, sequential models can help model human eye movement strategies

and help transfer that information to build better computer vision systems. RAM [53] introduced a sequential model capable of making a sequence of movements across the image to integrate information before classification. In addition, the hard-attention mechanism, implemented using reinforcement learning, was used to predict the sequence of fixation locations. [66] extended these ideas to recognize multiple objects in the images on a dataset constructed using MNIST. Third, sequential processing requires fewer parameters than a model using full-resolution image input. Other models [67] have proposed image captioning models based on both hard-attention and soft-attention. Additionally, the spatial bias introduced into CNNs due to padding [68] can be overcome using sequential models [69]. On the flip side, sequential models might suffer longer processing times due to sequential processing and slow convergence times for reasons similar to RNNs [70].

Computational models of categorization and eye movements have been proposed for rapid categorization in terms of low-level properties such as spatial envelopes [71] and texture summary statistics [72]. Saliency-based models [73, 74, 75] traditionally tried to model eye movements by identifying bottom-up properties in the image that will capture attention. [76] showed how saliency could be combined with contextual information to guide eye movements. Low-resolution periphery and high-resolution central fields are integrated with saliency to predict human-like eye movements [77]. Data-driven scan path prediction models [43] train on image content and human fixations to predict the fixations under a free viewing but do not consider decision accuracy in specific tasks after multiple fixations. Goal-directed attention control [78] showed the dependency of search patterns on target features and scene context. [54] implemented a biologically-inspired foveated architecture [79] with a deformable parts model to build a foveated object detector on PASCAL dataset [80], whose accuracy was close to a full-resolution model but using a fraction of the computations. Spatial transformer networks [81], an older technique different from the proposed Vision Transfomers, were used on CIFAR-10

dataset [82], with foveation to improve object localization using foveated convolutions [83] and achieve better eccentricity performance [84] on MNIST dataset [85].

FoveaTer combines biologically-inspired foveated architecture with a Vision Transformer Network. Unlike the previous architectures [54, 86], we do not scale the image and thereby retain the parallelism with biological mechanisms. We apply our model to real-world images from the ImageNet dataset for image classification. In contrast, the previous works were mainly limited to datasets with small image sizes or a smaller number of output classes. They did not extend to large-scale real-world databases like ImageNet, which has 1000 class labels. We also evaluate the functional roles of various components through ablation studies, including the memory of foveal and peripheral information from previous fixations, inhibition of return, and eye movement guidance algorithms.

A novel aspect of the proposed work is that the model also learns that all images are not equally difficult to classify, adapting the exploration of eye movements to different images and thus varying computational resources used to classify different images successfully. The model implements this idea using a confidence threshold to restrict the scene exploration to the necessary fixations to classify the image.

Also novel is an evaluation of the adversarial robustness of our model to understand the contributions of the foveated architecture and that of sequential fixations towards defense against adversarial attacks. We use the projected gradient descent method [87, 88], which iteratively computes the adversarial image. The architectural changes may not be trivially transferable to a new architecture. End-to-End training and hyper-parameter settings might be needed to adapt to the architectural differences.

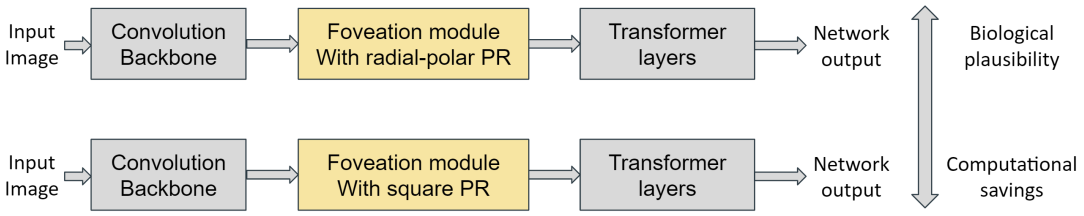

Figure 2.1: **Model architecture:** PR refers to the pooling regions. Foveation with radial-polar pooling regions is more biologically plausible than the square pooling regions but computationally slower and vice-versa.

2.3 Model

The model consists of three components, as shown in Figure 2.1 - convolution backbone, foveation module, and transformer layers. Interactions between different feature locations are limited to local regions in the convolution backbone. The Foveation module performs non-uniform pooling on the input features, reducing feature dimensionality. The Foveation module can contain two types of pooling regions, square pooling regions which provide computationally fast processing, or biologically plausible radial-polar pooling regions, [79]. Under this non-uniform average-pooling model, locations closer to the fixation location use smaller neighborhoods for pooling than locations far from the fixation location. The last component consisting of the transformer layers contributes in three ways - 1. They allow global interactions, which allows the possibility of using context-based decision-making. 2. They eliminate the need to design convolution layers on top of non-uniform sampled features from the Foveation module. 3. Self-attention weights of the transformer layers can be helpful in fixation guidance.

For the square pooling regions, the input image is first passed through the convolution backbone resulting in a feature vector of size $[384, 14, 14]$. After adding the sinusoidal position embedding and performing fixation-dependent average-pooling using the Foveation module, the feature size reduces to $[384, 22]$. Pooled features of size $[384, 22]$ are passed through the transformer layers, followed by the classification layer resulting in a logits

vector. We use the self-attention weights from the last transformer layer to predict the subsequent fixation location. We make five fixations on each image during the model training. This choice keeps the computational cost relatively the same as the Baseline model. Model architecture is shown in Figure 2.3. An alternate model architecture is shown in Appendix A.1.

The convolution backbone consists of six convolution layers and is structured similarly to the initial layers of the ResNet-18 model. Square pooling regions can exploit the fast average-pooling library functions, whereas the pooling in the radial-polar pooling regions needs custom implementation. Four architectural changes make it possible for the FoveaTer model to perform serial processing, achieve throughput improvements and retain information across fixations. Firstly, the Foveation module is a plug-in module that can be preceded or succeeded by the transformer layers. However, additional changes would be required for the convolution layers to follow the foveation module. Secondly, the periphery (i.e., the pooling regions other than Fovea) pools the feature vectors and, as a result, reduces the number of features processed by the subsequent layers. Thirdly, the attention-based fixation guidance mechanism (FGM) helps predict the subsequent fixation location using the attention values of current and past fixations. Lastly, the features from the past fixation's foveal locations are retained and processed along with the foveal and peripheral features of the current fixation. Thus, allowing the model to access memory. As an example, fixations made by the model are shown in Figure 2.2.

Retention of foveal features: The number of feature vectors processed by the Foveation module varies across fixations due to the retention of foveal features from past fixations. For each fixation, if the number of peripheral and foveal features are A and B, the number of features processed by the foveation module at Nth fixation equals A+NB.

Initial fixation for the Foveated model: The input feature map to the Foveation module has a spatial size of 14×14 for the condition of square pooling regions. All

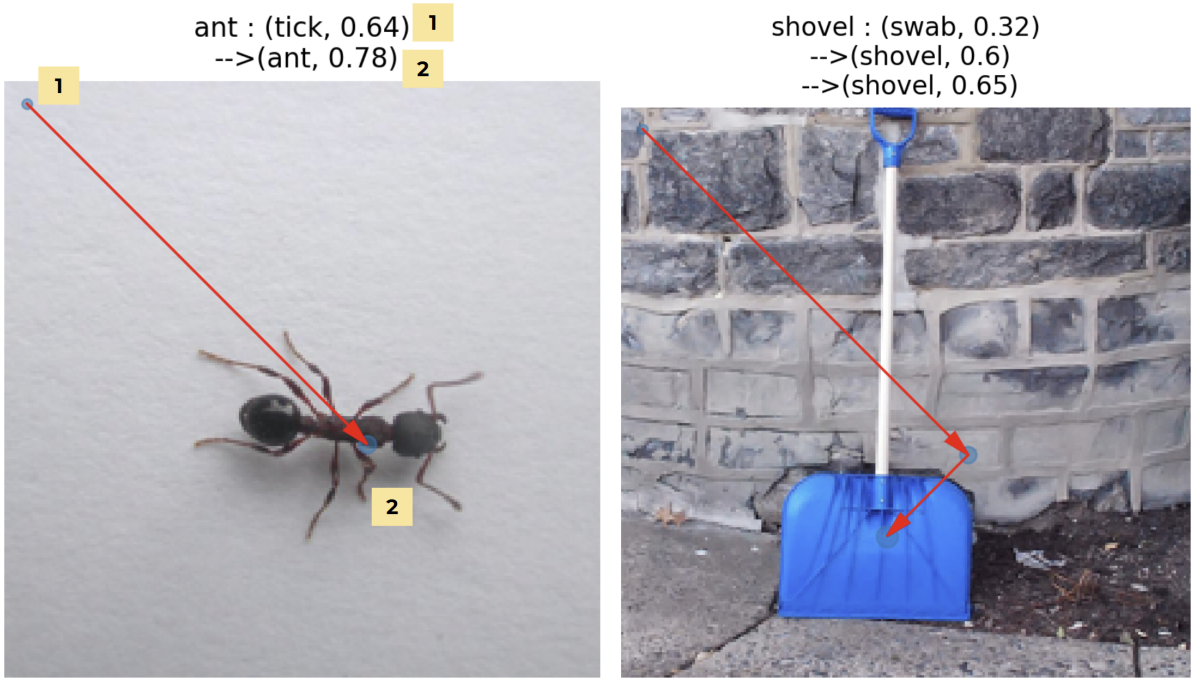

Figure 2.2: **Visualization of network fixations:** Model was trained with initial random fixation but tested with initial fixation at the top-left corner. **Left:** Due to the foveated nature of the model, after the initial fixation model, it decided that the image class is **tick**, and after making one more fixation, it came to the right decision that it is **ant**. **Right:** For this image, the network needed three dynamic fixations instead of two. It made the wrong decision with its initial fixation at the top-left corner. After making two more fixations, it came to the right decision, and was confident enough to terminate fixations.

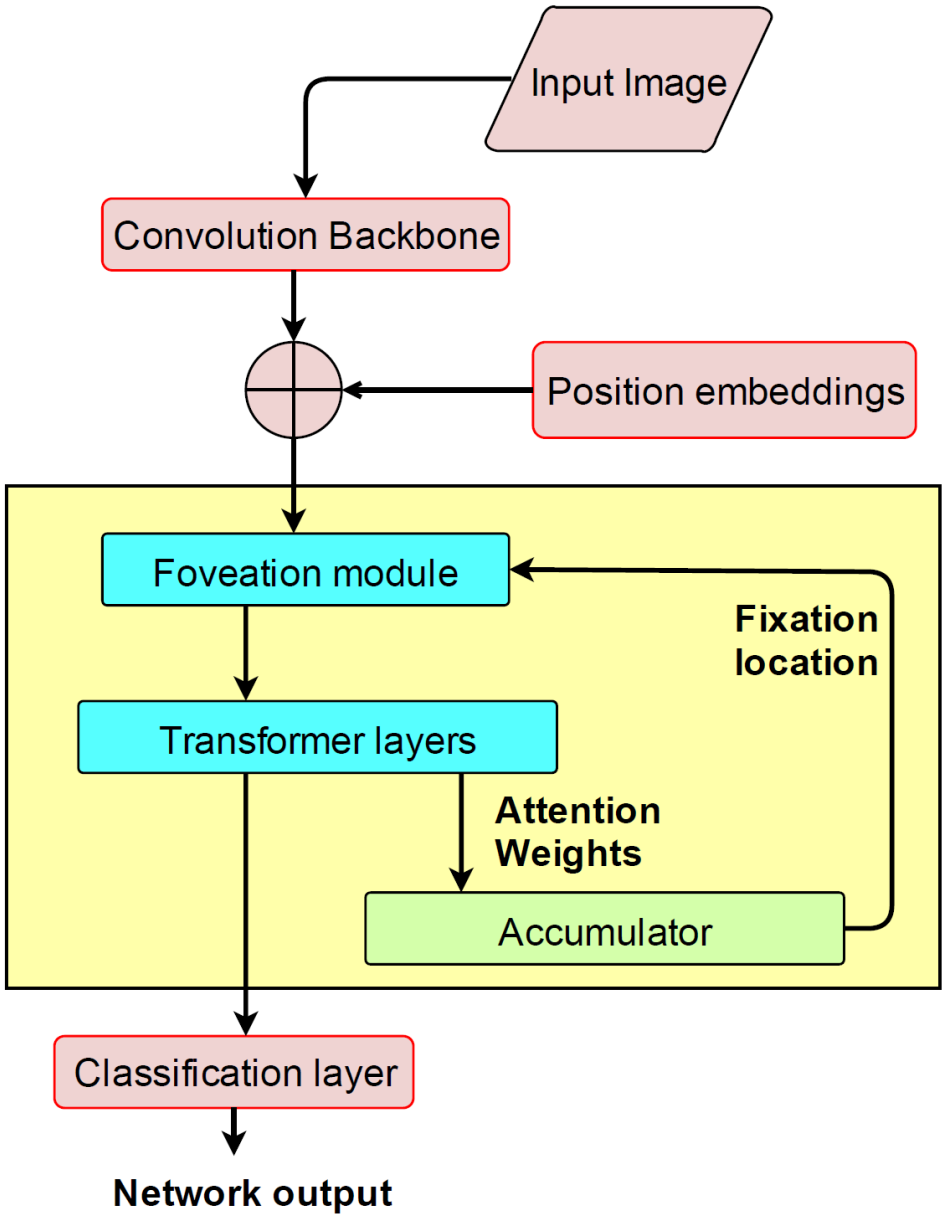

Figure 2.3: **FoveaTer architecture:** The foveation module performs fixation-dependent pooling. *Accumulator* uses the attention weights from the last transformer layer of past and present fixations to predict the next fixation location. Model blocks within the yellow region are executed for each fixation.

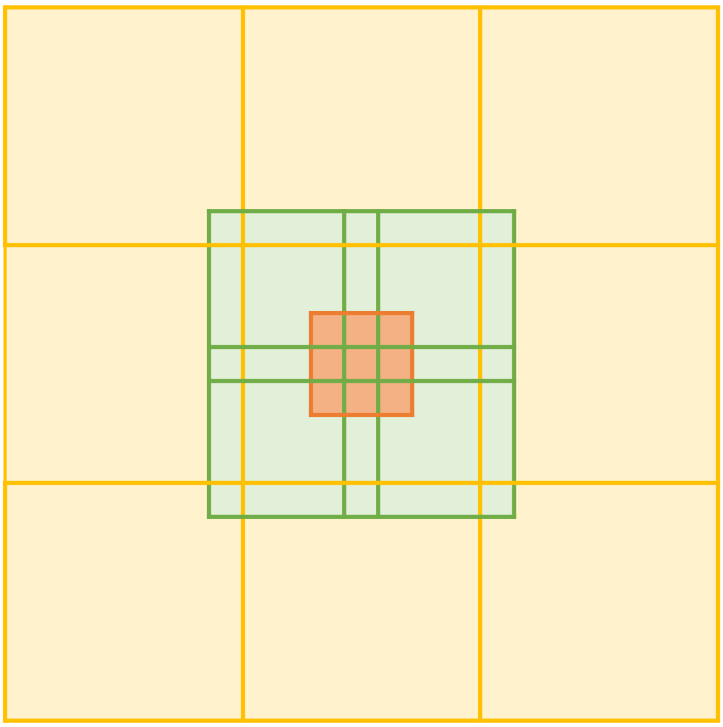

Figure 2.4: **Square pooling:** Input feature map is pooled to generate the pooled feature vectors. The Fovea is shown in red containing 9 feature vectors. The first level of pooling regions is of size 5×5 with stride 4 (green). The second pooling region level is size 7×7 with stride 7 (orange).

locations except the last and first row/column are potential fixation locations, resulting in 144 locations. We select a random location as the initial fixation point during training, and the model guides subsequent fixations.

Loss function: We use Cross-entropy for computing the classification loss. Loss from all fixations is incorporated to get the mini-batch loss, $loss = \sum_{i=1}^{N} L_{CE}(O_i, y)$ Where i corresponds to the fixation index, $N = 5$ for the Foveated model & $N = 1$ for the Baseline model, i.e., single-pass, y corresponds to the target label, O_i corresponds to the network output for fixation i, and L_{CE} correspond to cross-entropy loss.

2.3.1 Foveation module

The mean feature vector corresponding to each pooling region is computed using $P = (1/M) \sum_{j=0}^{M-1} E_j$, Where E_j is a feature vector belonging to that pooling region, and M is the number of feature vectors in that pooling region.

We use square pooling regions for computational speed-up. Each image in a mini-batch has a corresponding fixation location. The fixation location represents the center of the visual field, allowing us to align the input image/feature map with the visual field. After aligning the input feature map with the visual field, features falling within a pooling region are average-pooled, and the resultant pooled vector represents that pooling region. We use pooling regions with receptive field sizes 1×1, 5×5, 7×7 blocks on a feature map of size 14×14 blocks, as shown in Figure 2.4. Each block corresponds to a $[16, 16]$ pixel region in the input image of width and height 224. Central 3×3 red block represents the high-resolution Fovea, where there is no average pooling. The next ring of pooling regions, where the pooling region is green, has a receptive field of 5×5 which translates to an average pooling of 25 feature vectors to generate the representative feature vector for that pooling region. Similarly, the rings of orange-colored pooling centers have receptive

fields of 7×7.

2.3.2 Accumulator

Accumulator uses the self-attention weights from the last transformer layer for fixation guidance. Using the attention weights of the current and past fixations along with inhibition of return, the Accumulator (see below) predicts the subsequent fixation location. A confidence map (CM_N) is constructed based on the fixation point location by putting these weights back on a 14×14 map at the corresponding pooling region's location, where 14×14 corresponds to the size of the input feature map. Inhibition of return (IoR) [89] refers to a tendency in human observers not to attend to previously attended or fixated regions. Old accumulated attention map (AM_{N-1}) is weighted by 0.5 and added to the current confidence map to create the new accumulated attention map: $AM_N = 0.5 * AM_{N-1} + CM_N$. The inhibition of return (IOR_N) map is initialized with zeros and is the same size as the feature map. Locations corresponding to the current fovea location are changed to 16. After subtracting the IOR map from the accumulated attention map, max location of the resultant map is used as the next fixation location $Fix_{N+1} = \arg\max(AM_N - IOR_N)$.

2.3.3 Dynamic-stop of Fixation Exploration:

Due to various factors such as occlusion, camera angle, and brightness, the difficulty of making a classification decision varies across object classes and images. To achieve higher computational efficiency in our Foveated model during inference, we stop exploring the images with fixations when the predicted class with the highest probability reaches a pre-defined threshold corresponding to that class. We compute the threshold from the training dataset's set of all the correct prediction probabilities. The model stops

Table 2.1: **Ablation Studies:** Four network components are considered, and the percentage accuracy drop after five fixations with respect to the Benchmark model is reported in the last row. Checkmark (✓) indicates that the model includes the component, while the dashed-line (—) indicates that the component has been removed.

Network component	**Benchmark**	**Study 1**	**Study 2**	**Study 3**	**Study 4**
Foveation	✓	✓	✓	✓	✓
Peripheral features	✓	—	✓	✓	✓
Foveal features	✓	✓	—	✓	✓
Retention of foveal features	✓	✓	—	—	✓
Inhibition of return	✓	✓	✓	✓	—
Accuracy@1	76.29	62.85	72.60	75.29	75.23
Percentage drop		17.6	4.8	1.3	1.4

if the top prediction is above the 50th percentile of probabilities for that class and the second-best prediction is below the 5th percentile for that respective class.

2.4 Ablation Studies

We study the contribution of different network components to model performance in five ablation studies. The model was trained on ImageNet for 300 epochs and fine-tuned for 30 epochs for each ablation study. The first two studies asses the importance of peripheral and foveal features. Studies three and four assess the importance of memory provided by past foveal features and IoR, respectively. Lastly, we look at the contributions of the fixation guidance mechanism. Results are shown in Table 2.1 and Figure 2.5.

Study 1: Contribution of peripheral features: Peripheral features are essential because they contribute to image classification and help decide the subsequent fixation location. There is a sharp 17.6% drop in the network performance by removing the peripheral features.

Study 2: Contribution of foveal features: Foveal features provide high-resolution information. By removing access to foveal features of current and past fix-

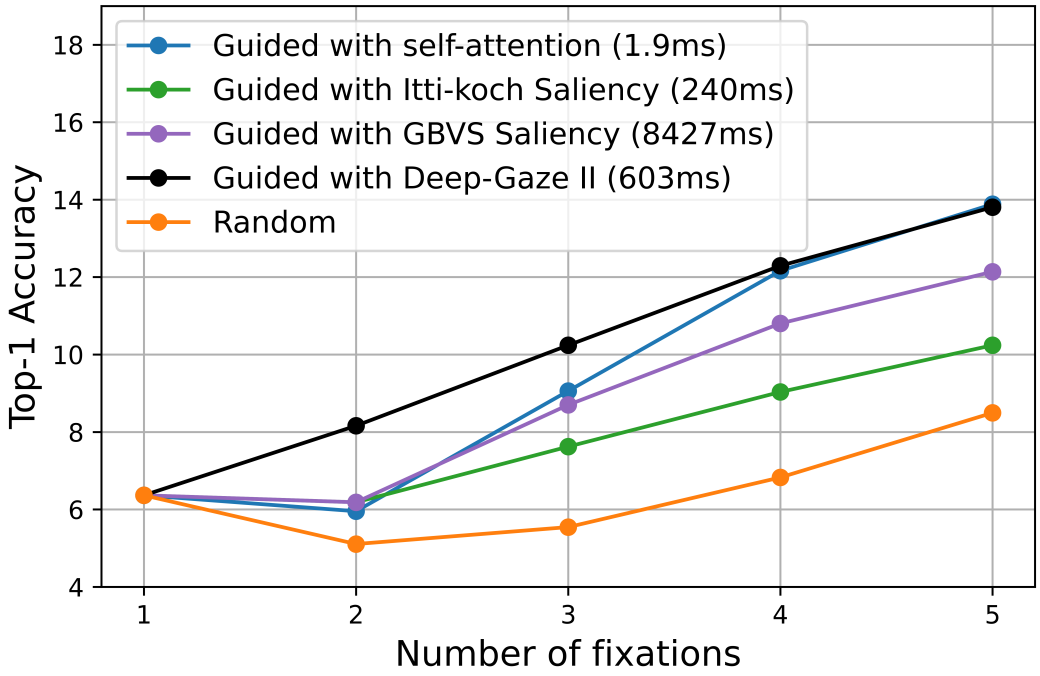

Figure 2.5: **Study 5: Fixation guidance mechanisms:** Self-attention guidance outperforms the random fixations by 63%. Initial fixation at the top-left corner of the image. Baseline algorithms use all image regions at high resolution to generate the heatmap. DeepGaze initially outperforms the self-attention-based guidance, but no difference is noticed after three fixations. The time taken to compute five fixations is shown in brackets. Self-attention is the fastest as no external algorithm is needed to compute additional heatmap.

ations, the model loses access to all full-resolution information. There is a 4.8% drop in the network performance by removing the foveal features.

Study 3: Retention of the foveal features: We incorporate memory by retaining the past foveal features and processing them along with the foveal and peripheral features of the current fixation. Even without this network component, the network has some memory as the model makes fixations to more informative locations using guided fixations. In this experiment, we remove the usage of foveal features from past fixations, and as a result, the model performance drops by 1.3%.

Study 4: Contribution of Inhibition of Return: By limiting the model's ability to revisit the fixation locations of the past, we force the model to explore rather than get stuck at one location. We only see a slight drop in performance of 1.4% without the IoR, signifying that the model can operate well without IoR. The results suggest that the model can learn not to revisit locations without explicitly implementing IoR.

Study 5: Effectiveness of Fixation guidance mechanism: Objects in the ImageNet dataset often occupy a large part of the image. As a result, image classification might be possible by fixating anywhere on a large percentage of the image. The importance of guided fixation is best illustrated when a few image regions are informative. To identify that subset of images, we separate the testing images into two groups, one with moderate difficulty and the other with too few or too many informative locations. To identify these two groups of images, we run our model under a one-fixation condition at each possible fixation location and calculate the percentage of locations (PoL) with the correct classification in that image. We use this as the metric for image difficulty, i.e., higher PoL signifies less difficulty and vice versa. As there are 144 locations, PoL ranges from 0 to 144. We label all the images where the PoL is more than one-eighth the maximum value, i.e., greater than 18, as too easy. Similarly, images with a PoL of zero are labeled as too difficult. After removing the images labeled as too easy or too

difficult, approximately 8% images fall in the middle, i.e., moderately difficult category. Figure 2.5 shows the comparison of random and guided fixations on this subset of images, and guided fixations have approximately 63% improvement over random fixations. We also compare the fixation guidance using the Itti-Koch, Graph Based Visual Saliency [20] and the DeepGaze-II model, where they take the image without foveation as input. Fixations guided by self-attention outperform the fixations guided by the Itti-Koch model and are as effective as those guided by the Deep-Gaze II for later fixations. Lower performance than Deep-Gaze II in the first fixations is not surprising since Deep-Gaze II is predicting the most likely regions to be fixated by humans (not the order).

Comparing the time taken for fixation prediction, our fixation guidance is the fastest as we leverage the model's internal attention weights rather than running a separate model. Time taken for computing five fixations - Guided by self-attention (1.9ms) $<$ Itti-Koch (240ms) $<$ Deep-Gaze (603ms) $<$ GBVS (8427ms). Sample image fixations are shown in Figure 2.6.

2.5 Accuracy and Robustness on ImageNet

In the following sub-sections, we compare the performance, computational complexity, and adversarial robustness of the Foveated model against the Baseline. The Foveated model is trained for five fixations, although it can work with any desired number of fixations at test time. Baseline and Foveated models have the same 24M parameters.

We use the Patchconvnet [90] architecture. We initialize the convolution backbone with the weights from ResNet-18 [91] model, and the transformer layers are initialized with the weights of the DeiT-small [59] model and trained for 300 epochs with an initial learning rate of $5e-4$ and a minimum learning rate of $1e-5$. We use AdamW [92, 93] optimizer with a decay of $1e-8$ and a cosine learning rate schedule. We use ImageNet [63]

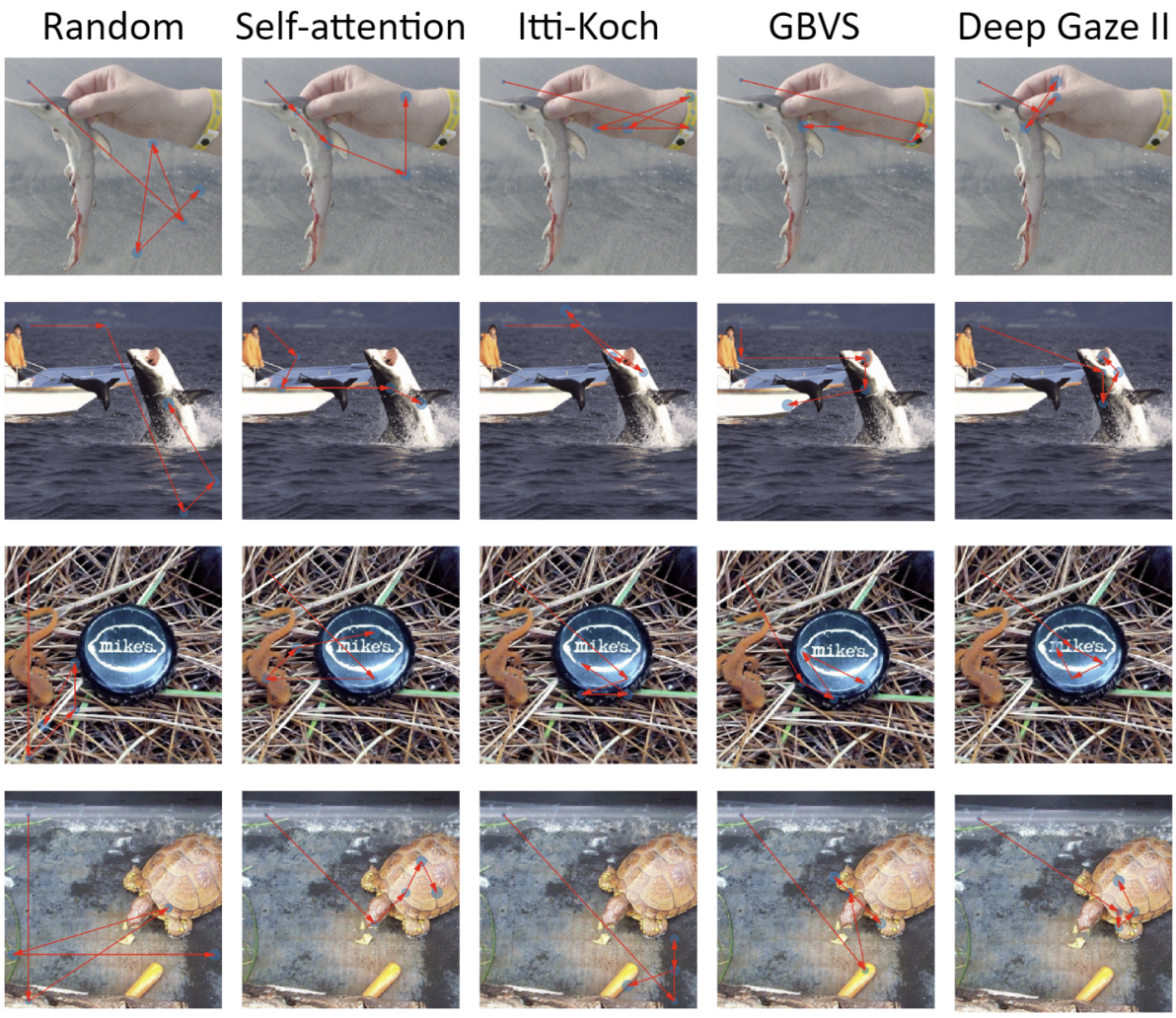

Figure 2.6: **Fixation guidance mechanism:** Fixation guidance by different models. Each row corresponds to a different image. **Random:** Fixations locations are selected at random. **Self-attention:** Fixations guided by the self-attention of the last transformer layers. **Itt-Koch, GBVS, DeepGaze II:** Fixations are guided by the top-locations of the heatmaps generated by these baseline algorithms.

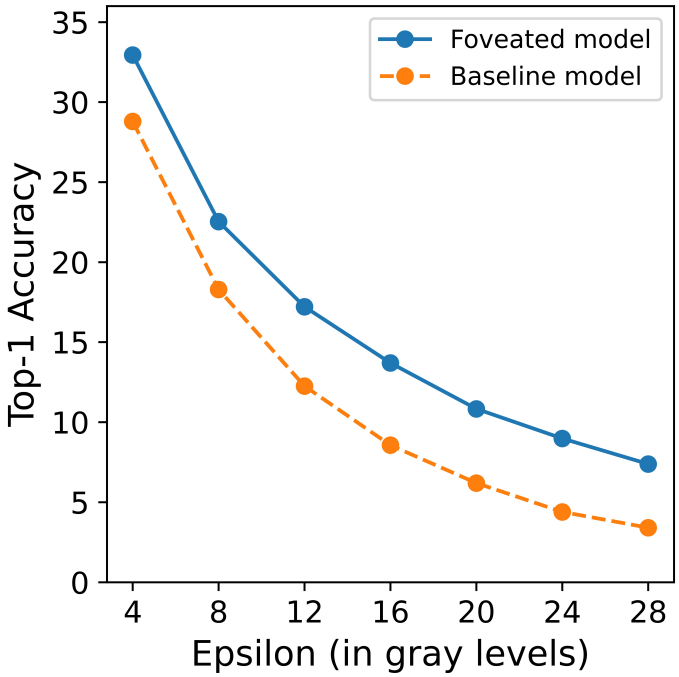

Figure 2.7: **PGD attack:** The strength of the attack is represented in terms of equivalent gray levels. Higher Epsilon results in a more potent attack and, as a result, in lower accuracy of the model. The foveated model outperforms the full-resolution baseline model.

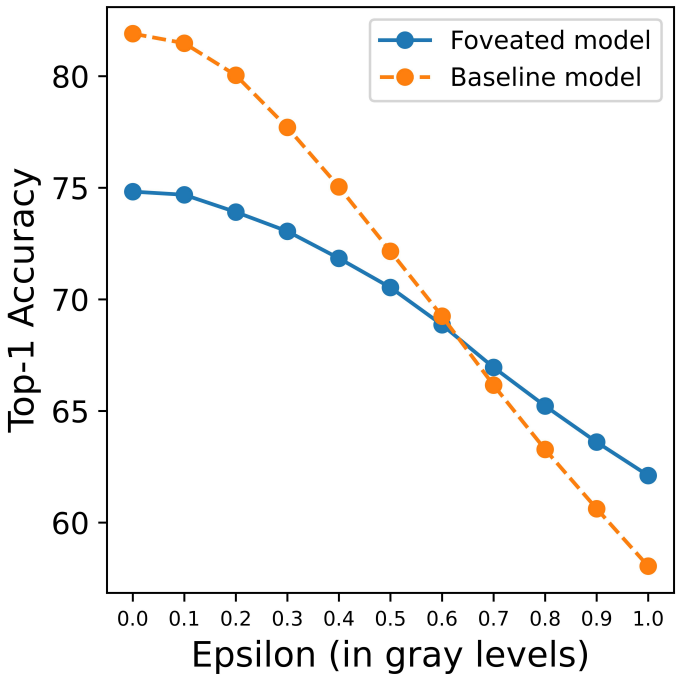

Figure 2.8: **PGD attack:** The foveated and full-resolution models have the same Top-1 accuracy at approximately 0.6 gray levels. As the rate of decrease in Top-1 accuracy of the full-resolution model is higher than that for the foveated model, foveated model outperforms the full-resolution model in terms of adversarial robustness after this threshold.

Model	Pooling type	Fixations	Type	Throughput	Acc@1
DeiT-Small				1699	79.83
Baseline				1229	81.90
			Uniform pool	2506	70.90
Foveated	Square	Rand-1		3820	69.80
		CF-1		3820	72.80
		CF-2		2307	74.70
		CF-3		1506	75.40
		CF-5		923	76.30
		CF-3	Dynamic Stop	2169	75.30
		CF-3	Ensemble	1236	81.30

Table 2.2: Throughput and Accuracy on ImageNet: We compare our models against the baseline model using Top-1 accuracy and Image throughput. (*Uniform pool* - uniform 5×5 pooling, *CF* - initial fixation at image center, *Rand* - random initial fixation)

dataset for the results shown in the following sub-sections. We use RTX A6000 GPUs for training and testing purposes. We report the number of inferences completed by the GPU during a one-second time interval to compare the computational complexity of different models during inference time.

2.5.1 Top-1 Accuracy:

For the Dynamic-stop, we first compute the throughputs of the Foveated model for each of the one to five fixation conditions, followed by the number of images belonging to each of those five fixation conditions. The throughput of the Dynamic-stop model is computed as the weighted Harmonic mean of the throughputs of individual fixation models. Ensemble refers to a model composed of both the Foveated and Baseline models. When the Dynamic-stop is applied, and the model cannot make a decision even after the maximum number of fixations, the Ensemble model transfers the responsibility of making a decision to the Baseline model.

We present the results on the ImageNet dataset in Table 2.2. The Deit-Small model has a throughput of 1699 and Top-1 accuracy of 79.83. The Baseline, which has the same architecture as the Foveated model except for the foveation module, has a throughput of 1229 and an accuracy of 81.90. Since the first level of the pooling region is of size 5×5, we construct a pooled version of the baseline model using 5×5 average-pooling. We compare this with the Foveated model with two fixations, with approximately the same throughput. The Foveated model with two fixations outperforms the uniformly pooled Baseline model, as shown in row 6. Dynamic-stop and Ensemble performances are shown in the last two rows. The performance of the ensemble model reaches close to the Baseline model in terms of throughput and accuracy.

2.5.2 Robustness against adversarial attacks

We consider the Projected Gradient Descent (PGD) attack to compare the robustness of Foveated and Baseline models. PGD uses ten iterations with a step-size of $\epsilon/5$ and l-infinity norm. We use Cleverhans library [94] for implementing the adversarial attacks. Figure 2.7 shows the model accuracy after attacking the input image with the adversarial attack. The crossover point between the foveated and the full-resolution models is shown in Figure 2.8. Epsilon (ϵ) represents the strength of the attack. Foveated model displays strong defense as compared to the Baseline model. Foveated model consistently outperforms the Baseline model. A comparison with existing models, showing the robustness of the foveated systems against adversarial attacks, is demonstrated in Appendix A.3.

2.5.3 Visualization

Network output is visualized for one, two, and three fixations are visualized in Figure 2.9, Figure 2.10 and Figure 2.11 respectively.

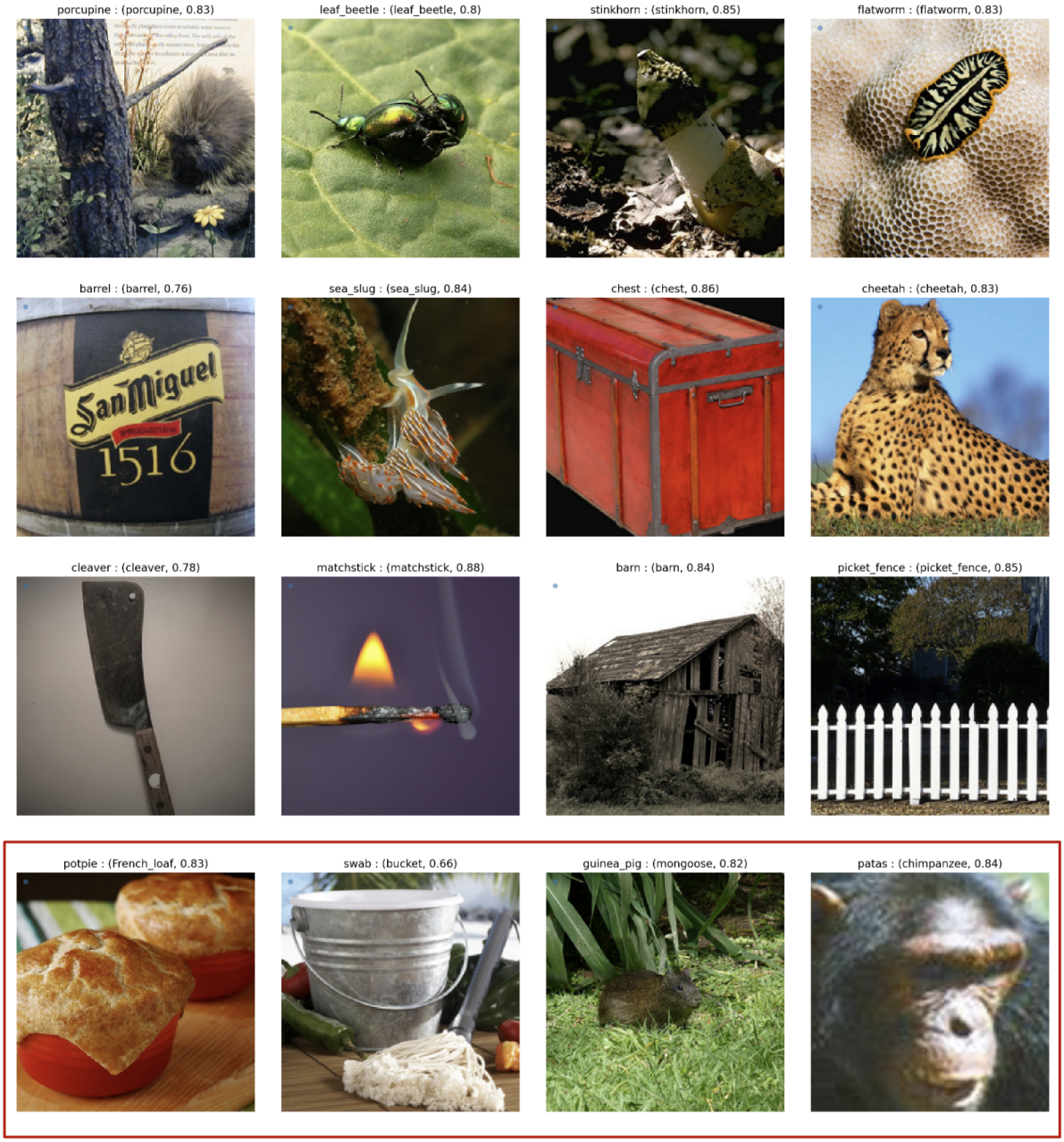

Figure 2.9: **Visualization:** Sample images in which the network needs **one** fixation with dynamic stop. The first three rows show examples of where the network made the right decision, and the last row shows examples of incorrect predictions.

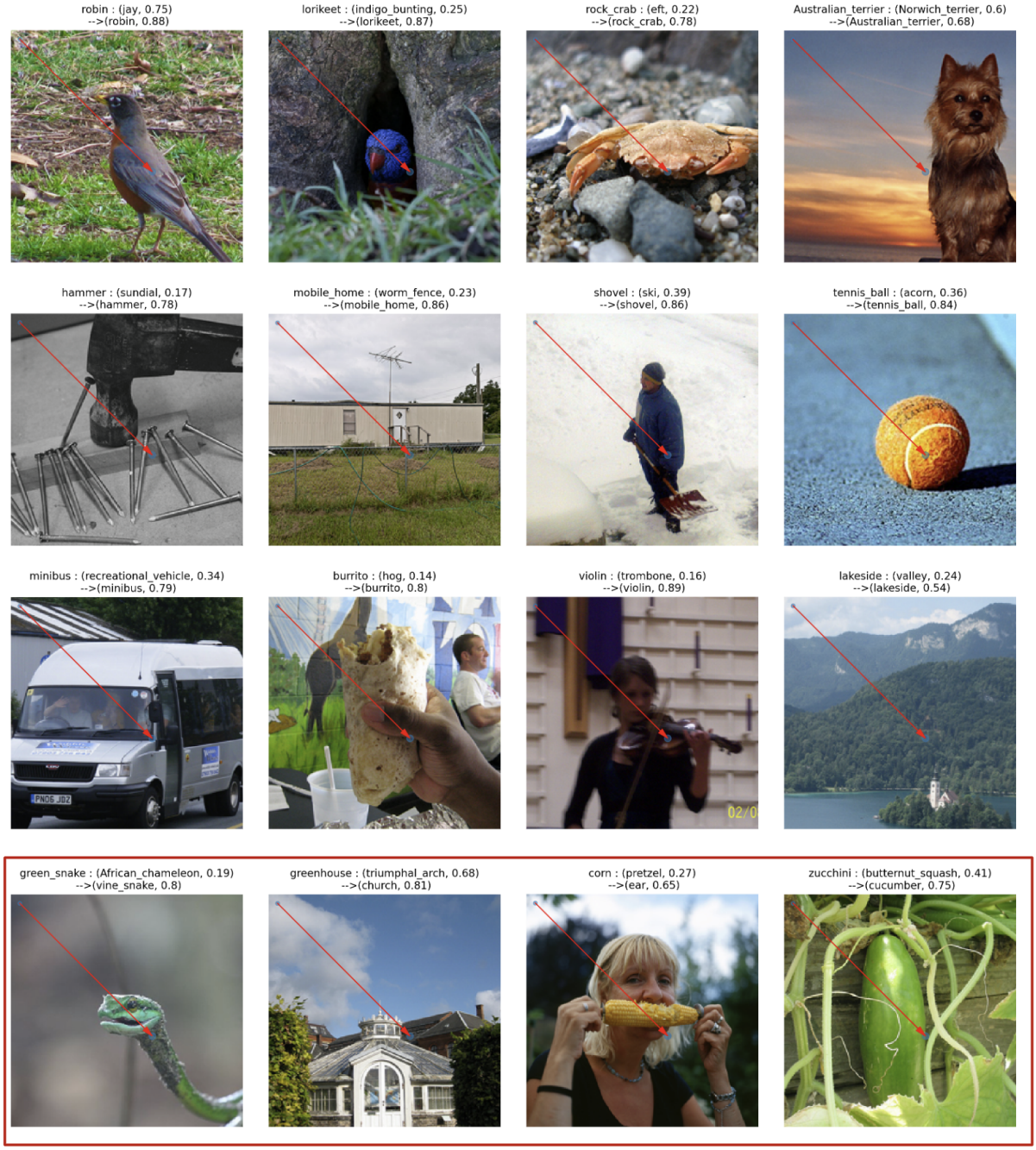

Figure 2.10: **Visualization:** Sample images in which the network needs two fixations with dynamic stop. The first three rows show examples of where the network made the right decision, and the last row shows examples of incorrect predictions.

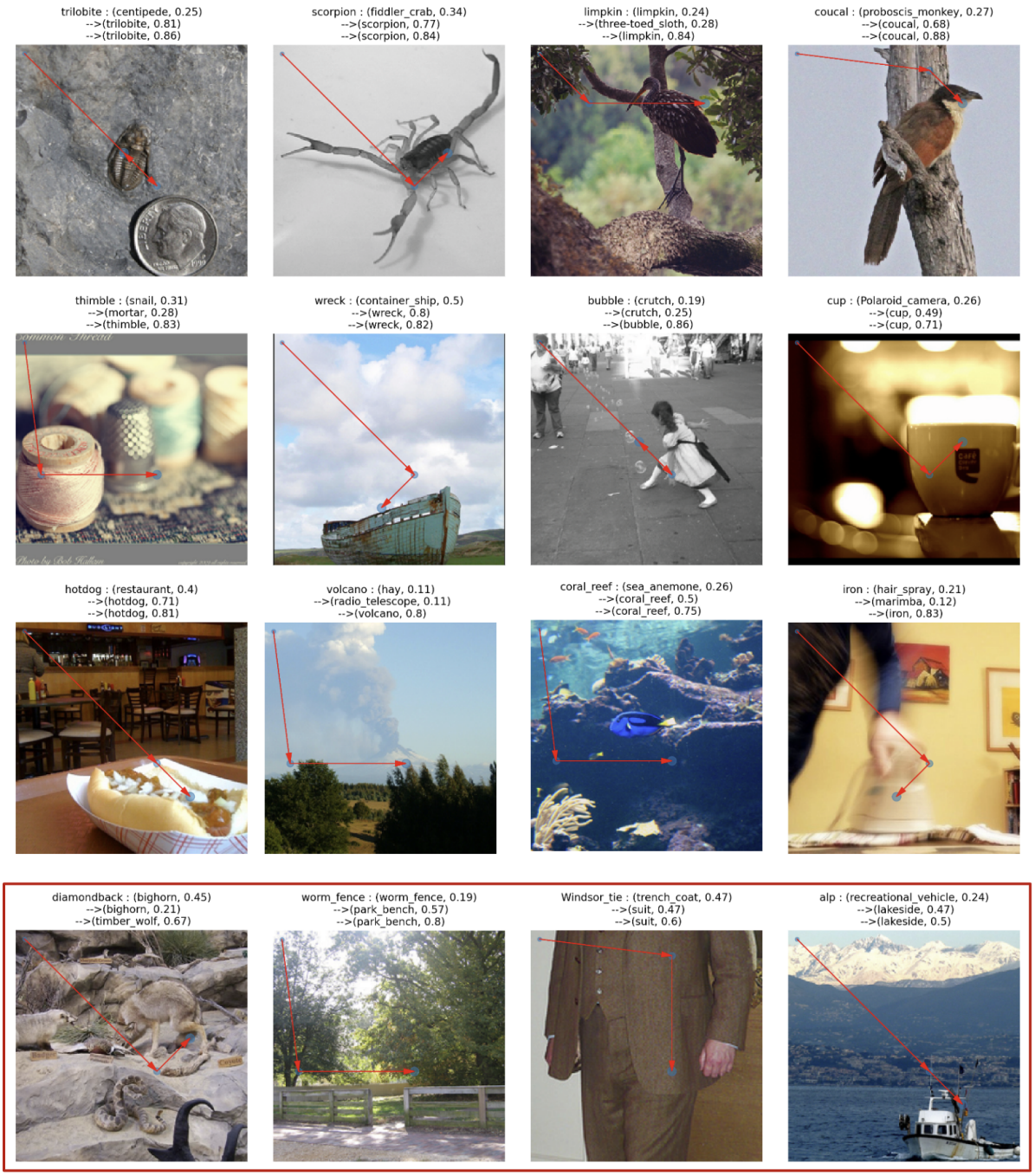

Figure 2.11: **Visualization:** Sample images in which the network needs **three** fixations with dynamic stop. The first three rows show examples of where the network made the right decision, and the last row shows examples of incorrect predictions.

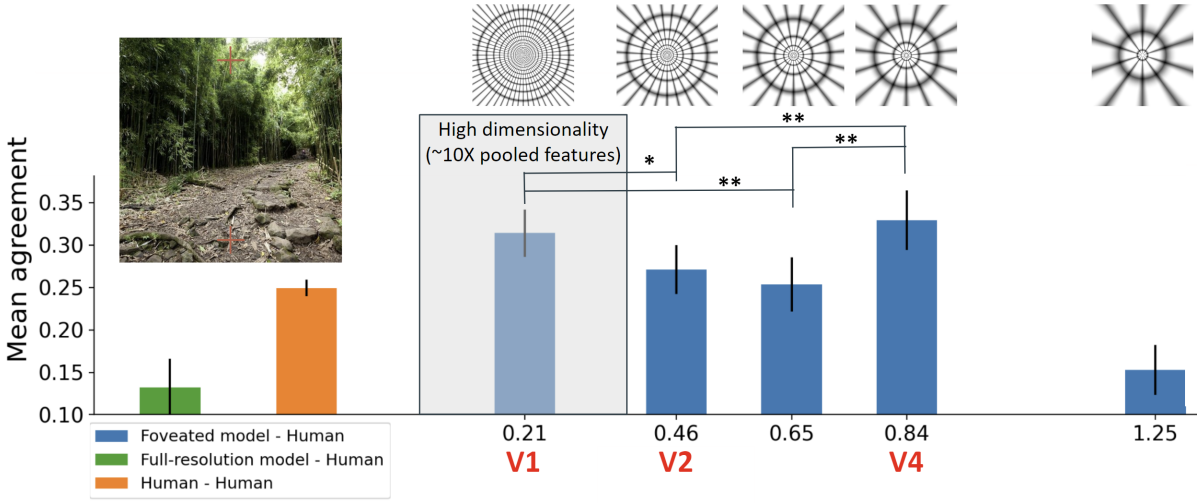

Figure 2.12: **Scene Classification:** Mean agreement values of the Baseline and the Foveated models with human decisions (correct/incorrect). Error bars refer to the standard error across 22 participants. Paired t-test p values indicate statistical significant agreement differences across scales, $^{**}p < 0.01$, $^{*}p < 0.05$.

2.6 Biologically plausible FoveaTer

Radial-Polar pooling makes the model more biologically plausible. Through psychophysics experiments of image discrimination and modeling, [79] showed that different layers of the visual cortex correspond to different *scales* where the *scale* parameter determines how many radial and polar pooling regions are present in that configuration. We use this model to predict human decisions in a scene classification psychophysics task while maintaining fixation and calibrate the scaling parameters of the pooling regions of FoveaTer. Figure 2.12 shows examples of various configurations.

2.6.1 Calibration of radial-polar pooling regions

We used thirty scene categories from the places365 dataset [95] to create the experiment dataset. Scene categories are listed in Appendix A.2. The task was to classify each image into one of the 30 categories. Sixty images were presented, with each image

subtending 22.7 x 22.7 degrees visual angle, and observers fixated at the bottom-center or top-center within the images (2.2 degrees from the top or bottom edges of the image, Figure 2.12). Real-time infra-red video eye tracking allowed for interruption of the displayed image when observers made an eye movement.

We tweaked the last convolutional layer so that the convolution backbone of the model outputs a $56 \times 56 \times 384$ feature map instead of a $14 \times 14 \times 384$ feature map, thus allowing us to apply the pooling regions on a higher resolution feature map. We train multiple models with different *scale* values for spatial pooling. For each *scale*, Foveated model is trained for 60 epochs after initializing the weights with the square-pooled Foveated model trained on ImageNet.

Error consistency metric [96] produces the normalized decision agreement between two observers, where the normalization is a function of the accuracy of both observers. We computed the mean agreement between the human decisions and the Foveated model for a set of *scales* as shown in Figure 2.12. We also computed the mean agreement between human decisions and the Baseline model (independent of *scale*). For *scales* corresponding to V2 (*scale*-0.46) and V4 (*scale*-0.84) layers of the visual cortex, we observe a significant difference between the mean agreement of humans with Foveated and Baseline models. Although the accuracy of the Baseline model (0.93) is higher than the Foveated model (0.86), human decisions with a mean accuracy of 0.83 are in better agreement with the Foveated model. The fixation at the top-center or the bottom-center limited the image information accessible to the human observers, which the Full-resolution fails to model. Our findings suggest that human categorization of scenes within a single fixation can be better predicted with FoveaTer with pooling regions that scale according to properties of the visual cortex (V1 and V4).

Response	# Humans
desert-sand	18
beach	3
cliff	1
	22

Figure 2.13: **Scene classification example 1:** Scene category of the image is beach. Human subjects and the foveated model fixated at the bottom center of the image. **Left:** Eighteen of the twenty-two subject judged the scene to be desert sand. **Middle:** Foveated model classified the scene as desert-sand. **Right:** Full-resolution model, which has access to all regions of the image without any degradation in detail, classified the image correctly as beach.

2.6.2 Visualization

To understand why the Foveated model's decisions are similar to the humans, two examples are shown in Figure 2.13 and Figure 2.14. They provide the intuition on why the foveated model makes decisions similar to humans.

2.6.3 Accuracy and robustness on ImageNet

We evaluated FoveaTer's accuracy and robustness using pooling parameters (scaling 0.84, V4) that predicted human scene classification decisions and were computationally efficient (relative to V1). Results are shown in Table 2.3. The throughput of the Foveated model with radial-polar pooling regions is very low due to the lack of library functions implementing the radial-polar pooling. As the specialized hardware performing neuro-foveal pooling becomes available in the future, throughput gaps will disappear, and the Foveated model will become competitive with the Baseline models. Adversar-

Response	# Humans
Glacier	11
Canyon	6
cliff	5
	22

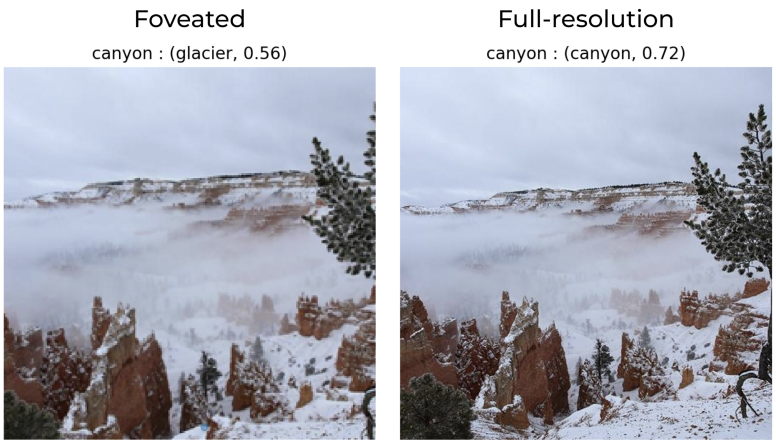

Figure 2.14: **Scene classification example 2:** Scene category of the image is canyon. Human subjects and the foveated model fixated at the bottom center of the image. **Left:** Eleven of the twenty-two subject judged the scene to be glacier. **Middle:** Foveated model classified the scene as glacier. **Right:** Full-resolution model, which has access to all regions of the image without any degradation in detail, classified the image correctly as canyon.

ial robustness of Foveated model against PGD attack with radial-polar pooling regions is illustrated in Figure 2.15. As with the square pooling regions, Foveated model with radial-polar pooling regions is also more adversarial robust than the Baseline model.

2.7 Conclusion

We provided a comprehensive framework for using foveal processing and fixation exploration on a Vision Transformer architecture for image classification. The proposed architecture introduces a way to limit computations required to process an image by flexibly adjusting the required number of fixations, providing robustness to adversarial attacks, and giving us a model that can allocate computational resources based on the difficulty of an image. Our ablation studies highlight the importance of peripheral processed features, how the self-attention guiding eye movements learn to inhibit revisits and results in accuracy similar to a model guided by predictions of human fixation

Model	Type	Throughput	Acc@1
Baseline		1229	81.90
Foveated (square)	Dynamic Stop	1506	75.40
	Ensemble	2169	75.30
		1236	81.30
Foveated (Radial-Polar)	Dynamic Stop	117	76.69
	Ensemble	198	76.65
		186	81.52

Table 2.3: **Throughput and Accuracy** on ImageNet using radial-polar pooling regions with *Scale* 0.84: All foveated models made three fixations. (*CF* - initial fixation at image center)

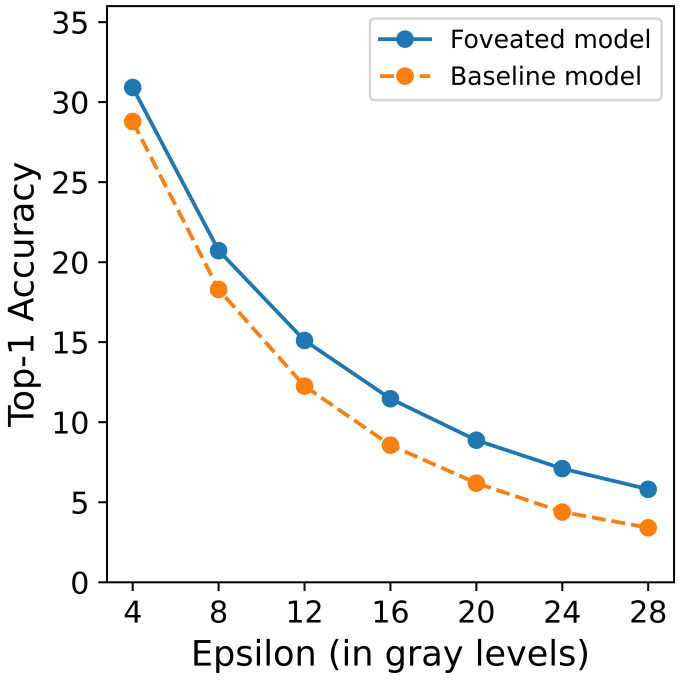

Figure 2.15: **PGD attack:** PGD adversarial attack on Foveated model with radial-polar pooling regions. Foveated model outperforms the baseline full-resolution model.

(DeepGaze). We also implemented a more biologically plausible implementation with radial polar pooling and showed that pooling parameters corresponding to visual cortical areas V1 and V4 could explain human scene categorization decisions better than the Baseline non-foveated model. In conclusion, we leveraged the most recent Vision Transformer architecture and combined it with ideas from foveated vision to come up with a model which has multiple knobs in terms of the number of fixations to be executed and limits on the computations performed so that the end-user will have the flexibility to fine-tune depending on their needs.

Chapter 3

Foveated Transformer for Object Search (FST)

3.1 Introduction

A large literature of study shows that scene context and object relationships influence human search decisions, eye movements, and neural activity. Humans direct their eyes to expected locations where objects might be found. They often miss a search object appearing at an unexpected location or unusual size relative to the surrounding objects. Some models of eye movements or visual search capture these landmark human behaviors but often incorporate a weighting "prior" map. In most instances, these prior maps are estimated separately from analyzing a large dataset of natural scenes. How to create models that automatically learn these contextual relationships has been challenging. Convolutional neural networks do not show the same degree of learning of contextual relations as humans. A fundamental property of visual Transformer models, which were first developed for natural language processing, is that they implement computations across different regions of the image (a stage referred to as multi-head attention) and can

thus learn relationships of visual features across large spatial extent. Unlike convolutional neural networks, transformer models can learn scene context, and their object detector probabilities are influenced by context. Visual Transformers typically directly operate on images rather than feature maps, which is less biologically plausible. We adopted an architecture that will process an input image with convolutional layers from ResNet-18 (Figure 3.10) that mimics early vision. We will use the Transformer model framework DeiT which is pre-trained to classify images. There are pre-23M pre-trained parameters on ImageNet. We only train the weights of the last CNN layer (59K) from scratch. After inserting the foveation architecture, we fine-tune 7.7M Transformer parameters on the MSCoco data set, which contains smaller objects more relevant to visual search.

Another vital component of the FST model is the fixations programmed to gather information to support the perceptual task, the foveated visual system, and the statistical properties of the images. Computational models of human vision have approached this problem using a simple Gaussian-filtered noise background for which the statistical properties (probability density functions) are known. In Najemnik & Geisler's ideal searcher (with assumptions of statistical independence), one can calculate the expected performance of a future fixation by numerical evaluation. However, the ideal searcher approach does not apply to real-world images where simple probability density functions do not specify the statistics. A second approach to finding performance-maximizing fixations come from the machine learning literature by using reinforcement learning (RL) which has been applied to find an optimal sequence of actions. We use Visual Transformers' attention maps (in the class token layer, Figure 3.10) to guide the fixations. The attention map indicates the contribution of various feature regions (mapped to image coordinates) towards the classification decision, updated after each fixation. An interpretation of this strategy is making saccades to the most informative feature for the decision (closer to the maximum posteriori probability strategy proposed for human eye movements with

synthetic images with noise). The results show how the FST model with this attention-map guided eye movements results in fixations guided by context directed to the right of the keyboard (irrespective of starting fixation position or placement of monitor and keyboard in the image!). We evaluated the impact of using different saliency algorithms for the task of fixation guidance and their resultant impact on decision accuracy.

We evaluated how the proposed algorithm, fixation guidance using Transformer attention maps, compares to various baseline algorithms inserted into the Transformer model. The baseline algorithms include random fixations, bottom-up saliency fixations, and Deep Gaze fixations. The baseline algorithms are tested with and without inhibition of return to avoid revisits. We showed that our task-relevant fixations improve perceptual performance more rapidly than the baseline algorithms. We also assessed whether the FST model correctly predicts some of the primary influences of scene context on human performance. Searched objects are more easily detected at expected vs. unexpected locations. Searched objects are harder to detect at unusual relative sizes than background objects.

3.2 Psychophysics experiments

We performed two psychophysics experiments for the visual search of the computer mouse. In the first experiment, in a forced-fixation experiment, the participants fixated on a pre-defined location on the screen. The computer mouse is presented at different distances (from the fixation location) in the visual periphery. This experiment helped us capture the decay rate in human detection performance as a function of eccentricity for the task of computer mouse detection. In the second experiment, the participants are allowed to make the eye-movements. The computer mouse is presented at expected or unexpected locations and at expected or unexpected scales. This experiment helped us

capture the contextual effects of spacial location and scale. In the following sections, we describe the experimental setups in detail.

3.2.1 Dataset for the experiments

We created the dataset for the experiment under a controlled lab environment. The monitor and keyboard were present in all images, whereas the computer mouse was present in some images and absent in others. The mouse was located either in the expected or unexpected spatial location, where the expected location (in-context) refers to the location where it is usually expected based on the natural images, and the unexpected location (out-of-context) refers to the rest of the image locations (Figure 3.1). We also created images where the mouse is in unexpected scale, i.e., four times bigger than the expected scale per its surrounding objects (Figure 3.2). A cell phone which looks like the computer mouse in the visual periphery, has been used as a decoy. We created two additional images using the decoy for each of the mouse in-context/out-of-context location images - 1. By placing the decoy in the out-of-context/in-context location while the mouse is in-context/out-of-context location and 2. By placing the decoy in the out-of-context/in-context location when the mouse is absent. The cell phone is used as the decoy, i.e., an object that looks like the mouse when presented in the visual periphery, and like for the case of the mouse, it is present either in the in-context or out-of-context locations where those locations are defined for the mouse and not the cell phone. The dataset contained eighteen scenes, i.e., images taken at different angles of monitor-keyboard configurations. Each scene had three distractor conditions containing four, eight, or fourteen distractors, i.e., the number of objects other than a monitor, keyboard, and mouse on the desk (Figure 3.3). As a result, the dataset comprised 378 images, for which we have shown the category-wise break-up in Table 3.1, and sample

Figure 3.1: **Spatial location:** Spacial location of the mouse according to the natural image statistics: **Left:** Mouse is in the expected location, **Right:** Mouse is in an unexpected location.

Figure 3.2: **Scale:** Scale of the mouse with respect to the surrounding objects according to the natural image statistics: **Left:** Mouse is in the expected scale, **Right:** Mouse is in an unexpected scale.

images for each category are shown in Figure 3.4.

3.2.2 Psychophysics 1: Mouse Eccentricity performance

This experiment aims to measure mouse detectability performance as a function of eccentricity, i.e., human detection performance when the computer mouse is presented in the visual periphery at different distances from the fovea/fixation location. We presented the computer mouse at six different visual eccentricity locations. Seven participants who were undergraduate students at the University of California, Santa Barbara, participated in the experiment. We compensated the participants with course credit and provided no

Figure 3.3: **Distractors:** Variation in the number of distractors: A scene containing, **Left:** Four distractors, **Middle:** Eight distractors and **Right:** Fourteen distractors

Mouse	Distractor	Image Count
In-Context	absent	54
In-Context	Out-of-Context	54
Out-of-Context	absent	54
Out-of-Context	In-Context	54
absent	absent	54
absent	In-Context	54
absent	Out-of-Context	54

Table 3.1: **Dataset for Psychophysics experiments:** The dataset was created under a controlled lab environment. Cell-phone, which looks like the mouse in the visual periphery, is chosen as the distractor. The dataset contained all combinations for the presence or absence of the objects, the computer mouse, and the distractor. The break-up of the number of images for each condition is shown.

monetary benefits.

Experiment Dataset

For the mouse-absent images, the fixation location is selected based on the mouse location in the corresponding mouse-present image. The location is selected from in-context mouse location 85% of the time. Since there are six eccentricity conditions, each image can give rise to six experiment trials. Thereby, 2268 experiment trials are created from the 378 images from the raw dataset. This dataset would have had mouse-present to mouse-absent images in a ratio of 4:3 and contained an equal number of in-context and out-of-context images. To maintain similarity to natural images where the in-context

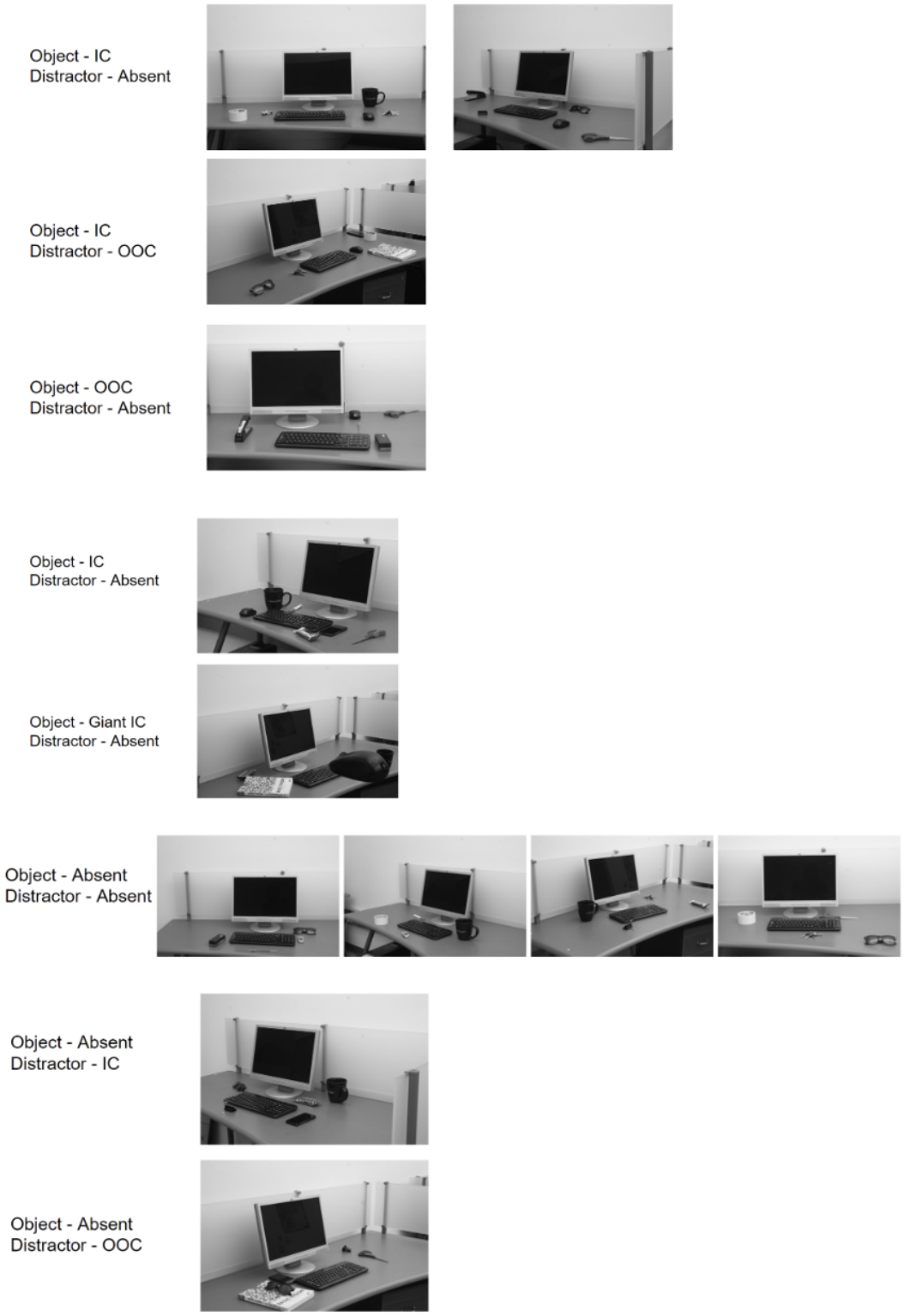

Figure 3.4: **Dataset for Psychophysics experiments:** Sample images for each of the eight conditions described in the Table 3.1.

mouse is much more highly probable than the out-of-context mouse location and to maintain a balance between the mouse-present and mouse-absent images, we sampled the 2268 trials accordingly. The resultant dataset contained 1321 trials and an approximately equal number of mouse present and absent images. Mouse is in the in-context location in about 92% of the mouse present images. The break-up of the final experiment trials is shown in Table 3.2. Each trial is shown twice, resulting in a total of 2642 trials.

Each participant's performance is shown in Figure 3.5. Participants displayed better performance with a smaller number of distractors in the image.

Mouse	Image Count
In-Context	564
Out-of-Context	49
Absent	658

Table 3.2: **Psychophysics 1:** Distribution of images for the experiment trials. An approximately equal number of mouse-present and mouse-absent images are used in the dataset. The mouse is present in the in-context location in approximately 92% of the mouse-present images.

3.2.3 Psychophysics 2: Visual search

The objective of the second psychophysics experiment is to capture the spatial and scale contextual effects in human subjects. The participants can make eye movements to search for the computer mouse. The computer mouse is presented at expected or unexpected locations and expected or unexpected scales. This experiment helped us capture the contextual effects of spacial location and scale. In the following sections, we describe the experimental setups in detail.

Experiment Dataset

The experiment dataset consisted of 102 valid images and 102 filler images. Sample images are shown in Figure 3.4.

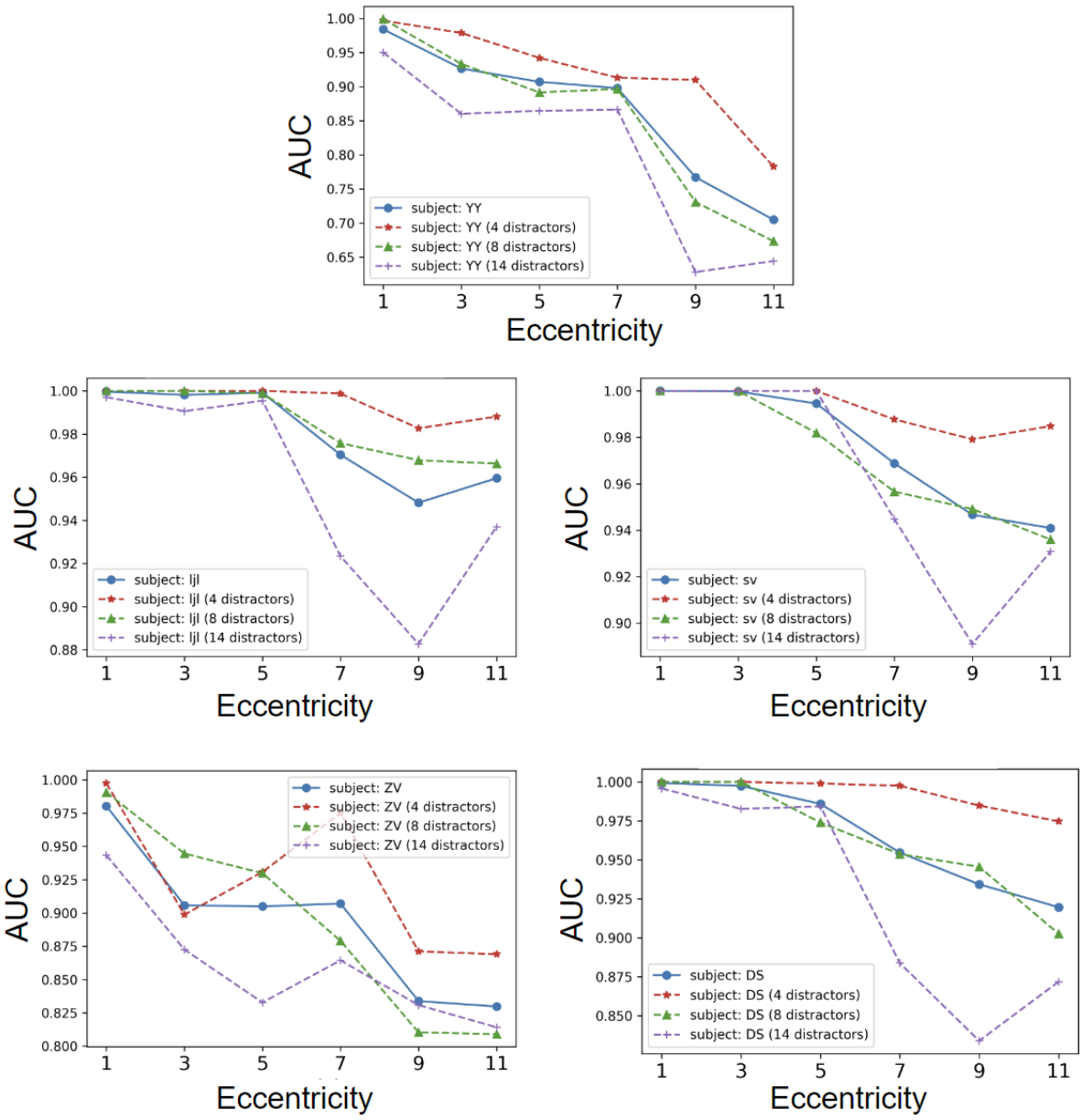

Figure 3.5: **Psychophysics 1:** Mouse eccentricity performance for each of the five subjects. It is also separated by the number of distractors, shown with the dotted lines in each plot.

There are three distractor conditions - four distractors, eight distractors, and fourteen distractors, each consisting of 34 images. There are three saccade conditions - one, two, and three saccade conditions, i.e., the display is interrupted after the participant made the pre-determined number of saccades for that image. Out of the 34 images for each distractor condition, 12 images belong to the one saccade condition, 12 images belong to the two saccade condition, and 10 images belong to the three saccade condition. Within each distractor-saccade condition, the breakup of different image conditions is shown in Table 3.3 for one/two saccade conditions and Table 3.4 for three saccade conditions. We only presented big-sized mouse under the one and two-saccade conditions. This was done not to let them familiarize themselves with the big-sized mouse in the three-saccade condition.

Mouse	Distractor	Image Count
In-Context	absent	2
In-Context	Out-of-Context	1
Out-of-Context	absent	1
Out-of-Context	In-Context	1
Big mouse In-Context	absent	1
absent	absent	4
absent	In-Context	1
absent	Out-of-Context	1

Table 3.3: **Psychophysics 2: Dataset: One/Two-saccade condition:** Number of images present in each mouse-distractor condition for the one and two-saccade conditions. The number of mouse-present images is ensured to be equal to the number of mouse-absent images.

Experiment details

The display monitor is of size 1024×1280 pixels. The image is of size 683×1024 pixels. The image is placed in the middle of the screen, i.e., 170 pixels from the top edge of the monitor and 128 pixels from the left edge, as shown in Figure 3.6. The distance of the human subjects to the image screen (d) is 95 centimeters. The monitor is of size

Mouse	Distractor	Image Count
In-Context	absent	2
In-Context	Out-of-Context	1
Out-of-Context	absent	1
Out-of-Context	In-Context	1
absent	absent	3
absent	In-Context	1
absent	Out-of-Context	1

Table 3.4: **Psychophysics 2: Dataset: Three-saccade condition:** Number of images present for each mouse-distractor condition for the three-saccade condition. Big-mouse is not present for the three-saccade condition, and the number of mouse-absent images is reduced by one.

30 centimeters in width and 38 centimeters in height. The visual angle subtended by the monitor is computed using,

$$Angle = 2 * arctan(z/2d) \tag{3.1}$$

where z is the width or height of the monitor.

As a result, the monitor subtended an angle of 18 degrees in height and 22.6 degrees in width. The monitor consists of 1024 pixels in height and 1280 pixels in width, so each pixel subtends a visual angle of 0.0176 degrees. And the displayed images, which are 1024 pixels in width and 683 pixels in height, subtend a visual angle of 12 degrees in height and 18 degrees in width.

The response screen consisted of a 1-10 scale where 1 corresponds to the mouse's absence, and 10 corresponds to the mouse's present condition.

Participants

Sixty-three participants participated in the experiment. These are undergraduate students at the University of California, Santa Barbara, compensated with course credit and are not provided any monetary benefits. Out of which 58 participants successfully

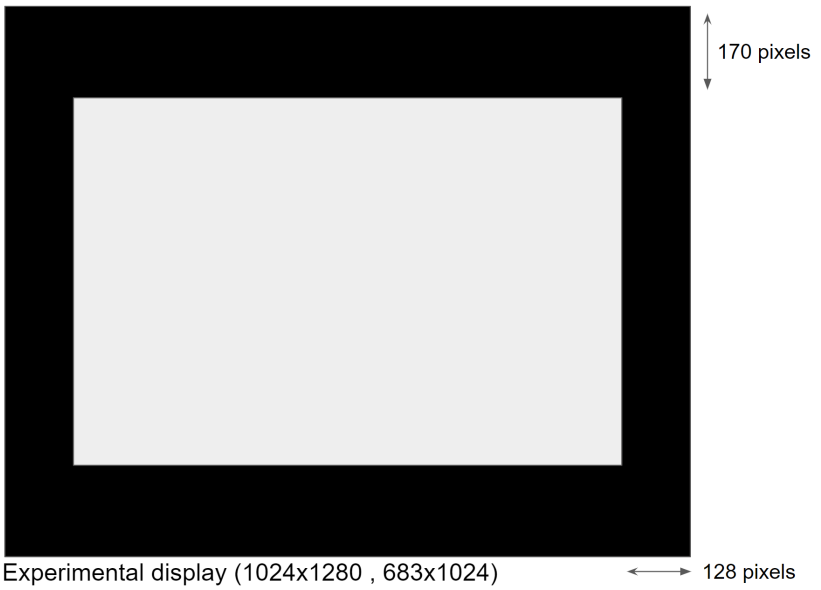

Figure 3.6: **Psychophysics 2:** Experiment screen for the visual search experiment. The screen is of size 1024 x 1280 with black pixels, and the image of size 683 x 1024 is placed at the center of the screen, shown with gray pixels for illustration.

finished the experiment. Each participant saw 102 images resulting in 5916 trials. In some trials, an additional fixation is observed in a small percentage of trials, i.e., the subject makes an additional fixation before the screen switches from the experiment screen to the response screen. As a result, 171 trials are removed from the 5916 trials, i.e., 3% trials.

3.3 Foveated Search Transformer Model (FST)

Foveated Image classification model (FoveaTer) trained on the ImageNet is finetuned for the Search task. Instead of deviating from the image classification and adopting a segmentation-style network, we continue working with the image-level classification to perform the visual search task. A transition to a segmentation-style network for object search is a natural extension, and we intend to explore it in future work.

Figure 3.7: **FST model: Training dataset:** Mouse-present images from the MSCOCO dataset are padded and used for model training.

3.3.1 Dataset

MSCOCO dataset is used to finetune the network for object search. Since the psychophysics counterpart is limited to the search of the computer mouse, we limit the model to performing a visual search to just the computer mouse (referred to as mouse hereafter) instead of extending it to all the eighty object categories present in the MSCOCO dataset. Since the visual search is limited to the mouse, only images containing the mouse are used for training. One pitfall of using an image-level classification instead of a pixel-level classification is that the network might mistakenly learn to recognize the contextual objects occurring along with the object of interest as the objects of interest. For example, the keyboard and monitor occurring along with the mouse might be learned as the objects of interest if the object-absent images do not have many images with the keyboard and monitor without the mouse. To counteract this effect, instead of using **any** mouse-absent images as the negative images, we construct the mouse-absent images from

Figure 3.8: **FST model: Training dataset:** Mouse instance from the mouse-present is erased using the pixels from the top-left corner and are used as the mouse-absent images for model training.

the mouse-present images by erasing the mouse using the pixels from the top-left corner of the image. The images are padded with gray pixels along width or height to make them square. Sample mouse-present and mouse-absent images are shown in Figure 3.7 and Figure 3.8, respectively. The training dataset consists of 1876 mouse present/absent images, and the validation set consists of 88 mouse present/absent images.

3.3.2 Radial-Polar Pooling

The loss of spatial detail away from the point of fixation occurs at many steps in the visual stream, starting with the photoreceptor density on the retina [97], the higher convergence of bipolar cells [98] to ganglion cells in the retinal periphery, cortical magnification in primary cortex V1 [99], V2, V4, and Superior Colliculus [100]. For computational simplicity, we will implement a single-stage foveation that increasingly pools features

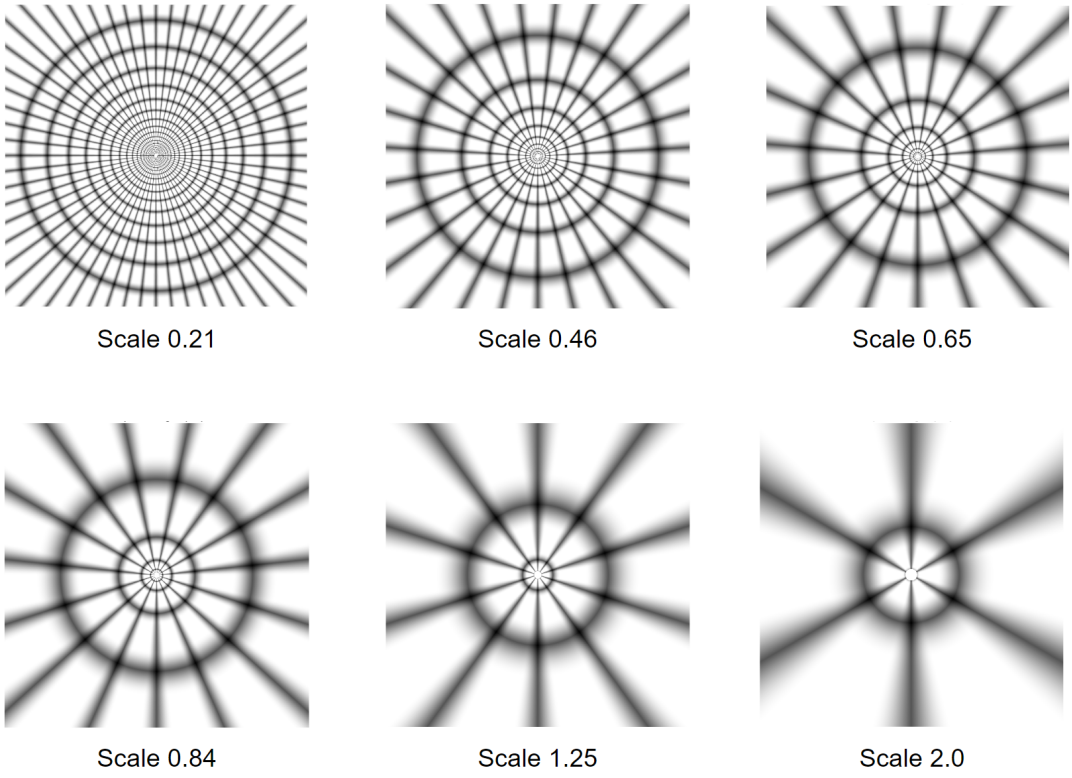

Figure 3.9: **Radial-Polar pooling regions:** Visualization of Radial-Polar pooling regions for different scales. The amount of peripheral pooling is proportional to the value of scale.

spatially with increasing retinal eccentricity. The framework is based on Freeman and Simoncelli [101] spatial pooling on all features extracted from the image. The amount of peripheral pooling is controlled using a parameter called *scale*. Visualization of the pooling regions for different scales is shown in Figure 3.9. Such a simplified model has been shown to capture many aspects of peripheral processing, including some crowding effects [101, 102, 103]. For the FST model, the foveated stage will be implemented at the last convolutional level before the Transformer model.

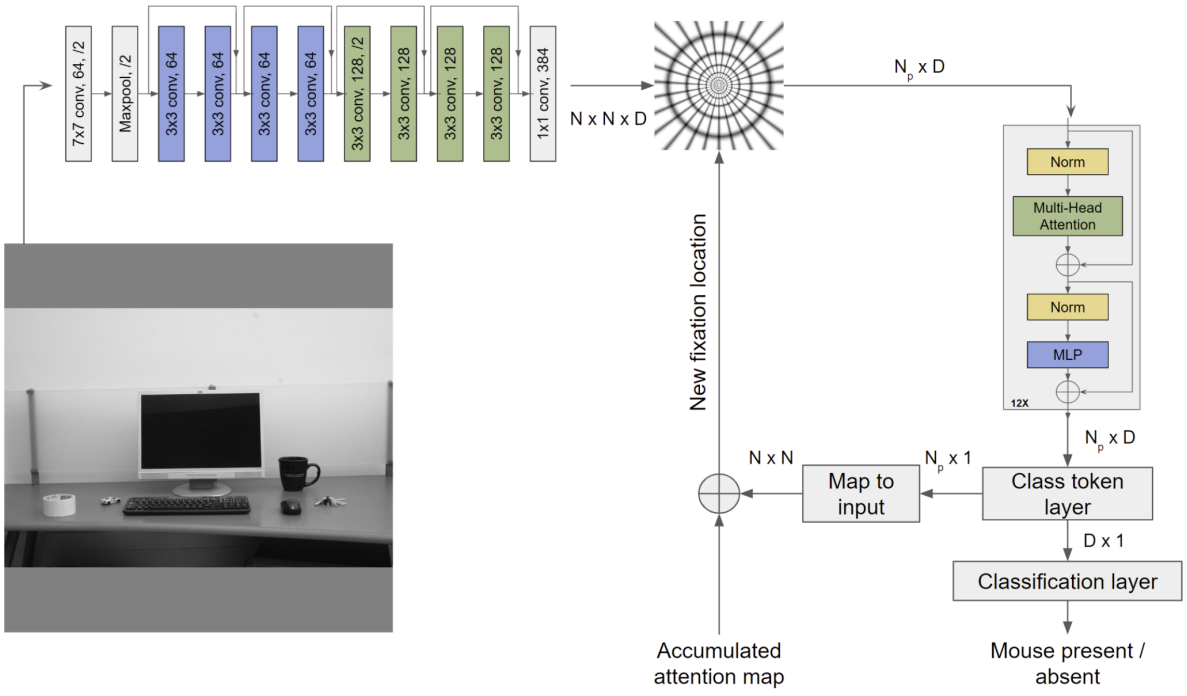

Figure 3.10: **Network architecture:** Foveated Search Transformer model. After passing the input image through a convolution backbone, radial-polar pooling regions operate on the output features of the convolution backbone, followed by the transformer layers. Attention weights of the last transformer layers are used for fixation guidance.

3.3.3 Network architecture

Network architecture closely resembles the FoveaTer architecture and is shown in Figure 3.10. Input image to the model is resized to 512×512 pixels and is passed through the convolution backbone consisting of the first two stages of the ResNet-18 model. Convolution backbone downscales the input image by a factor of 8 resulting in a feature map of size $64 \times 64 \times 128$, followed by a 1×1 convolution resulting in a feature map of size $N \times N \times D$, where N is 64 and D is 384. Radial-Polar pooling regions operate on the output feature map of the convolution backbone and results in one feature vector corresponding to each of the pooling regions, resulting in a feature map of size $N_p \times D$. Pooled features are passed through the twelve transformer layers, at the end of which the feature remains the same in size $N_p \times D$. The feature map of size $N_p \times D$ is then appended to the pre-trained class token feature vector of size $1 \times D$ and passed through a transformer layer. Feature vector corresponding to the class token feature vector is used for image classification, and the attention weights of all the N_p feature vectors towards the class token layer are used for the guidance of the model fixations.

3.3.4 Training

We initialize the model with weights trained for the ImageNet dataset. Since it was trained on the ImageNet dataset with 1000 classes, the final classification of the model has 1000 outputs. For the search task, there are only two possible outputs correspond to the presence or absence of the computer mouse. As a result, we replaced the classification layer with a new fully connected layer with only two outputs and trained these weights from scratch. We are modeling the search task as a classification task instead of formulating a segmentation or object localization task.

The input to the ImageNet classification model is 224×224 resulting in a feature map

spatial size of 28×28 at the output of the convolution backbone. For the FST model, we use a bigger input of size 512×512 resulting in a feature map of size 64×64 at the output of the convolution backbone. The training of the model is done in two stages. It is trained on full images, where the training dataset is constructed using MSCOCO and test dataset is made from a dataset created in the lab. To help the model learn that the target object is the mouse, we create the mouse absent dataset from the mouse present dataset by blocking the mouse location with pixels from the top-left corner. We observed that if such steps are not taken, it can learn to detect the co-occurring objects like monitor and keyboard instead of the mouse. A different solution would be to extend the model to do mouse segmentation, but this was not done as the current work is limited to classification.

The model trained for ImageNet classification is trained on color images. The Input image to the search model is 512×512 and is a colored image, whereas the participants saw the image in grayscale.

3.4 Results

3.4.1 Eccentricity performance:

During the model's training, the model made **four** fixations. For comparing the model performance against human subjects in the mouse eccentricity experiment, the model was fixed at the same location as the humans. Model performance is shown in Figure 3.11, where the x-axis shows the eccentricity at which the mouse was presented in the visual periphery, and the y-axis shows the human/model performance in terms of Area under the ROC. The performance of the individual human subjects and their average performance are shown in black. The model's performance is shown in gray. A

steep drop in the periphery is seen for the scale hyperparameter of 0.46, but it closely follows the human performance for the scale of 0.21. So, the subsequent results are shown for the scale of 0.21.

3.4.2 Fixation guidance mechanism

We evaluated how fixations generated by the model using the self-attention of the last transformer layer compared to the model guided by the fixations generated by the baseline algorithms. The baseline algorithms included random fixations, bottom-up saliency fixations generated by Itti-Koch [19] and GBVS [20] algorithms, and the fixations generated by the deep-learning-based DeepGaze II [42] model. The Itti-Koch, GBVS, and DeepGaze II models generate a map, and the top locations of the map are used as the fixation locations. The performance of the baseline models can be improved by incorporating Inhibition of Return (IoR) into the generation of fixations. IOR is implemented by prohibiting future fixations from being picked from 3×3 region of the feature map surrounding the fixation location, equivalent to 24×24 pixels. Results without IoR are shown in Figure 3.12 and with IoR are shown in Figure 3.13. In addition to the performance metric of Area under the ROC, the minimum distance to the target for each algorithm is shown. Fixations guided by the self-attention map goes to the target faster than the baseline algorithms (Itti-Koch, GBVS and DeepGaze II) for both the cases of with and without IOR.

3.4.3 Contextual effects

In this section, we evaluate if the model exhibits the spatial and scale contextual effects seen in human subjects. Spatial context [10, 104, 35, 105, 36, 39, 106] refers to the spatial position of the computer mouse with respect to the spatial position of the

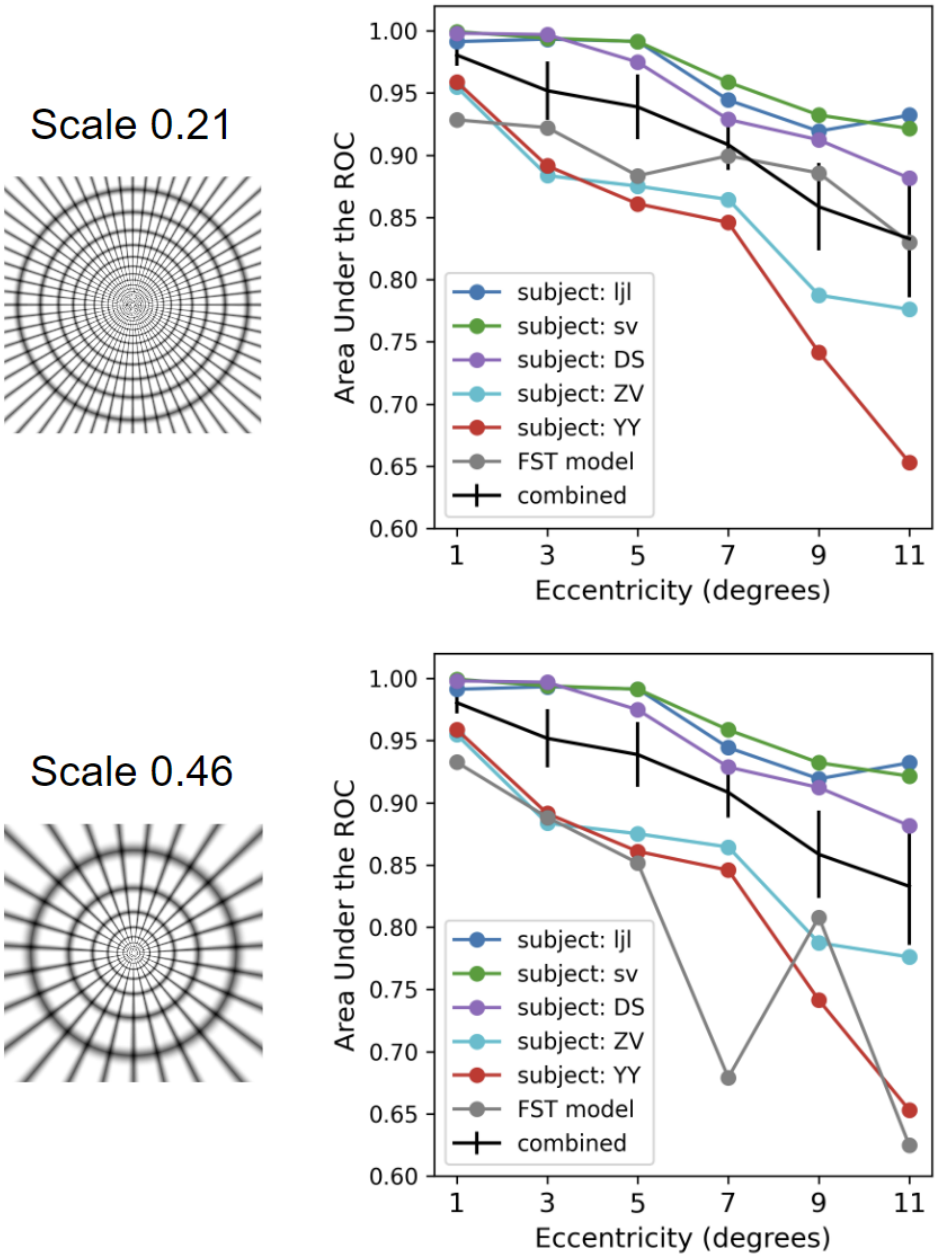

Figure 3.11: **Eccentricity performance: Top:** Results with radial-polar pooling regions generated using the scale hyper-parameter of 0.21 (corresponding to V1). **Right:** Results with radial-polar pooling regions generated using the scale hyper-parameter of 0.46 (corresponding to V2).

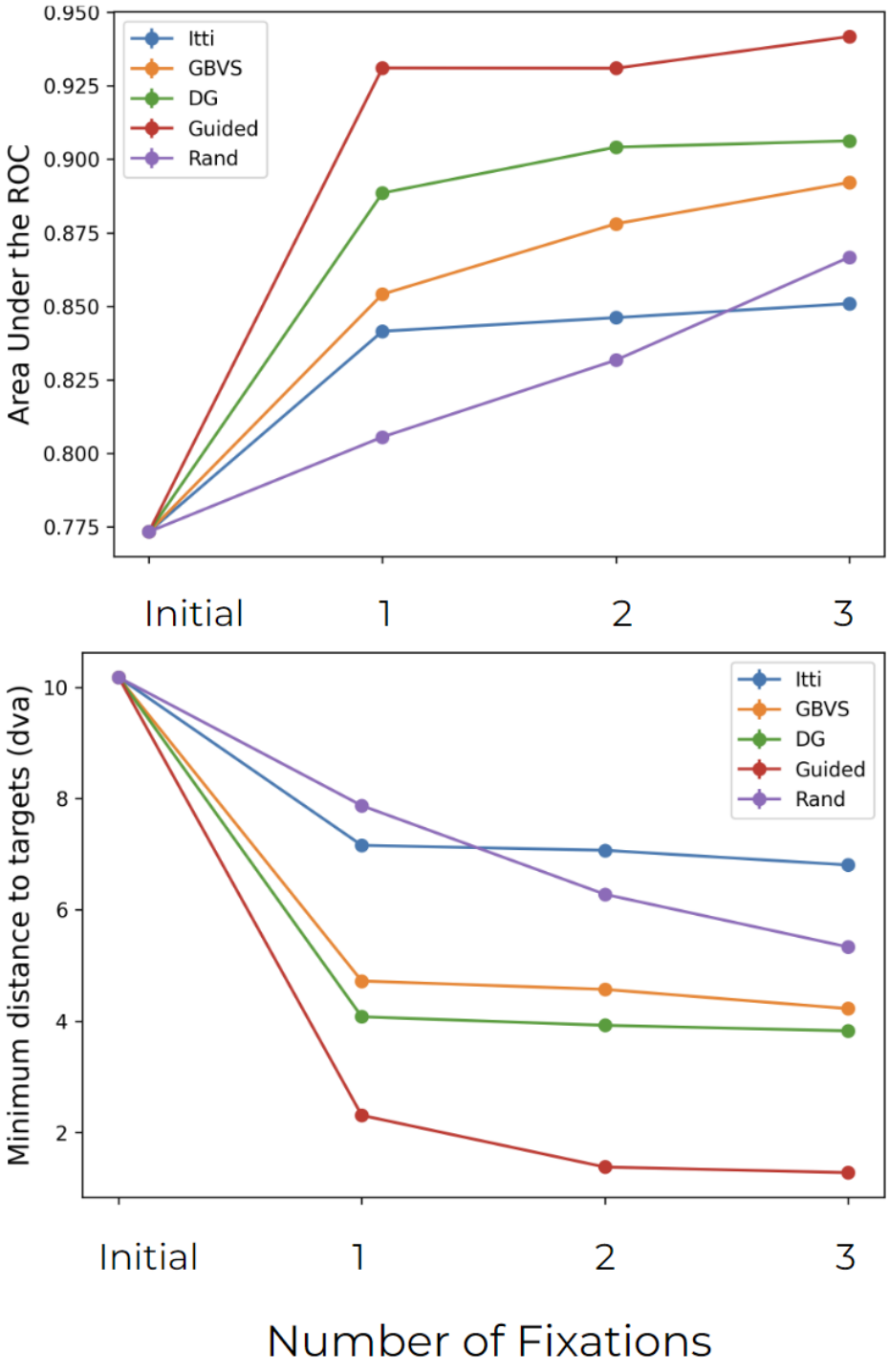

Figure 3.12: **Fixation guidance mechanism without IOR:** Baseline algorithms are guided without IOR. The area under the ROC and the minimum distance to the target are shown. All algorithms started at the same initial fixation.

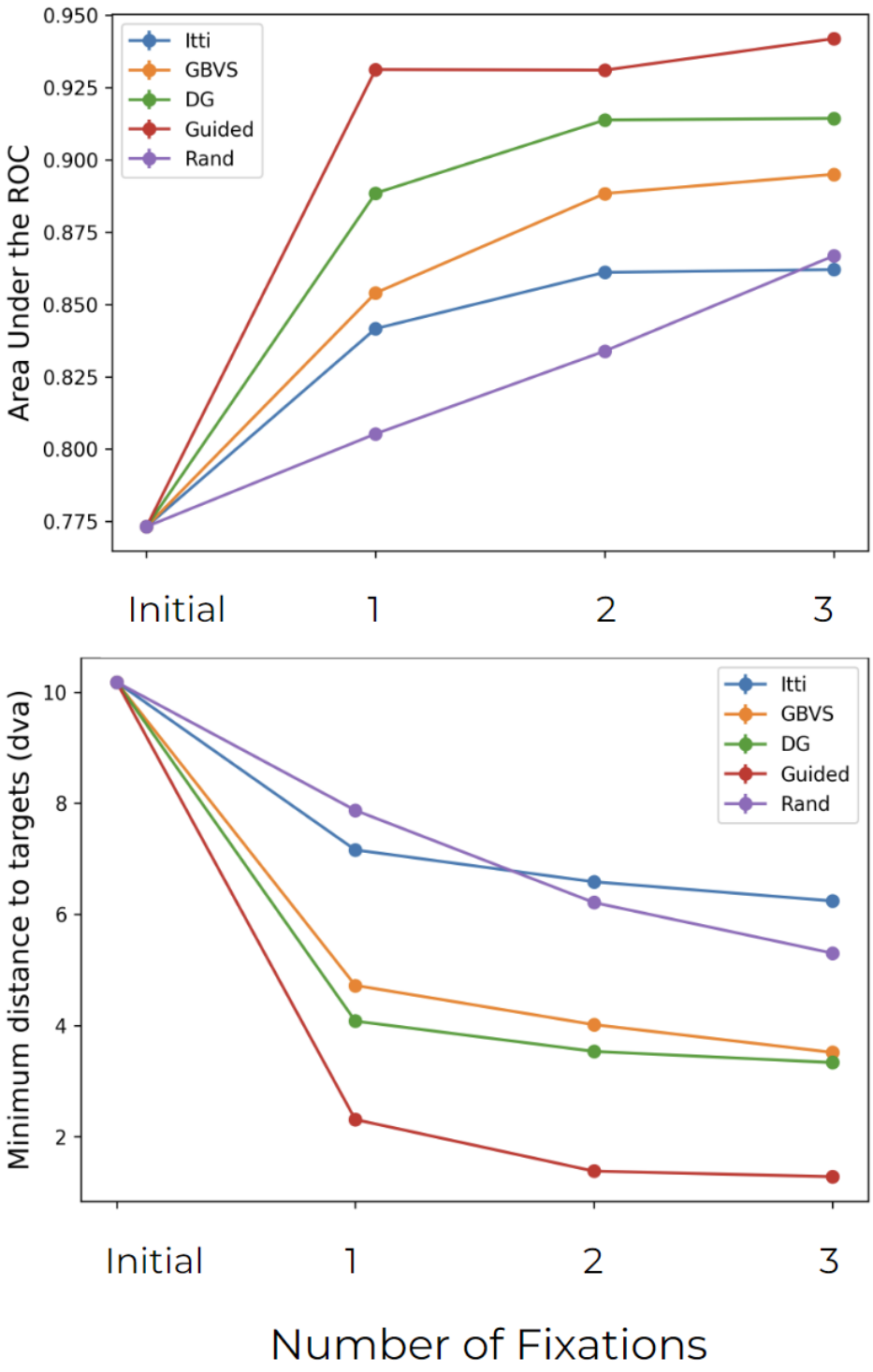

Figure 3.13: **Fixation guidance mechanism with IOR:** Baseline algorithms are guided with IOR. The area under the ROC and the minimum distance to the target are shown. All algorithms started at the same initial fixation.

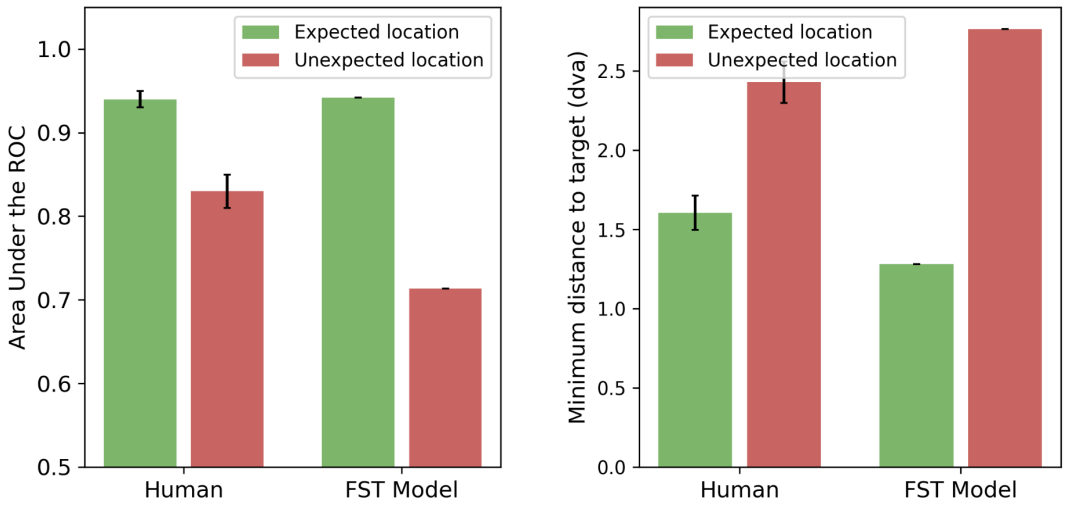

Figure 3.14: **Spatial context:** Performance comparison between human and model after three saccades. **Left:** Human and model performance for the mouse in expected and unexpected locations. **Right:** Distance of the closest fixation to the target location.

co-occurring objects, monitor, and keyboard. It is hypothesized to be easily detectable when present in the expected location, i.e., in reference to the past experience for humans and the training dataset for the model, as opposed to the unexpected location. Scale context [32] refers to the size of the mouse with respect to the scale of the co-occurring objects, monitor, and keyboard. It is hypothesized to be easily detectable when the scale of the computer mouse is of the expected scale, i.e., in reference to the scale of the co-occurring objects.

Spatial context

An example with the mouse in different spatial locations is shown in Figure 3.1. Results from spatial context are shown in Figure 3.14. Humans and the model performed better when the mouse appears at the expected location than the unexpected one. Humans and the model come closer to the target when the target is in the expected position than when the target is in an unexpected location.

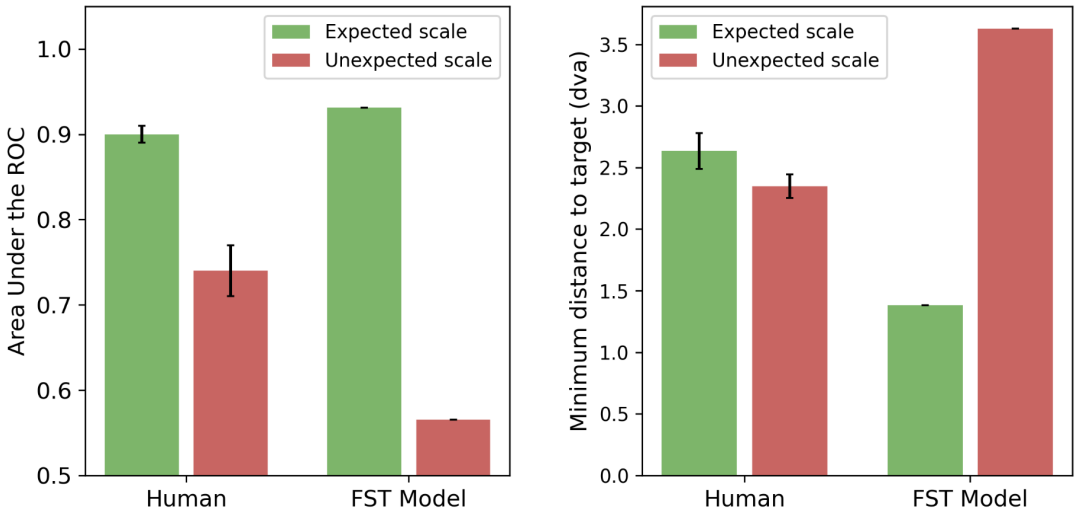

Figure 3.15: **Scale context:** Performance comparison between human and model after two saccades. **Left:** Human and model performance for the mouse at expected and unexpected spatial scales. **Right:** Distance of the closest fixation to the target location.

Scale context

An example with the mouse at different spatial scales is shown in Figure 3.2. Results from spatial context are shown in Figure 3.15. Humans and the model performed better when the mouse was shown at the expected scale than the unexpected one. Interestingly, humans' fovea came closer to the big mouse than the normal-sized mouse, whereas the model went closer to the normal-sized mouse than the big mouse.

3.4.4 Similarity to human fixations

We measure the similarity of human fixations to fellow humans, the FST model, and the baseline models using Kullback-Leibler (KL) divergence. The analysis is repeated separately for the first, second, and third fixations. This also illustrates which model best explains the human fixations. For this purpose, we use three baseline algorithms we previously used to analyze the fixation guidance mechanism: Itti-Koch, GBVS, and DeepGaze II. These methods generate the maps in a task-independent way, whereas

humans and the model are performing task-dependent fixations. As before, the baseline algorithms have access to the full image to predict the fixation location, whereas the foveated visual field limits the peripheral view of the humans and the FST model.

We acquired the fixation paths of the human subjects, the FST model, and the baseline algorithms. For each fixation condition (first, second, or third), the fixation map is smoothed using convolution with a Gaussian filter, and the KL-Divergence is computed for each image and each subject with respect to the fixations made by all the human subjects on that image other than the current subject. Results for fixations first, second and third fixations are shown in Figure 3.16, Figure 3.17 and Figure 3.18, respectively. For the first fixation, the baseline GBVS algorithm explains the human fixations the best. For second and third fixations, the FST model outperforms other baseline models in explaining the human fixations the best. This can be potentially attributed to the task-dependent nature of fixation guidance and the usage of the foveated visual system by both humans and the FST model.

3.4.5 Similarity of fixations with distribution map of relative location of objects in scenes (prior map)

To understand if the model learns the contextual knowledge, we first generate a map of the mouse locations with respect to the monitor in the training dataset. We refer to this map as the prior map. We compare the prior map to fixation distributions to assess whether eye movements guided by the model and humans are going to those locations. This can be captured using the KL-divergence between the prior map generated from natural image statistics and the heatmap of the model/humans generated from using different fixation guidance mechanisms.

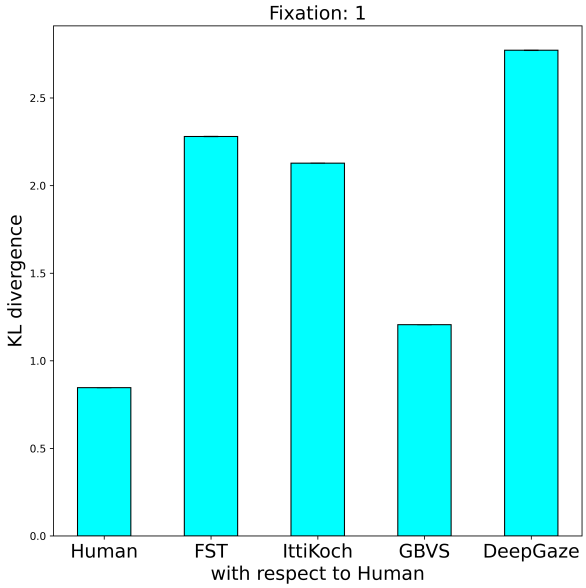

Figure 3.16: **Similarity to human fixations for first fixation:** Human-Human have the least KL-Divergence followed by GBVS-Human. DeepGaze-Human has the highest KL-Divergence. Human-Human < GBVS-Human < IttiKoch-Human < FST-Human < DeepGaze-Human

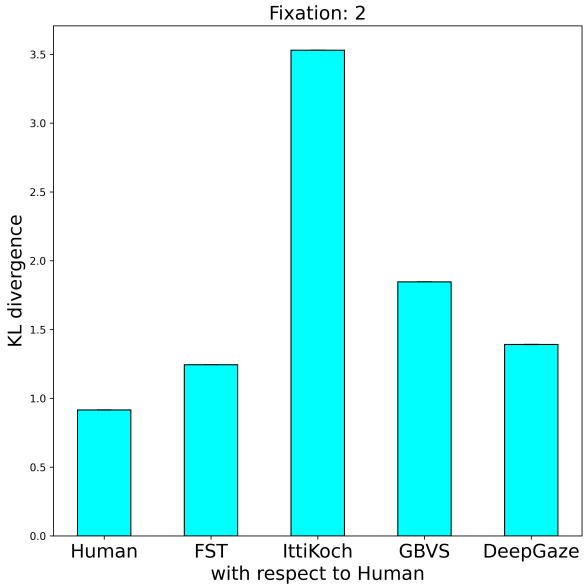

Figure 3.17: **Similarity to human fixations for second fixation:** Human-Human still have the least KL-Divergence now followed by the FST-Human. IttiKoch-Human has the highest KL-Divergence. Human-Human < FST-Human < DeepGaze-Human < GBVS-Human < IttiKoch-Human

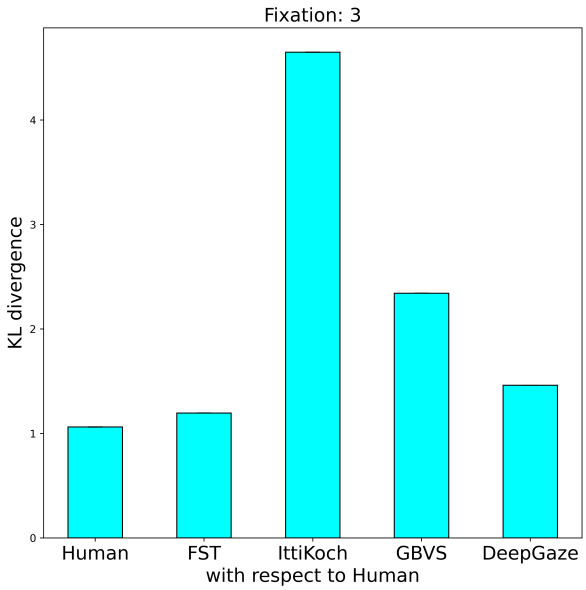

Figure 3.18: **Similarity to human fixations for third fixation:** Human-Human still have the least KL-Divergence now followed by the FST-Human, where the difference between the both is no longer significant. Human-Human = FST-Human < DeepGaze-Human < GBVS-Human < IttiKoch-Human

Generation of prior map

We use the MSCOCO training dataset to generate the prior map. First, we identify all the instances of the mouse, tv, and laptop in a given image using the annotation data available in the dataset. In the dataset, a normal desktop monitor has been annotated as tv and for this reason, we collect the tv instances. Next, for each mouse instance, we look for the closest tv or laptop instance and record the center coordinates of the monitor and the mouse, along with the width and height of the monitor. After collecting the information from all mouse instances of the training image dataset, we register the information on a common map. Since the monitor instances are of different sizes, they must first be normalized. We scale the monitor's width and height to 100 pixels each, resulting in width and height scaling values we use to normalize the distance between the monitor and mouse in x and y coordinates. After normalization, all the mouse instances

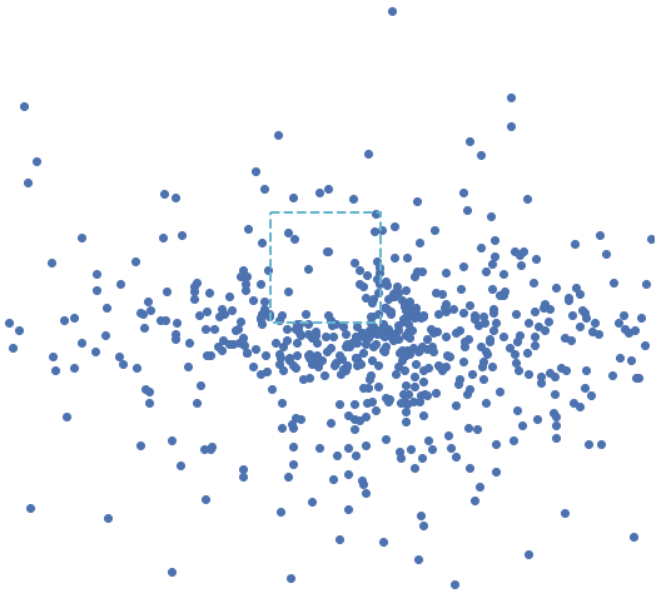

Figure 3.19: Mouse points registered to a normalized monitor from all the training images. A normalized monitor is shown as the dotted cyan square.

are mapped onto the same map.

The registered mouse instances with respect to the monitor is shown in Figure 3.19. We then convert the map of mouse instances to the monitor into a heatmap by smoothing it with a Gaussian filter. This results in a heatmap shown in Figure 3.20.

Fixation heatmaps

To understand the spatial distribution of fixations and the human/model biases of fixating at specific locations, we register the fixations made by humans and the model guided by different fixation mechanisms (GBVS / DeepGaze II / Itti-Koch) with respect to the monitor to create the normalized map. Because the monitor's size varies across images depending on the distance between the camera and the monitor and the magnification, we normalize the size of the monitor across images so that the distance of the

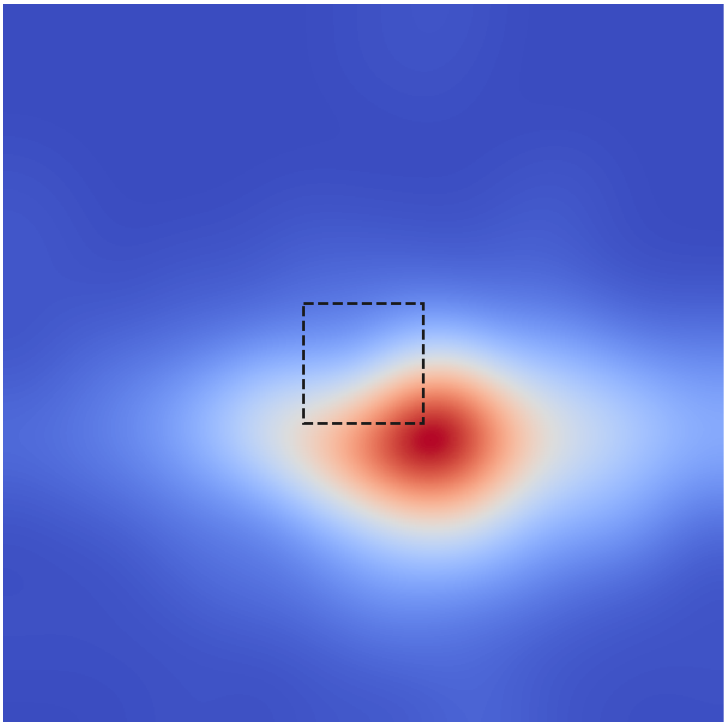

Figure 3.20: Heatmap of mouse points to a normalized monitor from all the training images. A normalized monitor is shown as the black square in the middle.

fixations to the monitor are measured in the same units. Next, we convolved the resultant map with the Gaussian filter to generate a low-pass filtered heatmap. An example is shown in Figure 3.21.

Finally, we use KL-Divergence to compare the fixation heatmaps against the prior map to see if humans/models guided by different fixation mechanisms are biased towards going to the hot spots of the prior map. A lower KL-Divergence value, implying a higher similarity, shows that the human/model would tend to go to the expected mouse location.

We use only the Nth fixation to compute the heatmap corresponding to that fixation. We repeat the analysis twice: First, considering all images for the similarity analysis i.e., mouse present/absent, and Secondly, considering only the mouse-absent images. The 2nd comparison is of more interest because fixations in mouse-present images are guided both by context but also the presence of the target. The analysis of target absent images

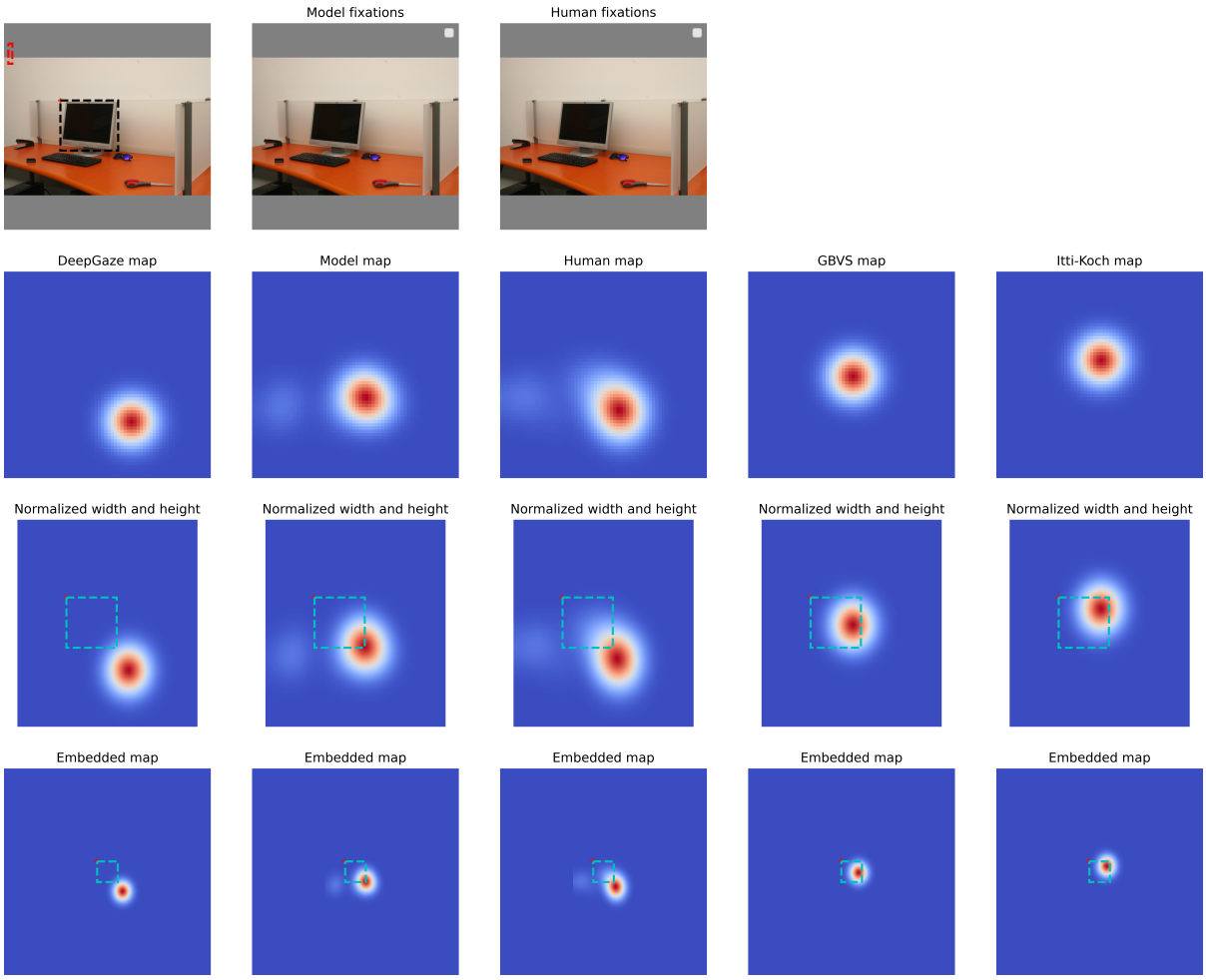

Figure 3.21: **Visualization of heatmaps:** Heatmap is generated using the Nth fixation followed by normalization based on the monitor size. **Top row:** Original image with annotated monitor. **Second row:** Heatmap generated from the Nth fixation made by humans, FST model and the baseline algorithms. **Third row:** Heatmap normalized to make the monitor into square shape. **Fourth row:** Normalized heatmap copied onto a bigger map, where maps from all images are summed together.

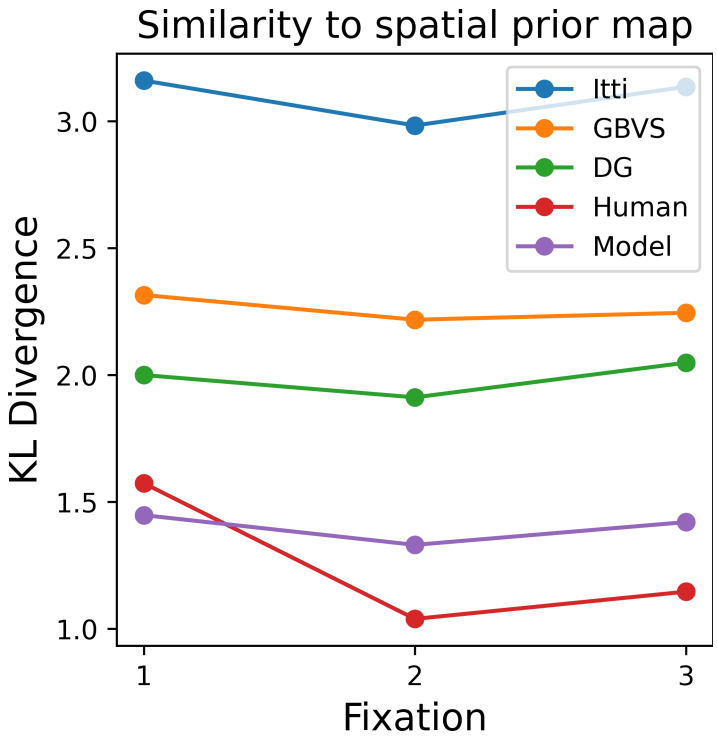

Figure 3.22: **Similarity to prior map using both mouse-present and mouse-absent images:** KL-Divergence is used to measure the similarity of Human/model heatmaps and the prior map generated using natural image statistics. **For the first fixation,** Model-Prior < Human-Prior < DeepGaze-Prior < GBVS-Prior < IttiKoch-Prior. **For the second and third fixations,** Human-Prior < Model-Prior < DeepGaze-Prior < GBVS-Prior < IttiKoch-Prior.

allows us to isolate influences of context on fixations.

Similarity analysis using mouse present and absent images

The results of doing this analysis on all the images are shown in Figure 3.22. Analysis for Fixation 1, 2, and 3 are shown in Figure 3.23, Figure 3.24, and Figure 3.25 respectively.

Similarity analysis using mouse-absent images only

The results of doing this analysis only on the mouse absent images are shown in Figure 3.26. Analysis for Fixation 1, 2, and 3 are shown in Figure 3.27, Figure 3.28 and

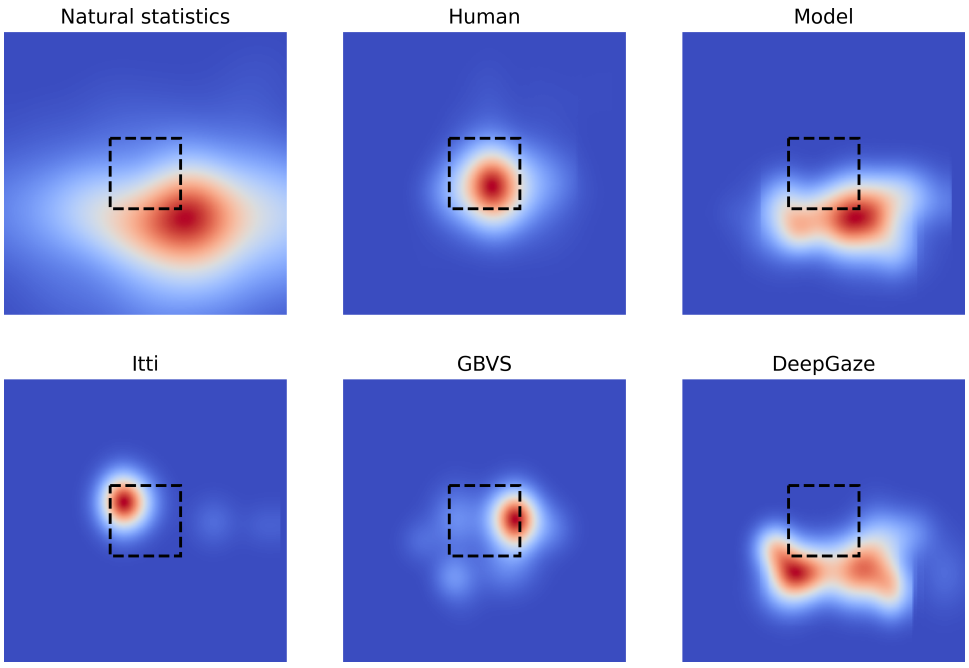

Figure 3.23: **All images - Fixation 1:** Humans tend to go to the center of the monitor. In contrast, the model goes directly to the expected location of the mouse, resulting in lower KL divergence between the model and the prior map.

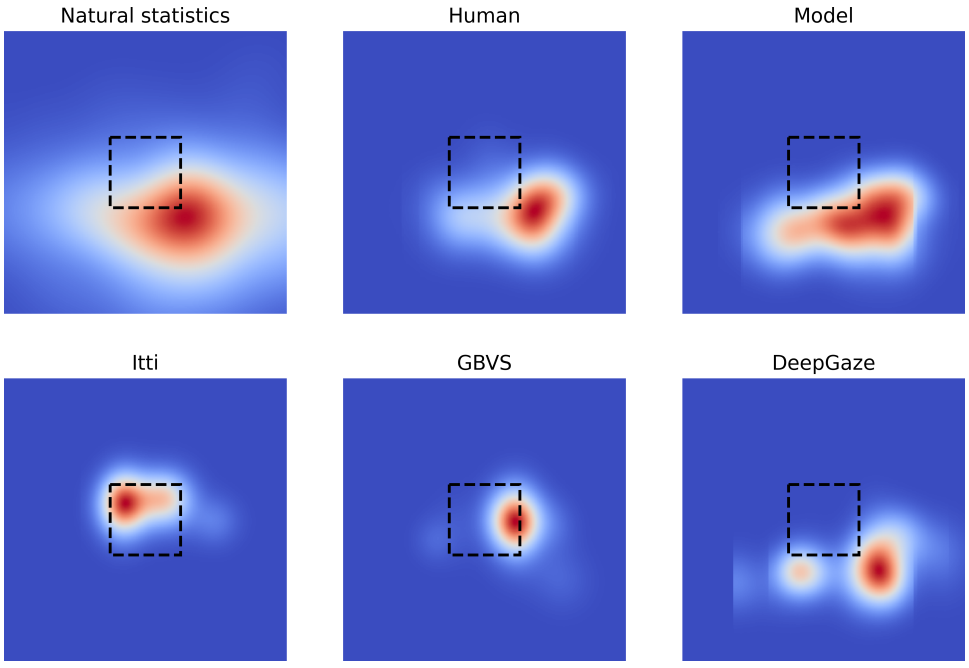

Figure 3.24: **All images - Fixation 2:** Using the second fixation, humans explore the expected mouse location, thereby decreasing the KL divergence with the prior map.

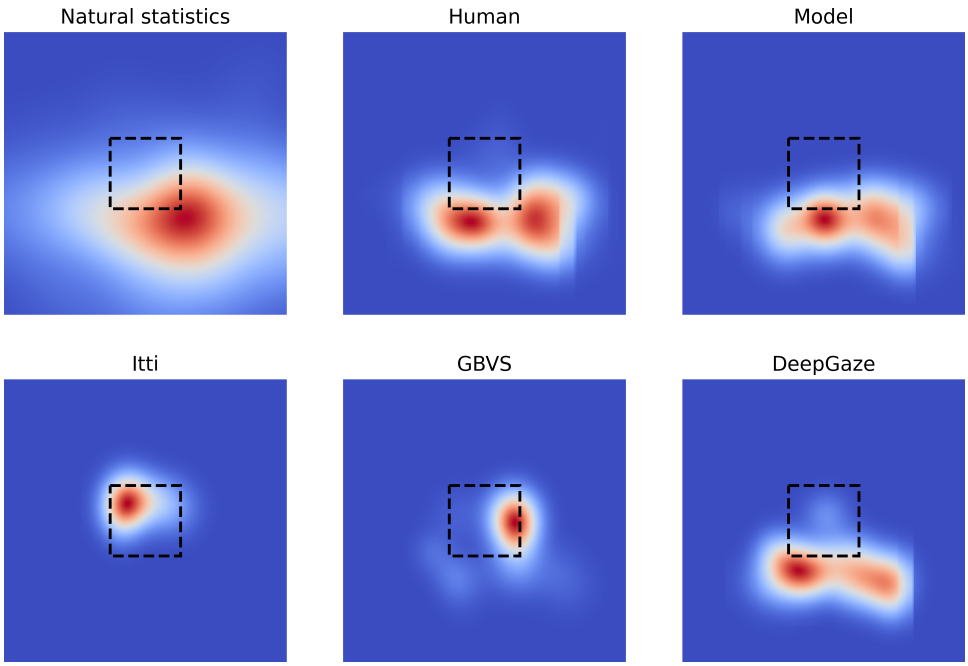

Figure 3.25: **All images - Fixation 3:** Using the third fixation, humans explore the unexpected location after exploring the expected location using their second fixation.

Figure 3.29 respectively.

Humans show a tendency to direct their fixation toward the monitor, whereas the FST direct its first fixation directly to the mouse instead of stopping at the monitor. Consistent with the more frequent location of a mouse occurring to the right of the monitor, the FST model guided by self-attention displayed a similarly strong right bias as humans whereas the model guided with the other algorithms (Itti-Koch, GBVS and DeepGaze II) did not display such a strong right bias.

3.5 Conclusion

In this chapter, we extended the foveated image classifier trained on ImageNet for the task of computer mouse search. We performed two psychophysics experiments: First, to get the human detection performance of the computer mouse when it is presented

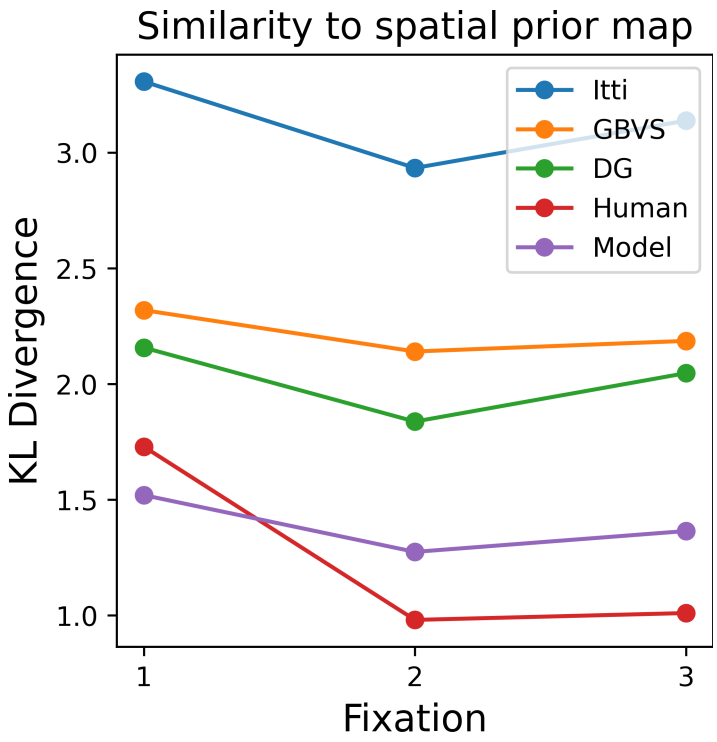

Figure 3.26: **Similarity to prior map using mouse-absent images:** KL-Divergence is used to measure the similarity of Human/model heatmaps and the prior map generated using natural image statistics. **For the first fixation,** Model-Prior < Human-Prior < DeepGaze-Prior < GBVS-Prior < IttiKoch-Prior. **For the second and third fixations,** Human-Prior < Model-Prior < DeepGaze-Prior < GBVS-Prior < IttiKoch-Prior.

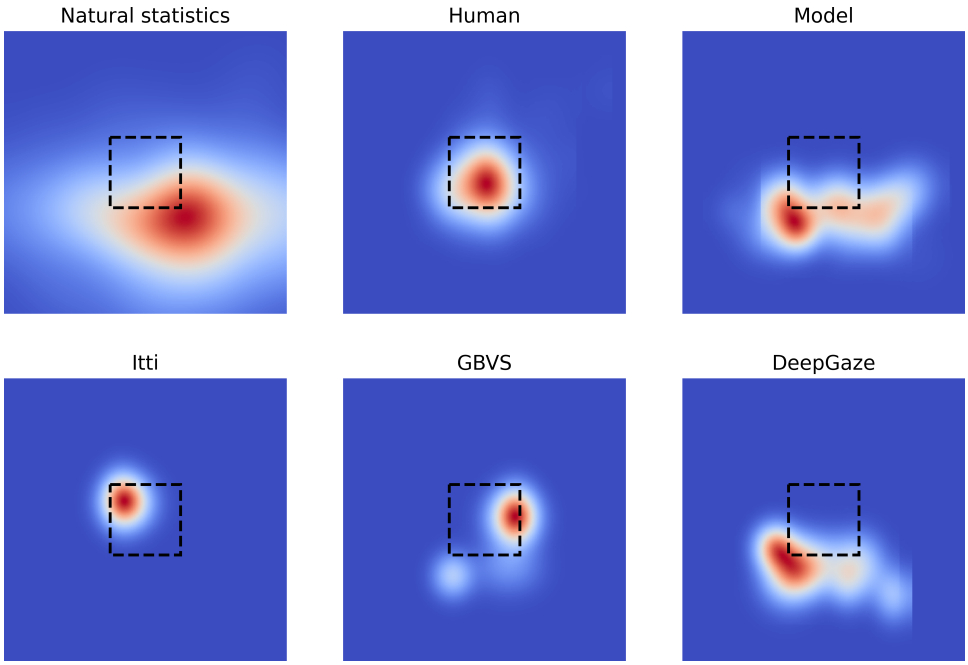

Figure 3.27: **Mouse-absent images - Fixation 1:** Even when the mouse is absent, humans tend to go to the center of the monitor. In contrast, the model goes directly to the expected location of the mouse, resulting in lower KL divergence between the model and the prior map.

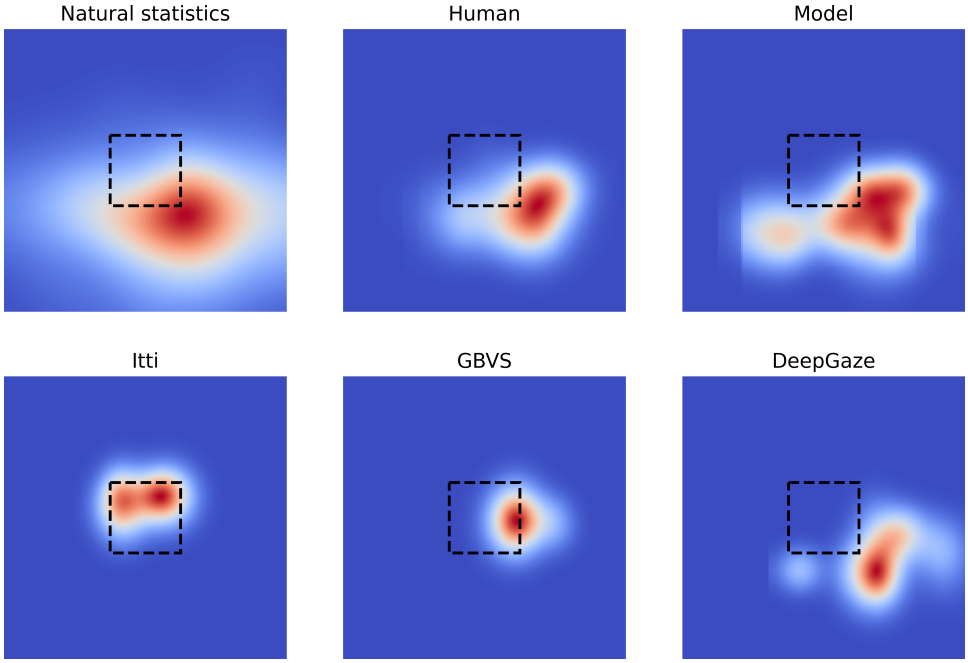

Figure 3.28: **Mouse absent images - Fixation 2:** Even when the mouse is absent, humans explore the expected mouse location using the second fixation, thereby decreasing the KL divergence with the prior map.

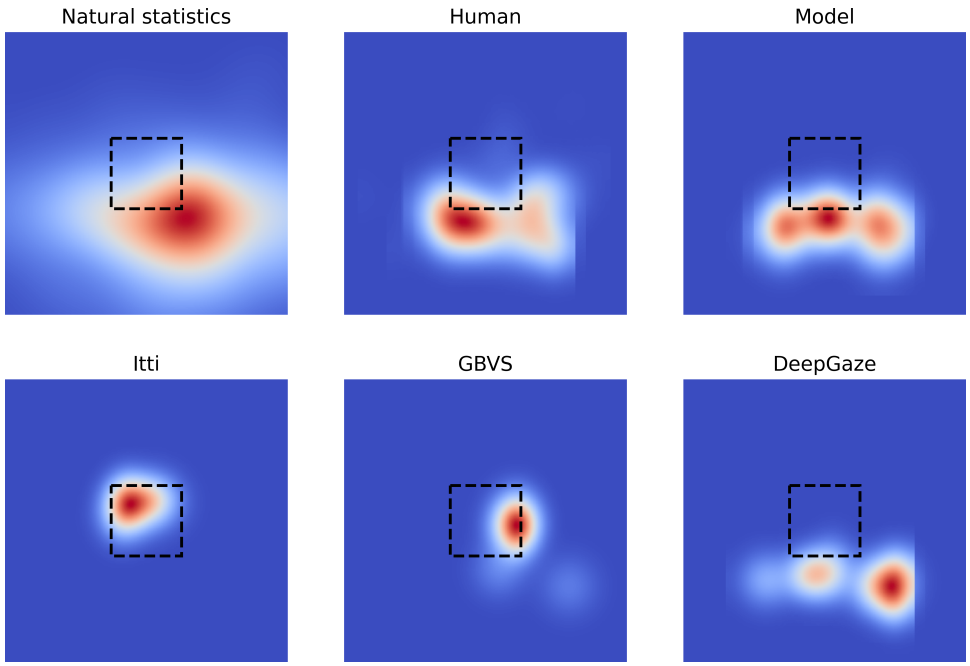

Figure 3.29: **Mouse absent images - Fixation 3:** Even when the mouse is absent, humans explore the unexpected location using their third fixation after exploring the expected location using their second fixation.

at different eccentricities in the visual periphery, and Second, to illustrate the context effects seen for the task of Visual Search for humans. In section 3.4.1, we fixed the *scale* hyperparameter of the radial-polar pooling regions to 0.21 to match the model performance to human performance in the visual periphery. In section 3.4.2, we showed the effectiveness, in terms of Area under the ROC and ability to go towards the target location, of the fixation guidance using self-attention (guided model) by comparing it against the fixation guidance using different algorithms such as Itti-Koch, GBVS and DeepGaze II. In section 3.4.3, we then showed that the model exhibits similar spatial and scale contextual effects shown in the human subjects. In section 3.4.4, we showed the fixations made by the guided model are most similar to humans for second and third fixations compared to the other comparison algorithms. In section 3.4.5, we showed that fixation heatmaps of the guided model, even for the case of mouse-absent images, are more similar to a map of frequent/expected locations for the computer muse than the

baseline algorithms suggesting the contextual influence of the co-occurring objects on the fixation guidance of the guided model.

Chapter 4

Convolution Neural Network based Model Observer for Virtual Phantoms

The ideal observer model provides the upper bound performance when the statistics of the signal and background are entirely known. For anatomical, phantom, simulated backgrounds where the statistics are unknown, linear model observers, such as Channelized Hotelling Observer, can predict human performance in detection tasks with a few possible signal locations. Here we compare the detection accuracy of the Channelized Hotelling Observer (CHO) [107, 108, 109, 110] and a Convolution Neural Network (CNN), against the radiologists' performance for two types of signals embedded in 2D/3D breast tomosynthesis phantoms (DBT). We show that for a signal known exactly task, the CHO model's accuracy is comparable to the CNN's. However, for the search task with 2D/3D DBT phantoms where the signal can appear in any location, the Channelized Hotelling models' detection accuracy was significantly lower than that of radiologists. In contrast, a CNN's accuracy was comparable to or higher than the radiologists. An analysis of

the eye position of radiologists showed that they fixated more often and longer times at the locations corresponding to CNN false positives. Many of the CHO false positives related to the phantom's anatomy were not fixated by radiologists. In conclusion, we show that CNN can be used as an anthropomorphic model observer for the search task where the traditional model observers fail due to complex backgrounds. In this work, we first train three model observers and compare their performance against the radiologists' performance for two types of signals, a simulated calcification (CALC) and a simulated mass (MASS). We also perform a false positive analysis where we compare the response maps of the model observers against the time spent by the radiologists at their fixation locations.

4.1 Introduction

For signals present at one or a few specified locations and anatomical backgrounds, model observers have been extensively used for image quality assessment in the field of medical imaging [107, 108, 111, 112]. When the signal and background statistics are completely known, an ideal observer (IO) implementation is feasible, and it provides the upper limit on performance for any observer on a perceptual task and is used to benchmark human performance [113, 114, 115, 116, 117]. When the exact statistics are unknown, IO implementation is not feasible. We must rely on sub-optimal observers such as Channelized Hotelling Observer (CHO) [118], involving a feature extraction stage through a set of linear channels. When the early visual processing in the human visual system is approximated using these linear channels, CHO becomes a better anthropomorphic model observer than an ideal observer [119].

In recent years, Convolution Neural Networks (CNN) have also been used as model observers in medical imaging [120, 121]. For the simple task of signal-known-exactly

and background-known-exactly, CNNs provided an excellent approximation to ideal observer [122, 123]. For the task of Signal-Known-Exactly (SKE) and Background-Known-Statistically (BKS), new training strategies for CNN-based anthropomorphic model observers had achieved good performance agreement with human observers [124]. Furthermore, for the SKE-BKS condition, CNNs were trained with adversarial robust training with the aim to generate more human-interpretable features [125]. For a defect forced-localization task, an exploration across different backgrounds, hyperparameters and loss functions, mean-squared error provided the best results [126]. A segmentation-based CNN architecture provides a basic output visualization without the necessity of advanced interpretability techniques [127, 128, 129]. nnU-Net [130] is U-Net-based [131] segmentation architecture capable of automating data pre-processing, network architecture, model training, and post-processing without manual intervention and provides a general-purpose solution for 23 publicly available medical-image datasets.

For simple backgrounds and Location-Known-Exactly tasks, model observers such as CHO and Filtered Channel Observer (FCO) [132] provide a good Ideal Observer approximation. However, for the search task and complex backgrounds, the difficulty increases along with uncertainty in the target's location. Due to a lack of a good understanding of the background, traditional observers (CHO and FCO) suffer in performance under these circumstances. In such cases, Convolution Neural Networks (CNN) can provide better alternatives [122, 123, 124, 125, 126, 130]. Due to the difference in architecture and training methodologies, CNNs can be used to train the 3D templates directly for the search task. As a result, they develop a thorough understanding of both signal and background simultaneously, making CNN a good alternative for 3D search tasks where IO implementation is infeasible.

This work explores the possibility of using segmentation-based nnu-Net [130] as an anthropomorphic model observer on a dataset of Digital Breast Tomosynthesis (DBT)

virtual phantoms [133], a dataset for which we have eye-tracking data from twelve radiologists. Firstly, we explore the effects of switching from a Location-Known-Exactly (LKE) task to a search task on model observers such as Channelized Hotelling Observer (CHO), Filtered Channel Observer (FCO), and Convolutional Neural Network (CNN). Secondly, we compare the performance of CHO, FCO, and CNN for 2D search against 3D search. Thirdly, by making use of the fixation location data and the amount of time spent by the radiologists at those locations, we compare the similarity of the response maps of all three model observers to those of the time-spent maps from radiologists. Using a general-purpose CNN network architecture, which was shown to perform well on different medical images, we maintain the potential generalization ability to new datasets. Moreover, the very realistic DBT dataset makes the model capable of adapting to real-world datasets.

4.2 Materials and Study

4.2.1 Dataset

DBT phantoms, the images used for the radiologist study and testing the model observers in this paper, were generated by the OpenVCT virtual breast imaging tool from the University of Pennsylvania [133, 134, 135]. This tool generates full phantom DBT images, including different tissues (skin, Cooper's ligaments, adipose, and glandular) in a realistic manner. Generated phantoms are of size $2048 \times 1792 \times 64$, where 64 is the number of slices for each phantom. The dataset contains two types of signals: 1. a small lesion similar to microcalcification and 2. a large lesion similar to a mass. We have a dataset of about 500 phantoms with microcalcification, 500 phantoms with mass, and 500 signal-absent phantoms. For 2D experiments, we used a single slice of the phantom,

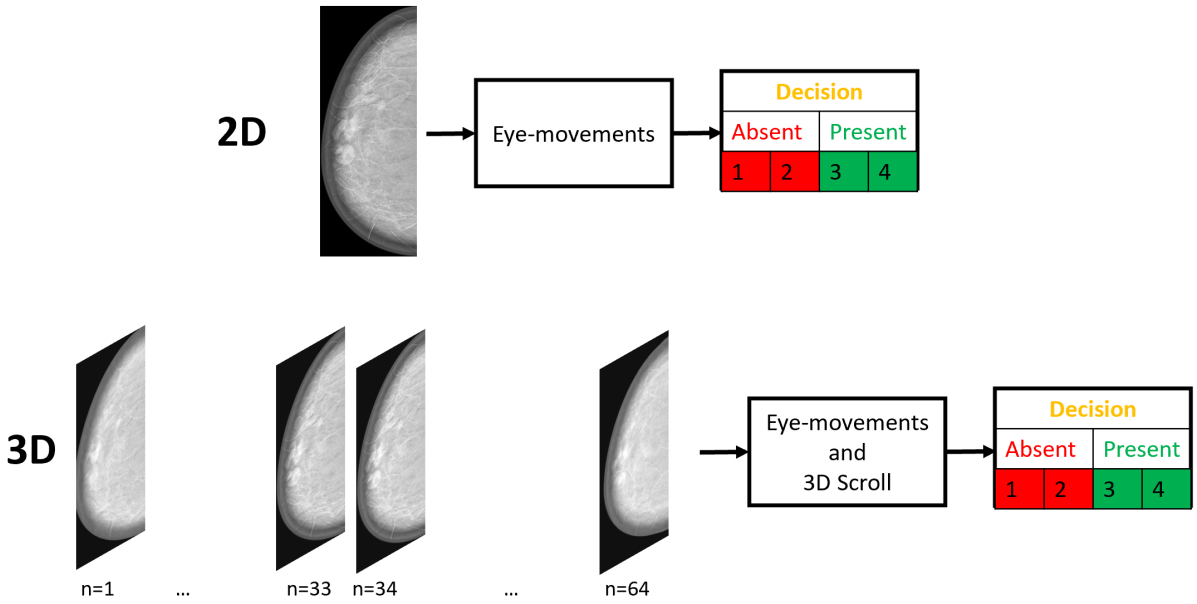

Figure 4.1: **Radiologist study:** Twelve radiologists participated in the study. Each of them was shown a total of 28 2D phantoms and 28 3D phantoms. Half of each set had signal-present and another half with signal-absent. Signal-present was made up of an equal number of CALC and MASS. After making a decision, they decided on a 4-point decision scale where four corresponds to strong confidence that the signal is present. For 2D DBT, only eye movements were made, and for 3D, eye movements and scrolling across slices were possible.

which contained the signal's center as the signal-present phantom.

4.2.2 Radiologist study

Twelve radiologists participated in the study. They sat 75 cm away from a vertical medical-grade monitor in a darkened room. Stimuli were displayed in a 5Mpx grayscale DICOM calibrated monitor (2560×2048 pixels), keeping their aspect ratio. In the study, each radiologist saw 28 signal-present (microcalcification or mass) and 28 signal-absent trials, out of which half were 2D trials and the other half were 3D trials. The prevalence between microcalcification and masses was 50%. Microcalcification and mass extended visual angles of 0.06 and 0.5 degrees, respectively. All four conditions were randomly intermixed. An eye tracker recorded the participant's eye movements in real-time at a

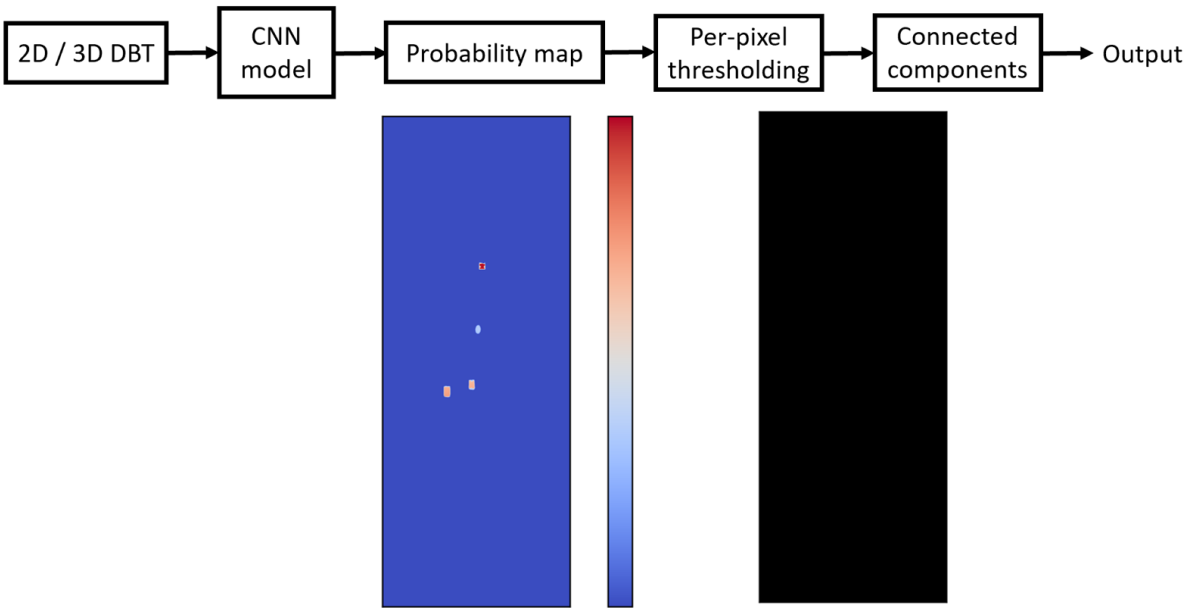

Figure 4.2: **CNN model:** Four different CNN models are trained for two modalities (2D/3D) and two signal types (microcalcification/mass). During the test phase, the segmentation-based CNN produces a probability map with a probability value assigned to each pixel, representing its probability of being the signal pixel. A per-pixel threshold computed using the validation set is applied to the CNN output to convert the probability map into a binary map. Connected components are computed using 8 and 26 connectivity for 2D and 3D, respectively, on the resultant binary map.

frequency of 500Hz (EyeLink Portable Duo, SR Research). We used Psychtoolbox to develop this experiment [136].

4.3 Model observers

An Ideal Observer (IO) optimally uses visual information to compute posterior probabilities and achieves upper-bound performance. It is impossible to design an IO for DBT phantoms due to the absence of complete signal and background statistics information. We trained three model observers: 1. Channelized Hoteling Observer (CHO), 2. Filtered Channel Observer (FCO) and 3. Convolution Neural Network (CNN). We used a set of approximately 1500 phantoms for training the model observers.

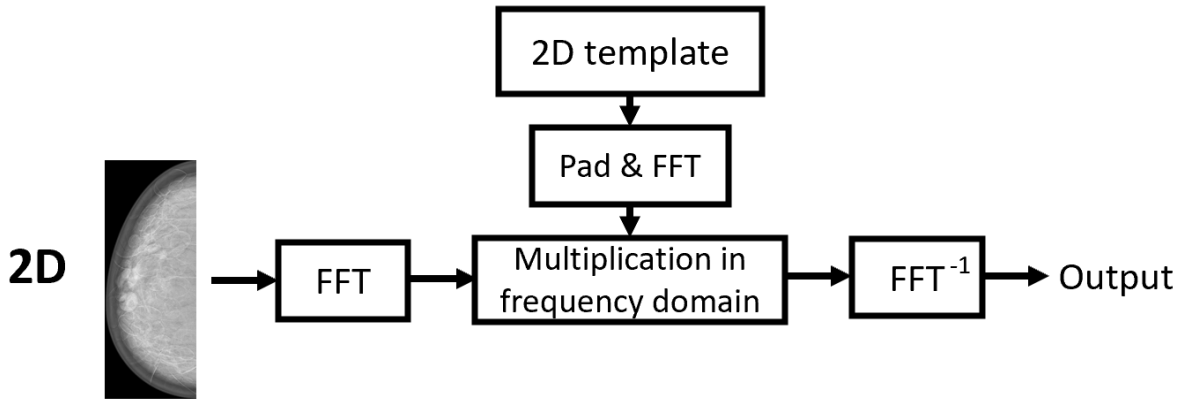

Figure 4.3: **2D search with CHO and FCO:** During the testing phase, the 2D template is padded to the size of the 2D input. After taking the Fast Fourier Transform (FFT) of both the input and template, the convolution of the two is performed in the Fourier domain by performing their multiplication. The final response map is generated by taking the inverse FFT of the convolution output in the frequency domain.

4.3.1 Gabor channel computation

We used Gabor channels with N_o orientations, N_p phases, and N_f spatial frequencies, resulting in a set of $N_o \times N_p \times N_f$ Gabor channels, represented by C^{Gabor}.

$$i^{'} = i\cos(\theta_k) + j\sin(\theta_k) \tag{4.1}$$

where i, j are the x-y coordinates, θ_k is the kth orientation and $i^{'}$ is the transformed x coordinate. And kth channel of the Gabor channel array, C_k^{Gabor}, is computed using,

$$C_k^{Gabor}(i,j) = Gaus(i,j) * \cos(2\pi * fc_l * i^{'} + \beta_p) \tag{4.2}$$

where $Gaus(i,j)$ is the Gaussian envelope, fc_l is the lth central frequency and β_p is the pth phase.

4.3.2 Channelized hotelling observer (CHO)

Using dot product, we compute the response of the training image crops to the Gabor channels (from Equation 4.2),

$$V = (C^{Gabor})^T S \tag{4.3}$$

where S_P and S_A are the arrays of signal-present and signal-absent training crops, respectively, and $S \in \{S_P, S_A\}$; V_P and V_A are the arrays of signal-present and signal-absent responses respectively and $V \in \{V_P, V_A\}$; T is the transpose operation.

Signal-present and Signal-absent covariance matrices, K_P and K_A, are computed using the response arrays V_P and V_A The mean of signal-present and signal-absent covariance matrices results in the channel covariance matrix.

$$K_{ch} = (K_P + K_A)/2 \tag{4.4}$$

Weights of the linear CHO are computed,

$$W_{CHO} = V^T K_{ch}^{-1} \tag{4.5}$$

Response of test images (g) to the CHO is computed using convolution in the frequency domain.

$$\lambda = \text{FFT}^{-1}(\text{FFT}(W_{CHO})\text{FFT}(g)) \tag{4.6}$$

Where FFT is the Fast Fourier transform, FFT^{-1} is the inverse Fast Fourier transform, and the CHO template is padded with zeros before taking the FFT to match its size to that of the test images.

4.3.3 Filtered channel observer

Filtered Channel Observer (FCO) [132] is a linear model observer based on convolution channels capable of modeling irregularly-shaped signals with fewer directional channels. Constructing the weights of FCO follows a similar procedure to that of CHO. The implementation differs only in the construction of the channel templates.

Signal mean ($signal_{mean}$) is the difference between the mean signal-present image and the mean signal-absent image. Furthermore, using Fast Fourier Transform (FFT), Signal mean ($signal_{mean}$) is transformed into frequency domain ($signal_{FFT}$).

$$signal_{mean} = (1/N_P) * \sum_{N_P} S_P - (1/N_A) * \sum_{N_A} S_A \tag{4.7}$$

$$signal_{FFT} = \mathrm{FFT}(signal_{mean}) \tag{4.8}$$

where S_P and S_A are signal-present and signal-absent training crops respectively, and N_P and N_A represent the number of signal-present and signal-absent training images.

Gabor channel array (Equation 4.2) is transformed using the signal mean in the frequency domain and is then converted back into the spacial domain using inverse Fast Fourier Transform (FFT^{-1}), followed by *L-2* normalization.

$$C_k^{'} = (1/nxy) * \mathrm{ABS}(\mathrm{FFT}(C_k^{Gabor}))^2 \tag{4.9}$$

$$C_k^{FCO} = \mathrm{FFT}^{-1}(C_k^{'} * signal_{FFT}) \tag{4.10}$$

$$C_k^{FCO} = C_k^{FCO} / sqrt(\sum_x \sum_y (C_k^{FCO})^2) \tag{4.11}$$

where C_k' is the power spectral density of the k^{th} channel C_k^{Gabor}, nxy is the spatial size of the channel C_k^{Gabor}, C_k^{FCO} is the k^{th} channel of the FCO model.

After the computation of the FCO channels, the template is computed similarly to the CHO template by replacing C^{Gabor} with C^{FCO} in Equation 4.3 followed by computation of template weights and the response map using Equations 4.4, 4.5 and 4.6.

4.3.4 Convolution neural network

U-Net [137] is a CNN architecture used for performing image segmentation. It consists of two stages, encoder, and decoder. The encoder stage gradually reduces the spatial dimensions while increasing the feature dimensionality. The decoder stage performs the opposite operation, gradually increasing the spatial dimension while decreasing the feature dimensionality, resulting in an output map of the same spatial size as the input image. Thus the model can be trained to produce a segmentation map as the final output map. NN-UNet [138] is a U-Net [137] based convolution neural network with automatic hyper-parameter computations. It showed successful results in medical decathlon [139] on a wide variety of input data. We choose to use this architecture due to its optimized hyper-parameter computation and successful applicability to a wide variety of tasks. It is a segmentation-based CNN that outputs a probability map signifying the likelihood of each pixel being a lesion or background.

One drawback of this dataset is that each phantom does not have an exact mask signifying the pixels belonging lesion. We only have access to the mean signal information and the lesion coordinates for the signal-present phantoms. We used the mean signal and lesion coordinates to create an approximate mask for each phantom. We were able to train the neural networks using this procedure without needing costly manual annotations. Although a custom CNN designed for this task might have resulted in better performance,

it might not be easily generalizable/applicable for phantoms designed slightly differently. For this reason, we expect the results to be more generalizable by using a standard architecture that works for different types of medical images.

In the encoder, downsampling is performed using strided convolutions; in the decoder, upsampling is performed using transposed convolutions. Each resolution stage in the encoder/decoder consists of two computational blocks, and each computational block consists of convolution followed by instance normalization [140] and leaky-ReLU [141]. For the 2D image modality, 2D architecture is used where the number of channels in the input image is just one slice. For 3D modality, there are two options, 1. 3D full-resolution U-Net, which operates on full-resolution input, 2. 3D U-Net cascade where first coarse segmentation is performed on low-resolution data, which is refined by a 3D U-Net operating on full-resolution data. 3D U-Net cascade is better suited for integrating context. We use 3D full-resolution U-Net.

After separating the training dataset into five folds, five CNN models are trained by choosing four folds for training and the fifth for validation. We train each of the five models for 1000 epochs. Stochastic Gradient Descent optimizer with a learning rate of 0.01 and nestrov momentum of 0.09 are used. The learning rate is controlled using a 'polyLR' learning schedule, where a polynomial function is used to decay the learning rate of each parameter group. After training, the five models are combined to form the ensemble model, which is then used for testing. All four CNN models (2D microcalcification, 2D mass, 3D microcalcification, and 3D mass) are trained using a similar procedure.

4.4 Implementation details

4.4.1 Gabor channels

We created Gabor channels with 8 orientations, 2 phases, and 5 spatial frequencies. The eight orientations are equally spaced apart between 0 to π ie., $N_o = 0, \pi/8, ..., 7\pi/8$. The two phases are orthogonal to each one another, i.e., $N_p = 0, \pi/2$. Five spatial frequencies are used for channel generation, i.e., $N_f = 4, 8, 16, 32, 64$ pixels per cycle. Assuming a monitor distance of 85 cms, equivalent spatial frequency in cycles per degree equals $16, 8, 4, 2, 1$, respectively. Image crops used for training are of size 101 pixels $\times$101 pixels. Using these orientations, phases, and spatial frequencies, a Gabor channel array of size $101 \times 101 \times 80$ with 80 channels is created.

4.4.2 Model observers CHO and FCO

Training the 2D template

Both 2D and 3D templates are of size 101 pixels in width and 101 pixels in height. In terms of depth, the 2D template consists of a single slice, and the 3D template consists of 17 slices, i.e., eight slices on either side of the central slice. The training dataset consists of 576 DBT phantoms, each of size $2048 \times 1792 \times 64$, for each microcalcification, mass, and signal-absent category. For the 2D case, for each signal-present phantom, a crop of size 101×101 is obtained from the signal location on the central slice resulting in an array of size $101 \times 101 \times 576$. For the signal-absent images, ten random crops are obtained from each phantom resulting in an array of size $101 \times 101 \times 5760$. The eighty-channel Gabor array is of size $101 \times 101 \times 80$. Channel response is computed using the Gabor array for both signal-present and signal-absent arrays resulting in response arrays of sizes 80×576 and 80×5760, respectively. The mean signal of size 80×1 is obtained

by computing the means of both these arrays and taking their difference. We compute the covariance matrices for signal-present and signal-absent separately, each of which is of size 80×80, and then obtain the mean covariance matrix by taking the mean of both signal-present and signal-absent covariance matrices. Finally, the template weights are computed according to equation 3 using the mean signal and the pseudo inverse of the mean covariance matrix.

For training the 3D templates, we train the individual 2D templates for all the 17 slices as per the procedure described above. Giving uniform weights to all the templates and combining them would be sub-optimal, and we choose to train the weights according to which the template responses need to be combined.

Training of 3D template with optimized slice weights

We follow a similar procedure to how we trained the 2D template except that here the different 2D templates are similar to the channels present during the generation of the 2D template, i.e., the mean signal is of size $101 \times 101 \times 17$ and the mean covariance matrix is of size 17×17. As a result, for each of the 3D templates, we have 17 2D templates, each of size 101×101 and 17 scalar weights, to combine them optimally. Appendix A.5 shows the visualization of the weights for the four possible cases.

Testing

For both CHO and FCO, Figure 4.3 illustrates the testing procedure for 2D and 3D inputs. We compute the convolution between the template and the input in the frequency domain by computing the Fast Fourier Transform (FFT) of the input phantom and the padded template and multiplying them in the frequency domain. We obtain the final response by performing the inverse FFT of the product of the signal and template in the frequency domain. False positives that occur in the skin tissue and the edge of the

phantom degrade the search performance of CHO and FCO. In order to boost the CHO and FCO performance, we mask these locations by multiplying the response map with a binary map. We use the largest value in the resultant map to represent the phantom.

4.4.3 Convolution neural network

Pre-processing

To aid the training of the segmentation-based CNN, we constructed binary masks representing the signal locations using foreground pixels and the noise locations with background pixels. The dataset only provides the coordinates of the signal location but does not provide the binary masks needed for the training. Using the template of the mean signal, we constructed these masks. Constructing more accurate masks using costly manual human annotations might be possible. Approximate binary masks result in more challenging CNN training with the trade-off of cheap binary mask generation.

Training

We trained four CNN models for the two signal types (microcalcification and mass) and the two modalities (2D and 3D). For the 2D case, 1500 single slice samples of size $2048 \times 793 \times 1$ are used for training, where the training samples are generated using the central slice (C_{slice}), slice before the central slice ($C_{slice}-1$) and slice after the central slice ($C_{slice}+1$) of the 500 3D signal-present training phantoms. For the 3D case, 500 samples of size $380 \times 380 \times 64$ are used for training, where the 380×380 crops are generated from the full spatial size of 2048×793 and a single sample is obtained from each of the 500 training phantoms. The cropping is done to save training time. Uncropped full phantoms were presented as input to the model during testing. This change works because it is a segmentation-based CNN. Each CNN model is an ensemble of five individually trained

CNN models, where each individual CNN is trained by splitting the training dataset into five folds. Each of the individual CNN uses four folds for training and the remaining for validation. We train each of the individual CNN models for 1000 epochs. Each epoch took approximately 240 seconds and 800 seconds for the 2D and 3D, respectively, on a single 12 GB Nvidia GPU. Therefore, training the 2D models approximately takes $240 \times 1000 \times 5 \times 2$ seconds, which translates to about 28 days on a single GPU, accounting for 1000 epochs for each model, 5 folds for each model and 2 models one for microcalcification and mass. Training the 3D models takes approximately 93 days on a single GPU. We used four GPUs to decrease the total training time. We did not make any changes to parameters like the number of epochs to reduce training time in order to maintain the generalization ability.

Post-processing

The presence or absence of the signal cannot be directly determined using the probability map output of the CNN. We use a two-step process of per-pixel thresholding and connected components to make it interpretable. Under per-pixel thresholding, a threshold optimized using the validation set is used to classify each pixel as a foreground (signal) pixel or a background (noise) pixel by comparing the probability value at that pixel against the pre-computed threshold. We compute the connected components using 8 and 26 connectivity, understanding that the presence of the signal will lead to a cluster of foreground pixels. We compute the volume of the largest connected component and use it as the response of the entire phantom, a value that will be higher if the actual signal is present.

Testing

During the testing phase, the input is passed through the ensemble CNN followed by per-pixel thresholding and the computation of the connected components. The per-pixel threshold is computed using a validation set of 50 signal-present and 50 signal-absent phantoms, and it is individually computed for each of the four models, two signal types, and two modalities. Since the CNN output is a probability map, the value of each output pixel varies between 0 and 1. In an ideal scenario, each foreground (signal) pixel will have a value of 1, and each background (noise) pixel will have a value of 0. By reducing the per-pixel threshold in steps of 0.05, the threshold at which the AUC for the validation set is maximized is used as the per-pixel threshold for the test set. The results of the validation set are shown in Appendix A.4. By using 8-connectivity for 2D and 26-connectivity for 3D, we compute all the connected components in the CNN output map. The maximum number of pixels in a single connected component represents this input. No further thresholding of the connected component volume is needed, as Area-under-the-curve (AUC) is used as the figure-of-merit.

4.5 Results

4.5.1 Search Vs LKE

In the search task, the signal is located at a random location in the image. Model observers convolve the image with the signal templates by performing multiplication in the frequency domain, and the phantom response to the template is obtained based on the maximum response location in the convolution output. Search is similar to the task performed by radiologists. Location-known-exactly (LKE) is a more artificial task where the signal-present images are image crops that contain the signal at their center. We

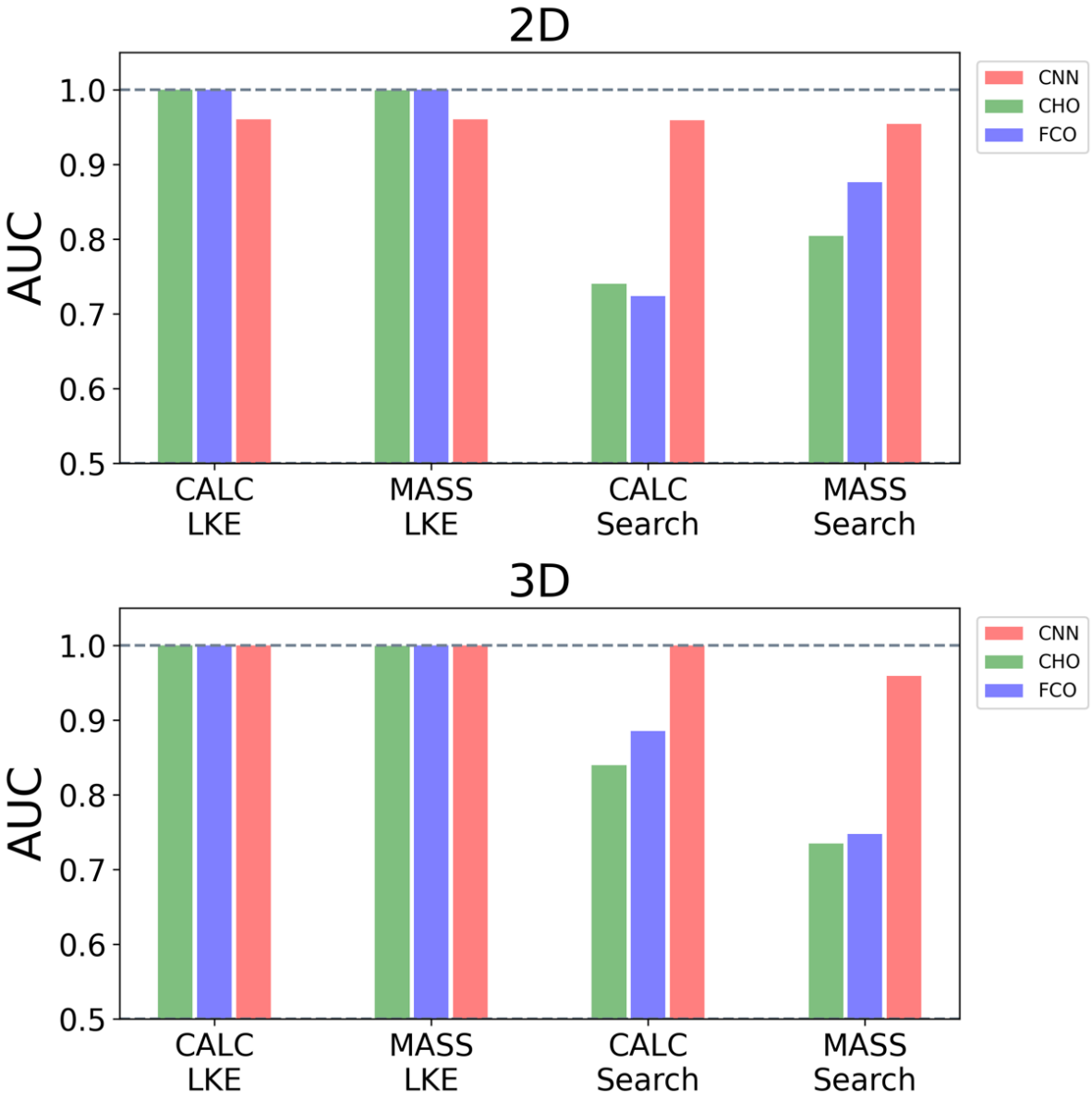

Figure 4.4: **Difficulty of search task:** For both types of signals, the performance of all three model observers is compared for search against LKE. While CNN's performance suffers very little drop from LKE to Search, there is a big drop for both CHO and FCO.

make this comparison to illustrate the increased difficulty of the search task that the radiologists perform against a simple LKE task.

2D case: Top row of Figure 4.4 shows the performance comparison between the

2D LKE and 2D search task. For both types of signal (microcalcification and mass), the performance of the LKE task is close to saturation and outperforms the search task. AUC of the CNN for each type of the signal remains approximately the same across search, and LKE tasks, i.e., a slight drop from LKE to search is observed $\Delta AUC_{\text{LKE vs. Search}} = 0.001$ for microcalcification and $\Delta AUC_{\text{LKE vs. Search}} = 0.006$ for mass. It might suggest that it learned to discard the backgrounds well, and limiting the signal location uncertainty as part of the LKE task does not add any performance benefit. For the CHO and FCO model observers, the search resulted in much lower performance for both signal types than LKE. For microcalcification ($\Delta AUC_{\text{LKE vs. Search}} = 0.26$ for CHO and $\Delta AUC_{\text{LKE vs. Search}} = 0.276$ for FCO) and mass ($\Delta AUC_{\text{LKE vs. Search}} = 0.195$ for CHO and $\Delta AUC_{\text{LKE vs. Search}} = 0.124$ for FCO).

3D case: Bottom row of Figure 4.4 shows the performance comparison between the 3D LKE and 3D search task. For both types of signal (microcalcification and mass), the performance of the LKE task is close to saturation and outperforms the search task. AUC of the CNN for each type of the signal remains the same across search and LKE tasks for microcalcification ($AUC = 1$), and a slight drop from LKE to search is observed for mass ($\Delta AUC_{\text{LKE vs. Search}} = 0.041$). It might suggest that even for the 3D case, the CNN learned to discard the backgrounds well, and limiting the signal location uncertainty as part of the LKE task does not add any performance benefit. For the CHO and FCO model observers, the search resulted in much lower performance for both signal types than LKE. For microcalcification ($\Delta AUC_{\text{LKE vs. Search}} = 0.16$ for CHO and $\Delta AUC_{\text{LKE vs. Search}} = 0.26$ for FCO) and mass ($\Delta AUC_{\text{LKE vs. Search}} = 0.11$ for CHO and $\Delta AUC_{\text{LKE vs. Search}} = 0.25$ for FCO).

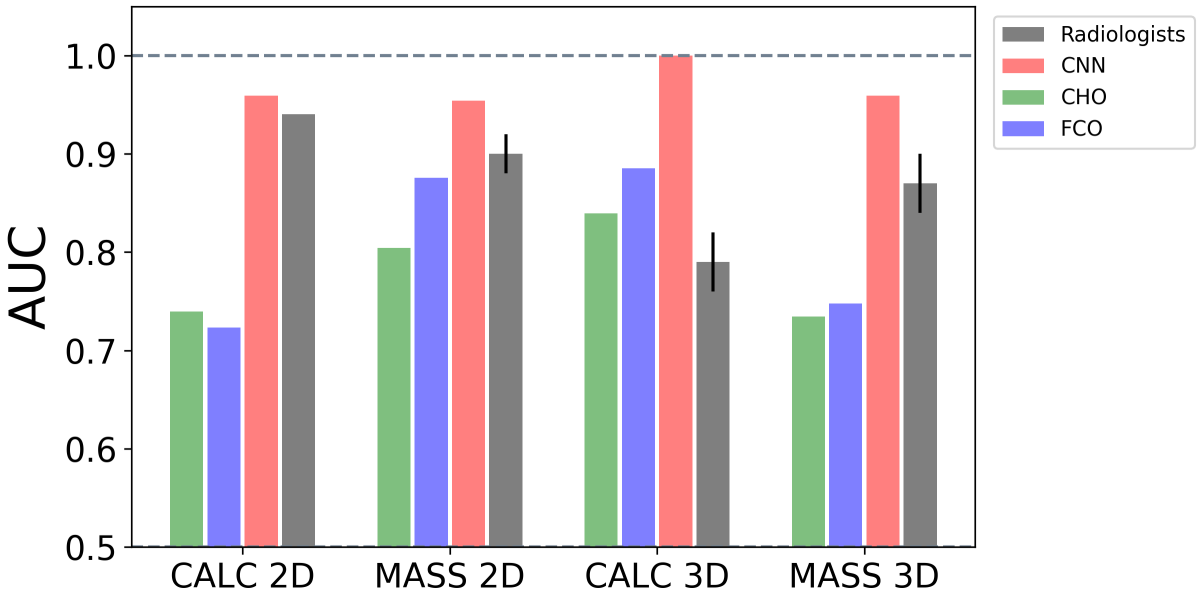

Figure 4.5: **CNN outperforms in 2D and 3D search:** For the microcalcification (CALC) signal, radiologists underperformed in 3D search whereas model observers improved their performance from 2D to 3D. For the mass signal (MASS), where the CNN performance still increased from 2D to 3D, CHO and FCO underperformed in 3D, where a better integration across slices is needed.

4.5.2 2D Vs 3D search

A comparison between 2D and 3D searches for radiologists and model observers is shown in Figure 4.5. **Radiologists:** Radiologists suffer from under-exploration in the 3D search as compared to the 2D search, whereas model observers do not suffer from such a bottleneck. Each radiologist only saw a subset of the phantoms. AUC is computed for each radiologist, and the error bar signifies the performance difference across radiologists. Since the signal template for mass is better visible and spans across multiple slices for the case of 3D, there is a counter-action between better visibility and under-exploration. For the case of microcalcification, radiologists did better for 2D compared to the 3D search ($\Delta AUC_{\text{2D vs. 3D}} = 0.15$), which can be attributed to the under-exploration. For the case of MASS, the performance only slightly drops from 2D to 3D ($\Delta AUC_{\text{2D vs. 3D}} = 0.03$). **CNN:** In a contrasting manner, the performance of CNN increases from 2D to

3D, which can be attributed to a better 3D signal template ($\Delta AUC_{\text{2D vs. 3D}} = 0.041$ for microcalcification and $\Delta AUC_{\text{2D vs. 3D}} = 0.005$ for mass). **CHO and FCO:** Going from 2D to 3D search, the performance of the CHO and FCO increased for CALC and decreased for MASS, which might signify that 3D mass is more confusable with the background for model observers CHO and FCO. For the 2D search, for microcalcification, CHO slightly outperforms FCO ($\Delta AUC_{\text{CHO vs. FCO}} = 0.016$), whereas for mass FCO is performs much better ($\Delta AUC_{\text{CHO vs. FCO}} = -0.07$) compared to the CHO model. For the 3D search, FCO outperforms CHO for both types of signals ($\Delta AUC_{\text{CHO vs. FCO}} = -0.05$ for microcalcification and $\Delta AUC_{\text{CHO vs. FCO}} = -0.01$ for mass).

4.5.3 False +ve analysis

Multiple radiologists searched for microcalcification and mass in each of the 28 signal-absent phantoms. The total number of radiologists varied between 4-8 for each phantom. During their search, eye-tracker recorded the eye-movements of the radiologists as they searched the slice and scrolled through the stack of slices. We used a Gaussian filter of size $45 \times 45 \times 3$ to smoothen the response maps from model observers and time-spent maps from radiologists. Along with the fixation location, we also have access to the information time spent at each fixation location from the eye-tracker. As a result, we can see if the most time-spent locations are related to the hot response locations from the model observer response maps. In order to evaluate the relationship only in the regions where the phantom is present, a binary mask removes all fixations falling outside the actual phantom region. We do this false positive analysis on the signal-absent phantoms as they indicate the locations confused by the observers to be a signal when there is no signal.

Time spent by radiologists in hot response regions

This analysis selects the top 1% locations from the model observer response map. We then get the corresponding time spent by the radiologists for those locations. To normalize across phantoms, we report the percentage of time spent at these locations to the total time spent on this phantom instead of reporting the raw time spent. As a result, the effect of the number of radiologists that saw this phantom is also normalized.

Figure 4.6 shows how much percentage of time radiologists spent in the regions with the highest model observer response. Although radiologists look for either microcalcification or mass in the signal-absent images, the model observers are trained individually for each signal. So, we repeat the analysis for both types of models, and the corresponding results are shown in the two columns. The top row shows the time spent by the radiologists corresponding to the top 1% locations of the model observer response map. For the model observers trained for detecting microcalcification, radiologists fixated longer in the top locations corresponding to the CNN response map than CHO (p = 5.67e-03) and FCO (p = 3.15e-03), but it is not significant. For the model observers trained for detecting mass, radiologists fixated longer at the top locations corresponding to the CNN, and the difference is significant. The difference between CHO and FCO is also significant (P=0.85). For the bottom row, the same analysis is repeated for many different percentages along with 1%. The percentage of time spent reaches 100% as a large percentage of the area is considered. Overall, top locations from CNN coincided with the location where more time was spent by radiologists showing potential similarity between CNN's false positive locations and false positives of the human template.

Therefore, CNN is a significantly better predictor of where humans will spend longer. It is an advantage that can be explored in the future as it adds a new dimension to the usual performance benefit seen in CNNs.

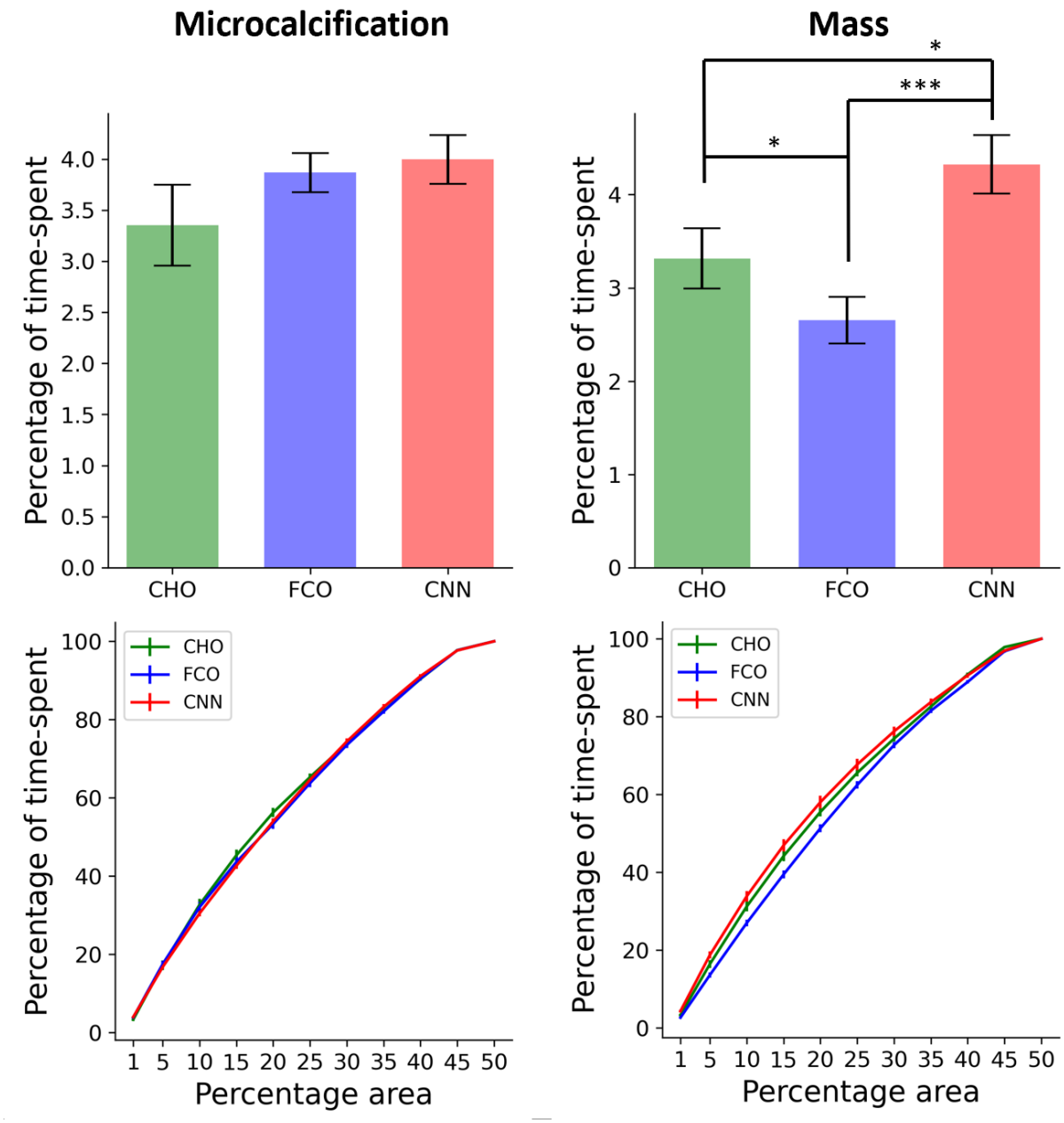

Figure 4.6: **Time-spent by radiologists at top model observer response locations: Top row:** Percentage of time spent by the radiologists corresponding to top 1% locations of the model observer response map. **Bottom row:** Percentage of time spent corresponding to the top locations in the model observer response map. Time-Spent corresponding to top 1%, 5%, 10%, 15%, 20%, 25%, 30%, 35%, 40%, 45%, 50%, locations of the model observer response map

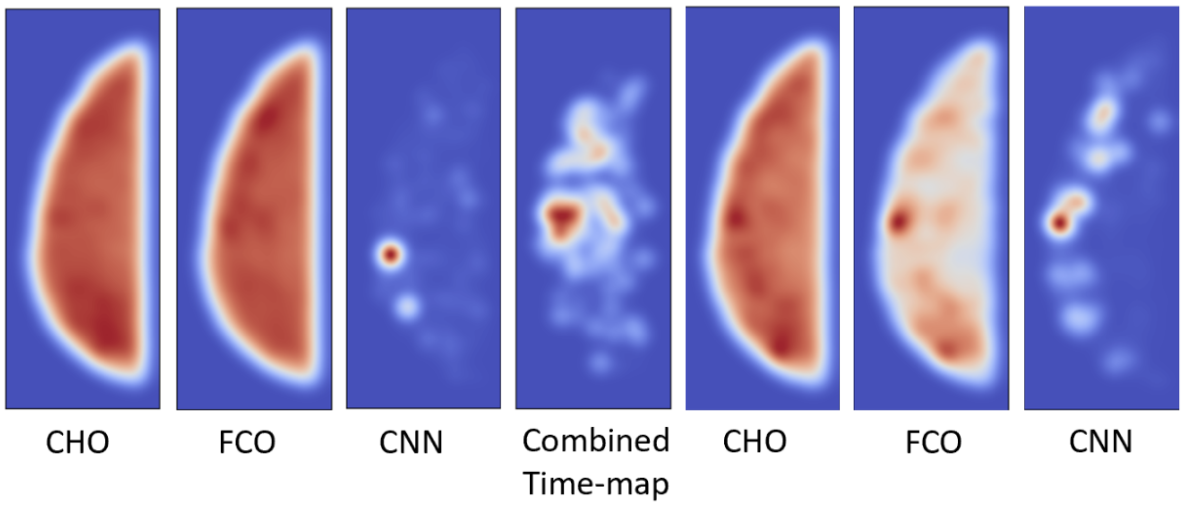

Figure 4.7: **Similarity with average radiologist response:** The combined time spent map and the response maps of the model observers are shown for one slice of a signal-absent phantom (normalized for visualization).

4.6 Discussion

We perform a two-fold analysis to show that a CNN can effectively replace standard model observers like CHO and FCO as an anthropomorphic observer for search tasks.

First, we show that it can do better than humans for both signals and modalities. CHO and FCO do better than radiologists for only one case, i.e., a microcalcification 3D. Human's low performance is explained because of under-exploration, and the lack of such an effect in model observers can explain the good performance from CHO and FCO. Although not reaching human performance, CHO and FCO do well enough for mass 2D but not for the mass in 3D. The 3D mass is more confusable with the background. Also important to note that the effect of under-exploration in humans is less pronounced than in 3D microcalcification, as the mass signal is better visible in the periphery and thus observers can underexplore and still detect the mass. The CHO and FCO also did not replicate human performance for the microcalcification 2D, where the smaller, harder-to-detect microcalcification could contribute to this result. FCO outperforms CHO for

three out of four cases, i.e., except for 2D microcalcification. This is likely related to the fact that the FCO capture the overall shape of the signal better than the sparse channels of the CHO

On the second part, we investigate the predictive power of a CNN-based model observer in predicting the amount of time spent by radiologists. False positive locations for humans are the location that looks similar to the signal. Since these false positive locations look similar to the signal, they tend to spend longer at these locations before moving to a new location. From the model observer's perspective, if a signal-absent location looks similar to the signal, it produces a stronger response. Therefore, if a location looks similar to the target, humans fixate longer at those locations, and model observers produce a stronger response. This is the rationale behind comparing the time spent on maps of radiologists against the response maps of the model observers. This analysis was done in two ways, 1. By picking the top response locations and seeing how much time was spent at those locations. By picking the top 1% locations, we showed that radiologists spent significantly longer at the top locations from CNN. 2. Correlation coefficient can be computed between the two maps to see how similar they are. Using this method, we showed that the correlation is higher for CNN and significantly better than for the FCO. We noticed a correlation between Individual radiologists and the average radiologist map (from the rest of the radiologists who saw this phantom), resulting in a lower correlation potentially explained by the radiologists' under-exploration. We repeated this process both with the microcalcification and mass signals' response maps. The significant results are only present for the mass response maps suggesting that locations looking like mass-signal are more confusing for radiologists, who tend to spend more time at these locations.

4.7 Conclusion

With the advancements in simulation techniques, we can now generate realistic phantoms that are ideal to develop and optimize medical imaging technology without the high costs of clinical trials.. Data-driven methodologies like convolution neural networks, whose modeling does not require signal and background statistics are a viable alternative to the traditional ideal observer. By using general-purpose architectures that can handle different types of data and segmentation-based networks, we can also make this viable IO alternative for signal localization. We also showed the segmentation-based CNN's predictive ability to predict where radiologists will fixate the most.

In the current work, we showed that a segmentation-based CNN could function as an anthropomorphic observer for the search task of DBT phantoms. As part of future work, current work can have three significant directions. First, foveation can be integrated into CNN to show that it will make it an even better anthropomorphic observer. Second, Model generalization can be explored by repeating the current analysis on many datasets. Third, changes to the architecture and training process can be explored to reduce training time without losing generality. With these research directions, it is possible to create faster-trainable easily-generalizable anthropomorphic CNN models.

Chapter 5

Future work

To apply the foveated search transformer model for the task of tumor search, we finetuned the FoveaTer model trained on the ImageNet dataset. We showed promising preliminary results on two-dimensional virtual mammograms.

5.1 Training details

We used the radial-polar pooling regions with a scale of 0.21. During training, three fixations are made by the model, with the initial fixation at a random location in the image. Input to the model is of size 512×512 resulting in a feature map of size 64×64 at the output of the convolution backbone. Two signal types are used: microcalcification and mass, which correspond to smaller and bigger lesions, respectively. Since the input data type is different from the natural images, we finetuned all the parameters instead of only the attention-related parameters as we did for the case of computer mouse search. We analyzed two types of virtual mammograms, 1. Simulated using filtered noise backgrounds, and 2. Digital Breast Tomosynthesis phantoms (DBTs).

DBT Dataset: DBTs are generated using the OpenVCT virtual breast imaging tool

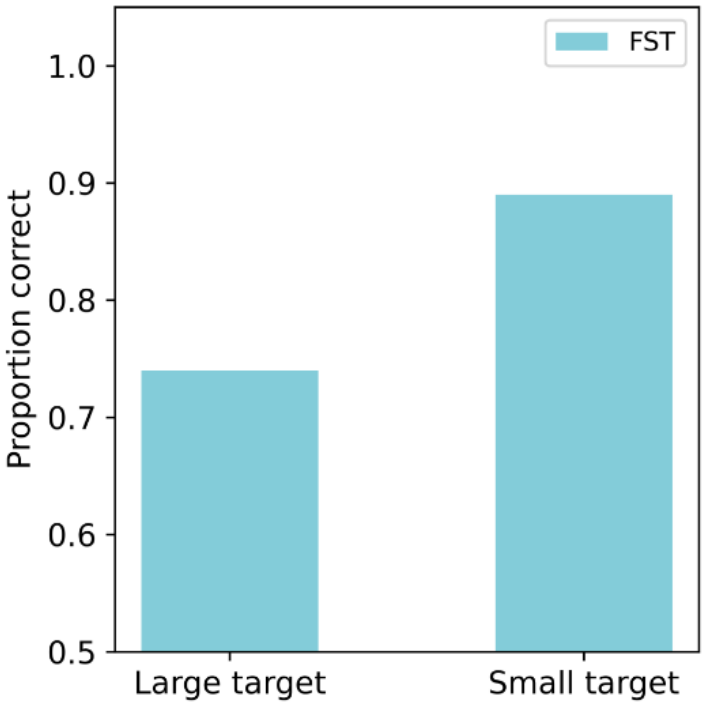

Figure 5.1: **Tumor search in DBT:** Search performance of the model on microcalcification and mass signal types.

from the University of Pennsylvania. For DBTs, we had 950, 100, and 100 phantoms for training, validation, and testing, respectively. All datasets have a 1:1 signal present to absent ratio. Each phantom is a 3D image of size $2048 \times 1792 \times 64$, where 2048 and 1792 are the height and width, respectively, and 64 is the number of slices in each phantom. We selected the central slice of the signal location for the signal-present phantoms, and for the signal-absent phantoms, we selected a random slice. We selected one-quarter of the slice height to give as input to the model.

5.2 Results

During the model training, the model fixated at a random initial fixation. Overall results are shown in Figure 5.1.

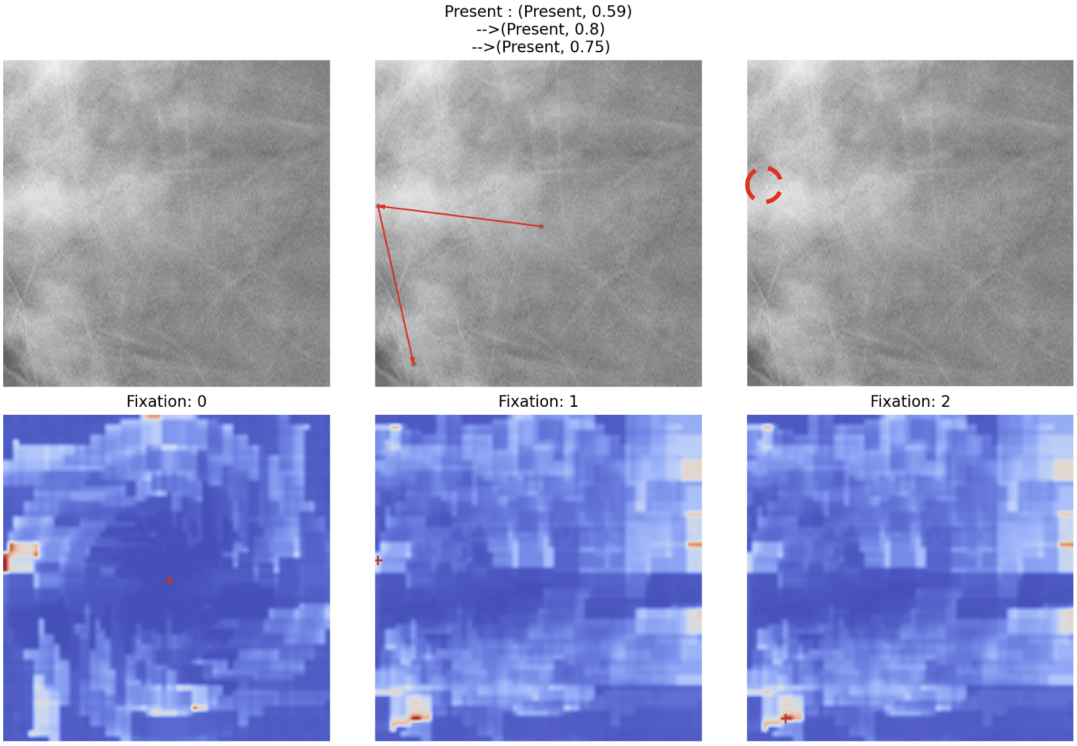

Figure 5.2: Visualization of DBT tumor search for microcalcification

The sample image for microcalcification and mass search is shown in Figure 5.2 and Figure 5.3, respectively.

5.3 Future direction

We intend to update the task of the foveated search model from image-level classification to segmentation. We believe this will help us in the training of the model and also extend the search task to multiple objects. We want to explore the possibility of predicting human perceptual behavior using the search transformer model. This can potentially be done in stages: First, we can replicate the eccentricity performance of the radiologists by learning the scale hyperparameter of the radial-polar pooling region (similar to how we did for the mouse search task). Then we can try to model the visual search patterns of

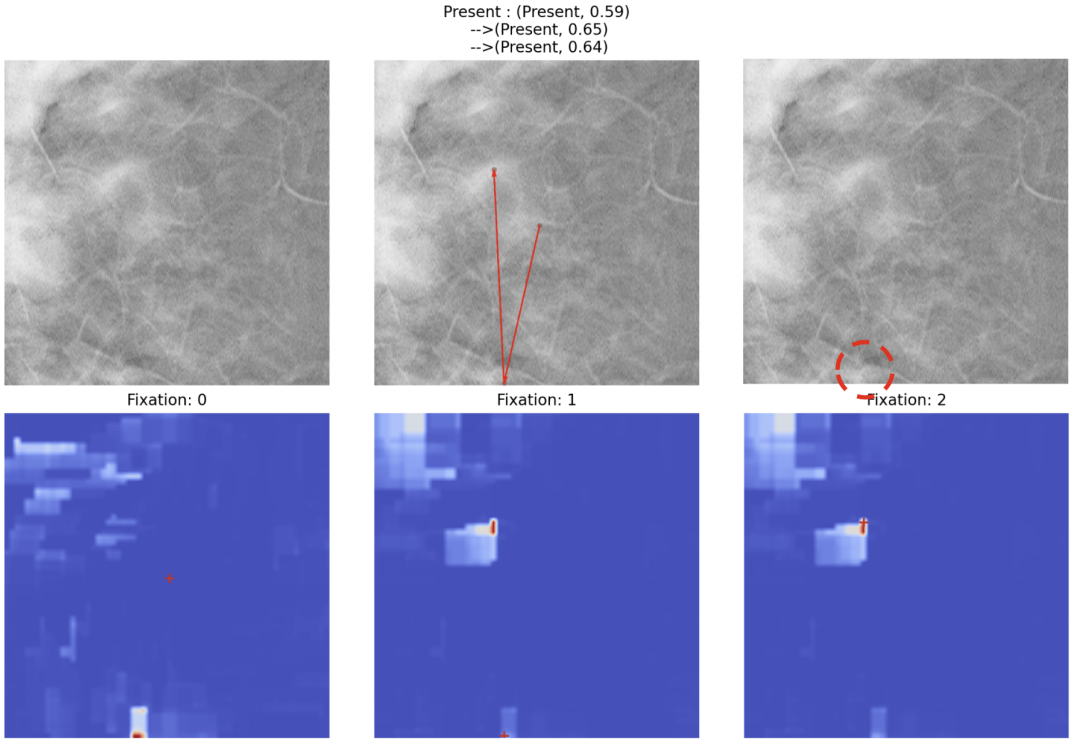

Figure 5.3: Visualization of DBT tumor search for mass

the radiologists, which can help model radiologist behavior in real clinical images, which is impossible with linear model observers.

Appendix A

Appendix

A.1 Alternate model

We present a less biologically plausible Foveated model in this section. With this architecture, the ensemble model can outperform the Baseline model.

DeiT-Small [59]: The DeiT-Small architecture begins with a convolution embedding layer that transforms the $[3, 224, 224]$ input image into a $[384, 14, 14]$ representation whose spacial size is 14×14, followed by a series of twelve transformer blocks, each sized for a 384-dimensional embedding.

Foveated model: Model architecture is shown in Figure A.1. The Foveation module can be plugged-in at any stage of the transformer architecture. The first m transformer layers process full-resolution features, and the last *(N-m)* transformer layers process the pooled features from the foveation module. The input image is first passed through the embedding layer resulting in a feature vector of size $[384, 14, 14]$. After adding the position embedding and flattening the spatial size of the embedding layer output, the resultant full-resolution feature vector of size $[384, 196]$ is passed through the m transformer blocks along with a learnable vector of size 384 values, called a class token. As the

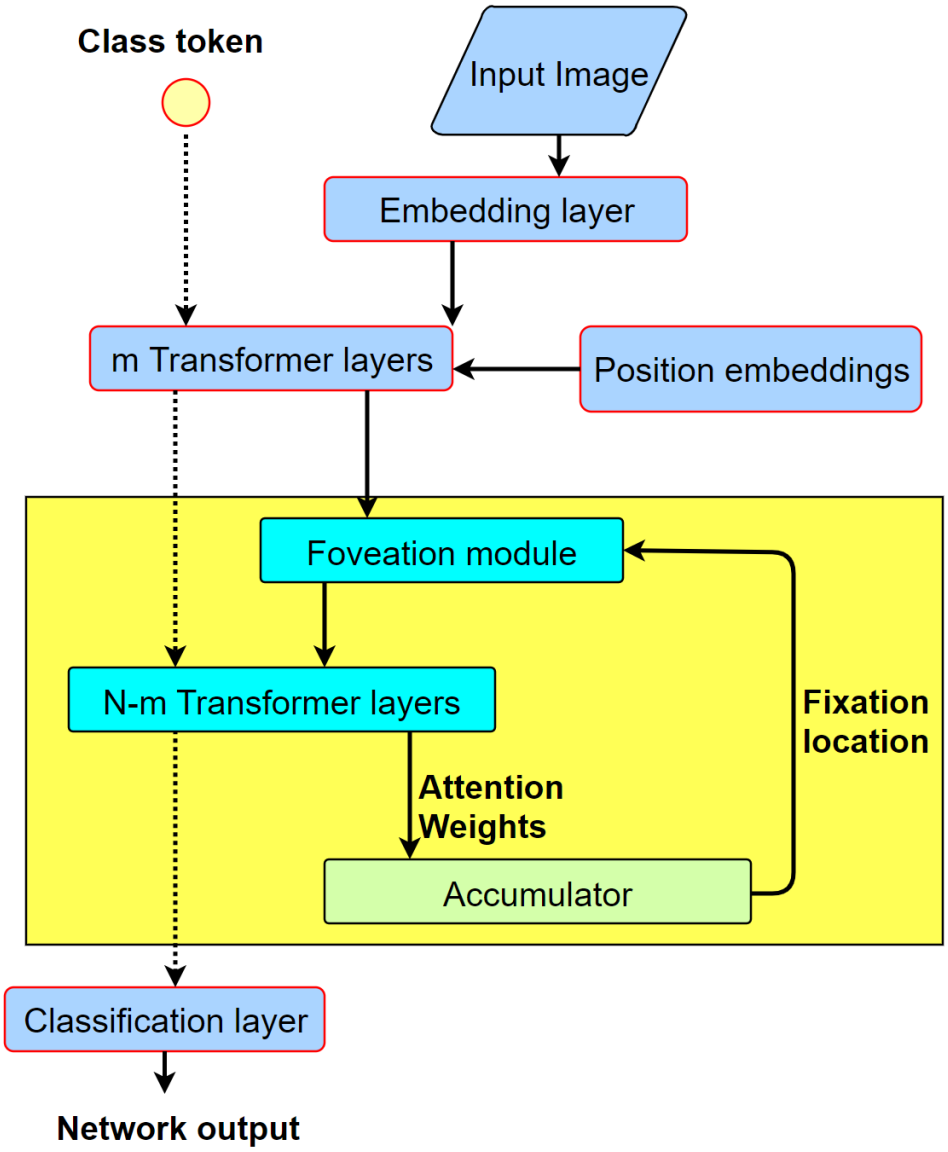

Figure A.1: **FoveaTer architecture:** Solid black arrows denote the flow of image-related features. N is the total number of transformer layers. The foveation module performs fixation-dependent pooling. *Accumulator* uses the attention weights from the last transformer layer of past and present fixations to predict the next fixation location. Model blocks within the yellow region are executed for each fixation.

Condition	Model	Fixations	Type	Throughput	Acc@1
Baseline	DeiT-Small		Baseline	323	79.83
Pooled	DeiT-Small		Pool	592	73.64
	Model	CF-1		564	75.2
Upper bound	Oracle	CF-3	DS	507	80.36
	Oracle	CF-3	Ens	388	84.27
Optimal		CF-3	DS	**489**	78.31
		CF-3	Ens	**348**	**79.99**

Table A.1: Throughput and Accuracy on ImageNet: We compare our models against the baseline model using Top-1 accuracy and Image throughput. (*DS* - Dynamic stop, *Ens* - Ensemble, *Pool* - uniform 5×5 pooling, *CF* - central fixation)

same size is maintained at the input and output of the transformer layer, a feature vector of size $[384, 196]$ is obtained at the input of the Foveation module. Then, we perform fixation-dependent average-pooling using the Foveation module, resulting in features of size $[384, 22]$. Under this non-uniform average-pooling model, locations closer to the fixation location use smaller neighborhoods for pooling than locations far from the fixation location. Pooled features of size $[384, 22]$ along with the class token are passed through the remaining *(N-m)* transformer layers. We use the self-attention weights corresponding to the class token from the last transformer layer to predict the next fixation location. Finally, the classification layer transforms the class token into a logits vector. During training, the total number of fixations is limited to five fixations.

We use a 6-6 configuration, i.e., six transformer layers before the Foveation module and six transformer layers after it. We present the results on the ImageNet dataset in Table A.1. The original full-resolution model is referred to as 'Baseline', which has a throughput of 323 and Top-1 accuracy of 79.83. Since the first level of the pooling region is of size 5×5, we construct a pooled version of the baseline model using 5×5 average-pooling. We compare this with the foveated model with one fixation at the image center, with approximately the same throughput. The foveated model with single fixation

outperforms the pooled baseline model, as shown in row 3. 'Oracle' refers to the model with perfect Dynamic-stop, i.e., it knows the ground truth and stops the model when the prediction matches the ground truth. Since 'Oracle' has the perfect stopping rule, it provides the upper bound on the performance of the Dynamic-stop model. Dynamic-stop and Ensemble performance is computed. Finally, the Foveated model's ensemble model outperforms the Baseline model in terms of throughput and accuracy.

A.2 Scene categories used for Psychophysics experiment

Classes present in the scene classification task,

1. airport terminal
2. amphitheater
3. assembly line
4. bamboo forest
5. banquet hall
6. basement
7. beach
8. boxing ring
9. bus interior
10. canal natural
11. canyon
12. classroom
13. cliff
14. corn field
15. department store

16. desert sand
17. dining room
18. forest path
19. glacier
20. greenhouse indoor
21. gymnasium indoor
22. jail cell
23. museum indoor
24. phone booth
25. railroad track
26. sauna
27. subway station platform
28. water park
29. wind farm
30. zen garden

A.3 Comparison of FoveaTer with existing models

	Luo (2016)	**Reddy (2020)**	**Ours**
Dataset	ImageNet	CIFAR10, ImageNet	ImageNet, Places365 subset
Baseline Architecture	CNN (AlexNet, VGG, GNT)	CNN (ResNet)	Vision Transformer (deit)
Image scaling	Yes	No	No
Adversarial attacks	BFGS, sign method	FGSM, PGD	PGD
Resource usage (N fix)	1x	Retinal - Nx Cortical - 1x	0.8x for 3 fix
Foveation Location	Input image	Input image	can plug-in anywhere

Table A.2: Comparison with existing models

Comparison with existing models, which show the robustness of the foveated systems against adversarial attacks, is demonstrated in Table A.2. Our model is based on Vision transformer architecture compared to the other models on CNN architectures. Our model can also be extended to have a convolution backbone, as shown in the supplementary material. We do not perform any image scaling. Our resource usage is 0.8× that of the full resolution model. We allow the possibility of applying foveation to an intermediate feature map rather than restricting it to be applied only to the input image.

Per-Pixel threshold	CALC 2D	MASS 2D	CALC 3D	MASS 3D
1	0.5	0.985	1	0.90
0.95	0.5	0.9992	**1**	**0.941**
0.9	0.5	0.9988	0.997	0.917
0.85	0.5	0.9992	0.9948	0.8892
0.8	0.5	**0.9992**	0.9948	0.8576
0.75	0.5	0.9968	0.992	
0.7	0.5	0.994	0.9848	
0.65	0.5	0.9942	0.975	
0.6	0.9864	0.9944		
0.55	0.9944	0.9926		
0.5	0.9956	0.9908		
0.45	**0.996**	0.978		
0.4	0.9882	0.9654		

Table A.3: AUC of the validation set

A.4 Computation of Per-Pixel threshold

Four per-pixel thresholds are computed corresponding to the 2D microcalcification, 3D microcalcification, 2D mass and 3D mass models. After reducing the per-pixel threshold in steps of 0.05, the threshold that maximizes the AUC of the validation set is selected as the per-pixel threshold for the validation set. Table A.3 shows the AUC value of the validation set for various per-pixel thresholds, where highest AUC for each model is shown in bold. Once the AUC drops significantly, we no longer need to sweep the threshold for that model.

A.5 Slice weights for the 3D template of CHO and FCO models

The CALC weights of both model observers have a peak in the middle, consistent with the CALC signal decaying faster. However, the MASS, which does not decay as fast, has larger weights even towards the edges of the 3D template. The weights are visualized in Figure A.2.

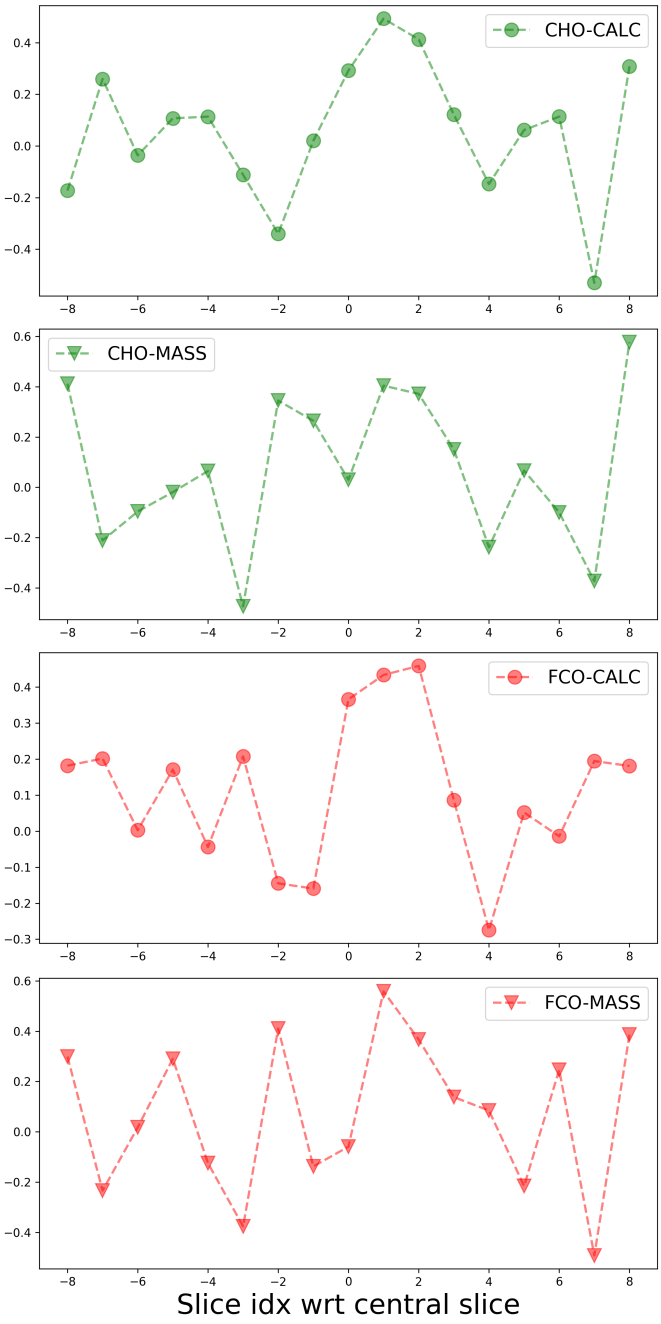

Figure A.2: Slice weights used to train the 3D template of CHO and FCO. Learned weights differ from the uniform weighting of 2D templates corresponding to all slices. The CALC weights of both model observers have a peak in the middle, consistent with the CALC signal decaying faster. However, the MASS, which does not decay as fast, has larger weights even towards the edges of the 3D template.

Bibliography

[1] M. M. Hayhoe and D. H. Ballard, *Modeling task control of eye movements*, *Current Biology* **24** (2014) R622–R628.

[2] M. M. Hayhoe and D. H. Ballard, *Eye movements in natural behavior*, *Trends in Cognitive Sciences* **9** (2005) 188–194.

[3] M. M. Hayhoe, *Vision and action.*, *Annual review of vision science* **3** (2017) 389–413.

[4] G. J. Zelinsky, *A theory of eye movements during target acquisition.*, *Psychological review* **115 4** (2008) 787–835.

[5] J. Najemnik and W. S. Geisler, *Optimal eye movement strategies in visual search*, *Nature* **434** (2005) 387–391.

[6] L. W. Renninger, P. Verghese, and J. M. Coughlan, *Where to look next? eye movements reduce local uncertainty.*, *Journal of vision* **7 3** (2007) 6.

[7] P. Verghese, *Active search for multiple targets is inefficient*, *Vision Research* **74** (2010) 61–71.

[8] G. E. Legge, T. A. Hooven, T. S. Klitz, J. S. Mansfield, and B. S. Tjan, *Mr. chips 2002: new insights from an ideal-observer model of reading*, *Vision Research* **42** (2002) 2219–2234.

[9] N. D. B. Bruce and J. K. Tsotsos, *Saliency, attention, and visual search: an information theoretic approach.*, *Journal of vision* **9 3** (2009) 5.1–24.

[10] A. Torralba, A. Oliva, M. S. Castelhano, and J. 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** (2006) 766–86.

[11] L. Zhang, M. H. Tong, T. K. Marks, H. Shan, and G. Cottrell, *Sun: A bayesian framework for saliency using natural statistics.*, *Journal of vision* **8 7** (2008) 32.1–20.

[12] J. R. Brockmole, D. Z. Hambrick, D. J. Windisch, and J. M. Henderson, *The role of meaning in contextual cueing: Evidence from chess expertise*, *Quarterly Journal of Experimental Psychology* **61** (2008) 1886 – 1896.

[13] S. Baron-Cohen, *Mindblindness: An essay on autism and theory of mind*, 1997.

[14] K. M. Dalton, B. M. Nacewicz, T. Johnstone, H. S. Schaefer, M. A. Gernsbacher, H. H. Goldsmith, A. L. Alexander, and R. J. Davidson, *Gaze fixation and the neural circuitry of face processing in autism*, *Nature Neuroscience* **8** (2005) 519–526.

[15] D. Kliemann, I. Dziobek, A. Hatri, J. Baudewig, and H. R. Heekeren, *The role of the amygdala in atypical gaze on emotional faces in autism spectrum disorders*, *The Journal of Neuroscience* **32** (2012) 9469 – 9476.

[16] E. M. Fine and G. Rubin, *Reading with simulated scotomas: attending to the right is better than attending to the left*, *Vision Research* **39** (1999) 1039–1048.

[17] Y. Tsank and M. P. Eckstein, *Domain specificity of oculomotor learning after changes in sensory processing*, *The Journal of Neuroscience* **37** (2017) 11469 – 11484.

[18] R. A. Schuchard, *Preferred retinal loci and macular scotoma characteristics in patients with age-related macular degeneration.*, *Canadian journal of ophthalmology. Journal canadien d'ophtalmologie* **40 3** (2005) 303–12.

[19] L. Itti and C. Koch, *A saliency-based search mechanism for overt and covert shifts of visual attention*, *Vision Research* **40** (2000) 1489–1506.

[20] J. Harel, C. Koch, and P. Perona, *Graph-based visual saliency*, in *NIPS*, 2006.

[21] K. Koehler, F. Guo, S. Zhang, and M. P. Eckstein, *What do saliency models predict?*, *Journal of vision* **14 3** (2014) 14.

[22] J. M. Henderson and T. R. Hayes, *Meaning-based guidance of attention in scenes as revealed by meaning maps*, *Nature Human Behaviour* **1** (2017) 743–747.

[23] J. Najemnik and W. S. Geisler, *Eye movement statistics in humans are consistent with an optimal search strategy.*, *Journal of vision* **8 3** (2008) 4.1–14.

[24] M. P. Eckstein, W. A. Schoonveld, S. Zhang, S. C. Mack, and E. Akbas, *Optimal and human eye movements to clustered low value cues to increase decision rewards during search*, *Vision Research* **113** (2015) 137–154.

[25] S. Zhang and M. P. Eckstein, *Evolution and optimality of similar neural mechanisms for perception and action during search*, *PLoS Computational Biology* **6** (2010).

[26] J. F. Ackermann and M. S. Landy, *Choice of saccade endpoint under risk.*, *Journal of vision* **13 3** (2013).

[27] D. Hoppe and C. A. Rothkopf, *Multi-step planning of eye movements in visual search*, *Scientific Reports* **9** (2019).

[28] M. F. Peterson and M. P. Eckstein, *Looking just below the eyes is optimal across face recognition tasks*, *Proceedings of the National Academy of Sciences* **109** (2012) E3314 – E3323.

[29] M. Zhang, J. Feng, K. T. Ma, J. H. Lim, Q. Zhao, and G. Kreiman, *Finding any waldo with zero-shot invariant and efficient visual search*, *Nature Communications* **9** (2018).

[30] J. Redmon and A. Farhadi, *Yolo9000: Better, faster, stronger*, *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016) 6517–6525.

[31] S. Ren, K. He, R. B. Girshick, and J. Sun, *Faster r-cnn: Towards real-time object detection with region proposal networks*, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **39** (2015) 1137–1149.

[32] M. P. Eckstein, K. Koehler, L. E. Welbourne, and E. Akbas, *Humans, but not deep neural networks, often miss giant targets in scenes*, *Current Biology* **27** (2017) 2827–2832.e3.

[33] P. F. Felzenszwalb, R. B. Girshick, D. A. McAllester, and D. Ramanan, *Object detection with discriminatively trained part based models*, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **32** (2010) 1627–1645.

[34] E. Akbas and M. P. Eckstein, *Object detection through search with a foveated visual system*, *PLoS Computational Biology* **13** (2014).

[35] M. P. Eckstein, B. A. Drescher, and S. S. Shimozaki, *Attentional cues in real scenes, saccadic targeting, and bayesian priors.*, *Psychological science* **17 11** (2005) 973–80.

[36] K. Koehler and M. P. Eckstein, *Beyond scene gist: Objects guide search more than scene background*, *Journal of Experimental Psychology: Human Perception and Performance* **43** (2017) 1177–1193.

[37] J. M. Wolfe, M. L.-H. Võ, K. K. Evans, and M. R. Greene, *Visual search in scenes involves selective and nonselective pathways*, *Trends in Cognitive Sciences* **15** (2011) 77–84.

[38] M. L.-H. Võ and J. M. Henderson, *Does gravity matter? effects of semantic and syntactic inconsistencies on the allocation of attention during scene perception.*, *Journal of vision* **9 3** (2009) 24.1–15.

[39] M. L.-H. Võ, S. E. P. Boettcher, and D. Draschkow, *Reading scenes: how scene grammar guides attention and aids perception in real-world environments.*, *Current opinion in psychology* **29** (2019) 205–210.

[40] M. L.-H. Võ, *The meaning and structure of scenes*, *Vision Research* **181** (2021) 10–20.

[41] M. B. Neider and G. J. Zelinsky, *Scene context guides eye movements during visual search*, *Vision Research* **46** (2006) 614–621.

[42] M. Kümmerer, T. S. A. Wallis, and M. Bethge, *Deepgaze ii: Reading fixations from deep features trained on object recognition*, *ArXiv* **abs/1610.01563** (2016).

[43] M. Kümmerer, M. Bethge, and T. S. A. Wallis, *Deepgaze iii: Modeling free-viewing human scanpaths with deep learning*, *Journal of Vision* **22** (2022).

[44] M. Land, *Oculomotor behaviour in vertebrates and invertebrates*, *The Oxford Handbook of Eye Movements* (01, 2012).

[45] N. Marshall, M. Land, and T. Cronin, *Shrimps that pay attention: Saccadic eye movements in stomatopod crustaceans*, *Philosophical transactions of the Royal Society of London. Series B, Biological sciences* **369** (02, 2014) 20130042.

[46] M. Hayhoe and D. Ballard, *Eye movements in natural behavior*, *Trends in cognitive sciences* **9** (April, 2005) 188—194.

[47] H. Strasburger, I. Rentschler, and M. Jüttner, *Peripheral vision and pattern recognition: A review*, *Journal of vision* **11** (05, 2011) 13.

[48] C. J. H. Ludwig, J. R. Davies, and M. P. Eckstein, *Foveal analysis and peripheral selection during active visual sampling*, *Proceedings of the National Academy of Sciences* **111** (2014), no. 2 E291–E299, [https://www.pnas.org/content/111/2/E291.full.pdf].

[49] H. Yamamoto, Y. Yeshurun, and M. Levine, *An active foveated vision system: Attentional mechanisms and scan path covergence measures*, *Comput. Vis. Image Underst.* **63** (1996) 50–65.

[50] S. Prince, J. H. Elder, Y. Hou, M. Sizintsev, and Y. Olevskiy, *Statistical cue integration for foveated wide-field surveillance*, *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* **2** (2005) 603–610 vol. 2.

[51] R. Girshick, J. Donahue, T. Darrell, and J. Malik, *Rich feature hierarchies for accurate object detection and semantic segmentation*, in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580–587, 2014.

[52] S. Ren, K. He, R. B. Girshick, and J. Sun, *Faster r-cnn: Towards real-time object detection with region proposal networks*, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **39** (2015) 1137–1149.

[53] V. Mnih, N. Heess, A. Graves, and K. Kavukcuoglu, *Recurrent models of visual attention*, in *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, (Cambridge, MA, USA), p. 2204–2212, MIT Press, 2014.

[54] E. Akbas and M. P. Eckstein, *Object detection through search with a foveated visual system*, *PLOS Computational Biology* **13** (10, 2017) 1–28.

[55] Y. Luo, X. Boix, G. Roig, T. A. Poggio, and Q. Zhao, *Foveation-based mechanisms alleviate adversarial examples*, *ArXiv* **abs/1511.06292** (2015).

[56] A. Deza and T. Konkle, *Emergent properties of foveated perceptual systems*, *ArXiv* **abs/2006.07991** (2020).

[57] T. Kiritani and K. Ono, *Recurrent attention model with log-polar mapping is robust against adversarial attacks*, *ArXiv* **abs/2002.05388** (2020).

[58] M. R. Vuyyuru, A. Banburski, N. Pant, and T. Poggio, *Biologically inspired mechanisms for adversarial robustness*, in *Advances in Neural Information Processing Systems* (H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, eds.), vol. 33, pp. 2135–2146, Curran Associates, Inc., 2020.

[59] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, *Training data-efficient image transformers & distillation through attention*, *arXiv preprint arXiv:2012.12877* (2020).

[60] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, *An image is worth 16x16 words: Transformers for image recognition at scale*, *arXiv preprint arXiv:2010.11929* (2020).

[61] R. Shao, Z. Shi, J. Yi, P.-Y. Chen, and C.-J. Hsieh, *On the adversarial robustness of visual transformers*, *ArXiv* **abs/2103.15670** (2021).

[62] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, u. Kaiser, and I. Polosukhin, *Attention is all you need*, in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, (Red Hook, NY, USA), p. 6000–6010, Curran Associates Inc., 2017.

[63] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, *Imagenet: A large-scale hierarchical image database*, in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

[64] C. Sun, A. Shrivastava, S. Singh, and A. Gupta, *Revisiting unreasonable effectiveness of data in deep learning era*, in *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 843–852, 2017.

[65] H. Larochelle and G. E. Hinton, *Learning to combine foveal glimpses with a third-order boltzmann machine*, in *Advances in Neural Information Processing Systems* (J. Lafferty, C. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta, eds.), vol. 23, Curran Associates, Inc., 2010.

[66] J. Ba, V. Mnih, and K. Kavukcuoglu, *Multiple object recognition with visual attention*, in *ICLR (Poster)*, 2015.

[67] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, *Show, attend and tell: Neural image caption generation with visual attention*, in *Proceedings of the 32nd International Conference on Machine Learning* (F. Bach and D. Blei, eds.), vol. 37 of *Proceedings of Machine Learning Research*, (Lille, France), pp. 2048–2057, PMLR, 07–09 Jul, 2015.

[68] B. Alsallakh, N. Kokhlikyan, V. Miglani, J. Yuan, and O. Reblitz-Richardson, *Mind the pad – {cnn}s can develop blind spots*, in *International Conference on Learning Representations*, 2021.

[69] J. K. Tsotsos, *A computational perspective on visual attention*, 2011.

[70] R. Pascanu, T. Mikolov, and Y. Bengio, *On the difficulty of training recurrent neural networks*, in *ICML*, 2013.

[71] A. Oliva and A. Torralba, *Modeling the shape of the scene: A holistic representation of the spatial envelope*, *International Journal of Computer Vision* **42** (May, 2001) 145–175.

[72] R. Rosenholtz, J. Huang, and K. Ehinger, *Rethinking the role of top-down attention in vision: Effects attributable to a lossy representation in peripheral vision*, *Frontiers in Psychology* **3** (2012) 13.

[73] C. Koch and S. Ullman, *Shifts in Selective Visual Attention: Towards the Underlying Neural Circuitry*, pp. 115–141. Springer Netherlands, Dordrecht, 1987.

[74] 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** (1998), no. 11 1254–1259.

[75] L. Itti and C. Koch, *A saliency-based search mechanism for overt and covert shifts of visual attention*, *Vision Research* **40** (2000), no. 10 1489–1506.

[76] A. Torralba, A. Oliva, M. Castelhano, and J. Henderson, *Contextual guidance of eye movements and attention in real-world scenes: The role of global features in object search.*, *Psychological review* **113** (11, 2006) 766–86.

[77] C. Wloka, I. Kotseruba, and J. K. Tsotsos, *Active fixation control to predict saccade sequences*, in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3184–3193, 2018.

[78] G. J. Zelinsky, S. Ahn, Y. Chen, Z. Yang, H. Adeli, L. Huang, D. Samaras, and M. Hoai, *Predicting goal-directed attention control using inverse-reinforcement learning.*, *Neurons, behavior, data analysis and theory* **2021** (2021).

[79] J. Freeman and E. P. Simoncelli, *Metamers of the ventral stream*, *Nature neuroscience* **14** (2011) 1195 – 1201.

[80] M. Everingham, S. M. A. Eslami, L. V. Gool, C. K. I. Williams, J. M. Winn, and A. Zisserman, *The pascal visual object classes challenge: A retrospective*, *International Journal of Computer Vision* **111** (2014) 98–136.

[81] M. Jaderberg, K. Simonyan, A. Zisserman, and k. kavukcuoglu, *Spatial transformer networks*, in *Advances in Neural Information Processing Systems* (C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, eds.), vol. 28, Curran Associates, Inc., 2015.

[82] A. Krizhevsky, *Learning multiple layers of features from tiny images*, 2009.

[83] E. W. A. Harris, M. Niranjan, and J. Hare, *Foveated convolutions: improving spatial transformer networks by modelling the retina*, in *Shared Visual Representations in Human and Machine Intelligence: 2019 NeurIPS Workshop*, December, 2019.

[84] G. Dabane, L. Perrinet, and E. Daucé, *What You See Is What You Transform: Foveated Spatial Transformers as a bio-inspired attention mechanism*, .

[85] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, *Gradient-based learning applied to document recognition*, *Proc. IEEE* **86** (1998) 2278–2324.

[86] A. Mnih and K. Gregor, *Neural Variational Inference and Learning in Belief Networks*, in *Proceedings of ICML*, 2014.

[87] A. Kurakin, I. J. Goodfellow, and S. Bengio, *Adversarial examples in the physical world*, *ArXiv* **abs/1607.02533** (2017).

[88] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, *Towards deep learning models resistant to adversarial attacks*, *ArXiv* **abs/1706.06083** (2018).

[89] K. R. Dukewich and R. M. Klein, *Inhibition of return: A phenomenon in search of a definition and a theoretical framework*, *Attention, Perception, & Psychophysics* **77** (Jul, 2015) 1647–1658.

[90] H. Touvron, M. Cord, A. El-Nouby, P. Bojanowski, A. Joulin, G. Synnaeve, J. Verbeek, and H. J'egou, *Augmenting convolutional networks with attention-based aggregation*, *arXiv preprint arXiv:2112.13692* (2021).

[91] K. He, X. Zhang, S. Ren, and J. Sun, *Deep residual learning for image recognition*, *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016) 770–778.

[92] D. Kingma and J. Ba, *Adam: A method for stochastic optimization*, *International Conference on Learning Representations* (12, 2014).

[93] I. Loshchilov and F. Hutter, *Decoupled weight decay regularization*, in *ICLR*, 2019.

[94] N. Papernot, F. Faghri, N. Carlini, I. Goodfellow, R. Feinman, A. Kurakin, C. Xie, Y. Sharma, T. Brown, A. Roy, A. Matyasko, V. Behzadan, K. Hambardzumyan, Z. Zhang, Y.-L. Juang, Z. Li, R. Sheatsley, A. Garg, J. Uesato, W. Gierke, Y. Dong, D. Berthelot, P. Hendricks, J. Rauber, and R. Long, *Technical report on the cleverhans v2.1.0 adversarial examples library*, *arXiv preprint arXiv:1610.00768* (2018).

[95] B. Zhou, À. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, *Places: A 10 million image database for scene recognition*, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **40** (2018) 1452–1464.

[96] R. Geirhos, K. Meding, and F. Wichmann, *Beyond accuracy: quantifying trial-by-trial behaviour of cnns and humans by measuring error consistency*, *ArXiv* **abs/2006.16736** (2020).

[97] C. A. Curcio, K. R. Sloan, O. S. Packer, A. Hendrickson, and R. E. Kalina, *Distribution of cones in human and monkey retina: individual variability and radial asymmetry.*, *Science* **236 4801** (1987) 579–82.

[98] P. R. Martin and U. Grünert, *Spatial density and immunoreactivity of bipolar cells in the macaque monkey retina*, *Journal of Comparative Neurology* **323** (1992).

[99] R. O. Duncan and G. M. Boynton, *Cortical magnification within human primary visual cortex correlates with acuity thresholds*, *Neuron* **38** (2003) 659–671.

[100] C.-Y. Chen, K.-P. Hoffmann, C. Distler, and Z. M. Hafed, *The foveal visual representation of the primate superior colliculus*, *Current Biology* **29** (2017) 2109–2119.e7.

[101] J. Freeman and E. P. Simoncelli, *Metamers of the ventral stream*, *Nature neuroscience* **14** (2011) 1195 – 1201.

[102] A. Deza and M. P. Eckstein, *Can peripheral representations improve clutter metrics on complex scenes?*, in *NIPS*, 2016.

[103] R. Rosenholtz, *Capabilities and limitations of peripheral vision.*, *Annual review of vision science* **2** (2016) 437–457.

[104] X. Chen and G. J. Zelinsky, *Real-world visual search is dominated by top-down guidance*, *Vision Research* **46** (2006) 4118–4133.

[105] M. S. Castelhano and C. Heaven, *Scene context influences without scene gist: Eye movements guided by spatial associations in visual search*, *Psychonomic Bulletin & Review* **18** (2011) 890–896.

[106] M. P. Eckstein, *Probabilistic computations for attention, eye movements, and search.*, *Annual review of vision science* **3** (2017) 319–342.

[107] H. H. Barrett, *Model observers for assessment of image quality*, *2002 IEEE Nuclear Science Symposium Conference Record* **2** (1993) 652 vol.2–.

[108] Y. Zhang, B. Pham, and M. P. Eckstein, *Evaluation of jpeg 2000 encoder options: human and model observer detection of variable signals in x-ray coronary angiograms*, *IEEE Transactions on Medical Imaging* **23** (2004) 613–632.

[109] J. P. Rolland and H. H. Barrett, *Effect of random background inhomogeneity on observer detection performance.*, *Journal of the Optical Society of America. A, Optics and image science* **9 5** (1992) 649–58.

[110] C. Castella, M. P. Eckstein, C. K. Abbey, K. Kinkel, F. R. Verdun, R. S. Saunders, E. Samei, and F. O. Bochud, *Mass detection on mammograms: influence of signal shape uncertainty on human and model observers.*, *Journal of the Optical Society of America. A, Optics, image science, and vision* **26 2** (2009) 425–36.

[111] C. P. Favazza, K. A. Fetterly, N. J. Hangiandreou, S. Leng, and B. A. Schueler, *Implementation of a channelized hotelling observer model to assess image quality of x-ray angiography systems*, *Journal of Medical Imaging* **2** (2015).

[112] L. Yu, S. Leng, L. Chen, J. M. Kofler, R. E. Carter, and C. H. McCollough, *Prediction of human observer performance in a 2-alternative forced choice low-contrast detection task using channelized hotelling observer: impact of radiation dose and reconstruction algorithms.*, *Medical physics* **40 4** (2013) 041908.

[113] C. K. Abbey and M. P. Eckstein, *Classification images for detection, contrast discrimination, and identification tasks with a common ideal observer.*, *Journal of vision* **6 4** (2006) 335–55.

[114] H. B. Barlow and B. C. Reeves, *The versatility and absolute efficiency of detecting mirror symmetry in random dot displays*, *Vision Research* **19** (1979) 783–793.

[115] W. S. Geisler, *Contributions of ideal observer theory to vision research*, *Vision Research* **51** (2011) 771–781.

[116] W. L. Braje, B. S. Tjan, and G. E. Legge, *Human efficiency for recognizing and detecting low-pass filtered objects*, *Vision Research* **35** (1995) 2955–2966.

[117] M. P. Eckstein, J. S. Whiting, and J. P. Thomas, *Detection and contrast discrimination of moving signals in uncorrelated gaussian noise*, in *Medical Imaging*, 1996.

[118] B. D. Gallas and H. H. Barrett, *Validating the use of channels to estimate the ideal linear observer.*, *Journal of the Optical Society of America. A, Optics, image science, and vision* **20 9** (2003) 1725–38.

[119] A. E. Burgess, X. Li, and C. K. Abbey, *Visual signal detectability with two noise components: anomalous masking effects.*, *Journal of the Optical Society of America. A, Optics, image science, and vision* **14 9** (1997) 2420–42.

[120] M. A. Kupinski, D. C. Edwards, M. L. Giger, and C. E. Metz, *Ideal observer approximation using bayesian classification neural networks*, *IEEE Transactions on Medical Imaging* **20** (2001) 886–899.

[121] K. J. Myers, M. P. Anderson, D. G. Brown, R. F. Wagner, and K. M. Hanson, *Neural network performance for binary discrimination tasks. part ii: effect of task, training, and feature preselection*, in *Medical Imaging*, 1995.

[122] W. Zhou and M. A. Anastasio, *Learning the ideal observer for ske detection tasks by use of convolutional neural networks (cum laude poster award)*, in *Medical Imaging*, 2018.

[123] W. Zhou and M. A. Anastasio, *Learning the ideal observer for joint detection and localization tasks by use of convolutional neural networks*, in *Medical Imaging*, 2019.

[124] B. Kim, M. Han, and J. Baek, *A convolutional neural network-based anthropomorphic model observer for signal detection in breast ct images without human-labeled data*, *IEEE Access* **8** (2020) 162122–162131.

[125] S. Sengupta, C. K. Abbey, K. Li, and M. A. Anastasio, *Investigation of adversarial robust training for establishing interpretable cnn-based numerical observers*, in *Medical Imaging*, 2022.

[126] I. Lorente, C. K. Abbey, and J. G. Brankov, *Cnn based anthropomorphic model observer for defect localization*, in *Medical Imaging*, 2021.

[127] A. Singh, S. Sengupta, and V. Lakshminarayanan, *Explainable deep learning models in medical image analysis*, *Journal of Imaging* **6** (2020).

[128] Z. Salahuddin, H. Woodruff, A. Chatterjee, and P. Lambin, *Transparency of deep neural networks for medical image analysis: A review of interpretability methods*, *Computers in biology and medicine* **140** (2021) 105111.

[129] Z. Rguibi, A. Hajami, D. Zitouni, A. Elqaraoui, and A. Bedraoui, *Cxai: Explaining convolutional neural networks for medical imaging diagnostic*, *Electronics* (2022).

[130] F. Isensee, P. F. Jaeger, S. A. A. Kohl, J. Petersen, and K. Maier-Hein, *nnu-net: a self-configuring method for deep learning-based biomedical image segmentation*, *Nature Methods* **18** (2020) 203 – 211.

[131] O. Ronneberger, P. Fischer, and T. Brox, *U-net: Convolutional networks for biomedical image segmentation*, *ArXiv* **abs/1505.04597** (2015).

[132] I. Díaz, C. K. Abbey, P. Timberg, M. P. Eckstein, F. R. Verdun, C. Castella, and F. O. Bochud, *Derivation of an observer model adapted to irregular signals based on convolution channels*, *IEEE Transactions on Medical Imaging* **34** (2015) 1428–1435.

[133] D. D. Pokrajac, A. D. A. Maidment, and P. R. Bakic, *Optimized generation of high resolution breast anthropomorphic software phantoms.*, *Medical physics* **39 4** (2012) 2290–302.

[134] P. R. Bakic, D. D. Pokrajac, and A. D. A. Maidment, *Computer simulation of the breast subcutaneous and retromammary tissue for use in virtual clinical trials*, in *Medical Imaging*, 2017.

[135] P. R. Bakic, B. Barufaldi, D. Higginbotham, S. P. Weinstein, A. N. Avanaki, K. S. Espig, A. Xthona, T. Kimpe, and A. D. A. Maidment, *Virtual clinical trial of lesion detection in digital mammography and digital breast tomosynthesis*, in *Medical Imaging*, 2018.

[136] M. Kleiner, D. H. Brainard, and D. Pelli, *What's new in psychtoolbox-3?*, *Perception* **36** (2007) 1–16.

[137] O. Ronneberger, P. Fischer, and T. Brox, *U-net: Convolutional networks for biomedical image segmentation*, *ArXiv* **abs/1505.04597** (2015).

[138] F. Isensee, J. Petersen, S. A. A. Kohl, P. F. Jäger, and K. Maier-Hein, *nnu-net: Breaking the spell on successful medical image segmentation*, *ArXiv* **abs/1904.08128** (2019).

[139] A. L. Simpson, M. Antonelli, S. Bakas, M. Bilello, K. Farahani, B. van Ginneken, A. Kopp-Schneider, B. A. Landman, G. J. S. Litjens, B. H. Menze, O. Ronneberger, R. M. Summers, P. Bilic, P. F. Christ, R. K. G. Do, M. J. Gollub, J. Golia-Pernicka, S. Heckers, W. R. Jarnagin, M. McHugo, S. Napel, E. Vorontsov, L. Maier-Hein, and M. J. Cardoso, *A large annotated medical image dataset for the development and evaluation of segmentation algorithms*, *ArXiv* **abs/1902.09063** (2019).

[140] D. Ulyanov, A. Vedaldi, and V. S. Lempitsky, *Instance normalization: The missing ingredient for fast stylization*, *ArXiv* **abs/1607.08022** (2016).

[141] A. L. Maas, *Rectifier nonlinearities improve neural network acoustic models*, 2013.