Learning to search for and detect objects in foveal images using deep learning

Beatriz Paula[1][0000−0001−6153−7838] and Plinio Moreno[1,2][0000−0002−0496−2050]

[1] Instituto Superior Técnico, Univ. Lisboa, 1049-001 Lisboa, Portugal
bia.paula11@gmail.com
[2] Institute for Systems and Robotics (ISR/IST), LARSyS, 1049-001, Lisbon, Portugal
plinio@isr.tecnico.ulisboa.pt

Abstract. The human visual system processes images with varied degrees of resolution, with the fovea, a small portion of the retina, capturing the highest acuity region, which gradually declines toward the field of view's periphery. However, the majority of existing object localization methods rely on images acquired by image sensors with space-invariant resolution, ignoring biological attention mechanisms. As a region of interest pooling, this study employs a fixation prediction model that emulates human objective-guided attention of searching for a given class in an image. The foveated pictures at each fixation point are then classified to determine whether the target is present or absent in the scene. Throughout this two-stage pipeline method, we investigate the varying results obtained by utilizing high-level or panoptic features and provide a ground-truth label function for fixation sequences that is smoother, considering in a better way the spatial structure of the problem. Additionally, we present a novel dual task model capable of performing fixation prediction and detection simultaneously, allowing knowledge transfer between the two tasks. We conclude that, due to the complementary nature of both tasks, the training process benefited from the sharing of knowledge, resulting in an improvement in performance when compared to the previous approach's baseline scores.

Keywords: Visual Search · Foveal Vision · Deep Learning

1 Introduction

A fundamental difference between the human visual system and current approaches to object search is the acuity of the image being processed [1]. The human eye captures an image with very high resolution in the fovea, a small region of the retina, and a decrease in sampling resolution towards the periphery of the field of view. This biological mechanism is crucial for the real-time image processing of the rich data that reaches the eyes (0.1-1 Gbits), since visual attention prioritizes interesting and visually distinctive areas of the scene, known as salient regions, and directs the gaze of the eyes. In contrast, image sensors, by default, are designed to capture the world with equiresolution in a homogeneous space invariant lattice [2], and current solutions to vision system performance

rely on the increase of the number of pixels. This limits real-time applications due to the processing bottleneck and the excessive amount of energy needed by state-of-the-art technologies.

The Convolutional Neural Network (CNN) [3, 5], a very sucessful Deep Learning (DL) technique, is inspired by the human visual processing system. In the ImageNet Challenge, the winner, Alex Krizhevsky, introduced a CNN [4] that showed its massive power as a feature learning and classification architecture. Although AlexNet is very similar to LeNet [5] (published in 1998), the by scaling up of both the data and the computational power brought large performance improvements. Nevertheless, it remains challenging to replicate and model the human visual system. Recent advances combine DL with image foveation and saliency detection models. In [6], a foveated object detector has performance similar to homogeneous spatial resolution, while reducing computational costs.

In the context of foveated image search, our work aims to utilize goal-guided scanpath data for object detection in foveated images. Our object search approach receives as input an image and an object category, then indicates the presence or absence of instances of that category in the scene while adjusting the acuity resolution to mimic the human visual system.

Our contributions include: (i) Benchmark of recent approaches based on DL, which are able to predict fixations, on a recent large-scale dataset; (ii) a ground-truth label function for fixation sequences that is smoother, considering in a better way the spatial structure of the problem; (iii) evaluation of two alternative visual representations (conventional high-level features from VGG and a more elaborate multi-class presence description); and (iv) the introduction of a novel dual task approach that simultaneously performs fixation and target detection.

2 Related Work

Human attention is driven by two major factors, bottom-up and top-down factors [7]. While bottom-up is driven by low-level features in the field of vision, which means saliency detection is executed during the pre-attentive stage, top-down factors are influenced by higher level features, such as prior knowledge, expectations and goals [8]. Depending on the goal/task description, the distribution of the points of fixation on an object varies correspondingly [9].

In Computer Vision, Gaze Prediction models aim to estimate fixation patterns made by people in image viewing. These models can have a spatial representation, in fixation density maps, and an added temporal representation when predicting scanpaths. In this area of study, most work focuses on free-viewing, which, as mentioned, is led by bottom-up attention.

In [10], CNNs are used for feature extraction and feature maps compilation, which are then used in a Long Short Term Network (LSTM) responsible for modeling gaze sequences during free-viewing. LSTM [11] was proposed as a solution to the vanishing gradient problem of RNNs. LSTM networks have a more complex structure that tweak the hidden states with an additive interaction, instead

of a linear transformation, which allows the gradient to fully backpropagate all the way to the first iteration.

However, human scanpaths during search tasks vary depending on the target items they are trying to gain information from, therefore guided search cannot be predicted based on free-viewing knowledge. Goal-directed attention is additionally relevant due to the human search efficiency in complex scenes that accounts for scene context and target spatial relations [12].

In [13], a guided search approach inspired in the architecture of [10] shows promising results. Their approach relies on a Convolutional Long Short Term Memory (ConvLSTM) architecture, and introduced a foveated context to the input images on top of an additional input encoding the search task, which found human fixation sequences to be a good foundation for object localization. The ConvLSTM had been previously introduced in [14] as a variant of LSTMs better suited for 3-dimensional inputs, such as images. This adaptation still contains the same two states: a hidden state, h, and a hidden cell state, c; and the same four intermediate gates: the input gate i, forget gate f, output gate o and candidate input $\tilde{c}$; as the LSTM architecture. However, a convolution is performed during the computation of the gates instead of the previous product operations, as seen in the following equations:

$$
\begin{aligned}
i_t &= \sigma(W_i * x_t + U_i * h_{t-1}) & f_t &= \sigma(W_f * x_t + U_f * h_{t-1}) \\
o_t &= \sigma(W_o * x_t + U_o * h_{t-1}) & \tilde{c}_t &= \tanh\left(W_c * x_t + U_c * h_{t-1}\right) \\
c_t &= f_t \odot c_{t-1} + i_t \odot \tilde{c}_t & h_t &= o_t \odot tanh(c_t)
\end{aligned}
$$

where $\odot$ denotes an element wise product, $*$ denotes a convolution, and W and U are the weight matrices of each gate that operate over the hidden states.

The limited amount of available data containing human scanpaths in visual search was, however, identified as a significant obstacle in [13]. Since then a new large-scale dataset has been introduced in [15], which has shown promising results in [16], where an inverse reinforcement learning algorithm was able to detect target objects by predicting both the action (fixation point selection) and state representations at each time step, therefore replicating the human attention transition state during scanpaths. This approach additionally utilized features extracted from a Panoptic Feature Pyramid Network (Panoptic FPN) model [18], that performs panoptic segmentation which is the unification of "the typically distinct tasks of semantic segmentation (assign a class label to each pixel) and instance segmentation (detect and segment each object instance)" [17].

3 System Overview

In this section, we present the architecture of the two strategies used in this study: a two-stage pipeline system consisting of a gaze fixations predictor and an image classifier, and a dual-task model that conducts scanpath prediction and target detection simultaneously.

3.1 Fixation Prediction Module

We consider the same network architecture for the two types of features of the fixation model: (i) High-level feature maps and (ii) panoptic image features. At each time-step $T = t$, the Input Transformation Section aggregates the features of the foveated pictures at each fixation location from the beginning of the gaze sequence, $T \in 0, ..., t$, as well as the task encoding of the target object. This combined input is then sent to the Recurrent Section, which uses ConvLSTM layers to emulate human-attention through its hidden states. The Recurrent Section then outputs its final hidden state h_{t+1} to the model's Output Section, which predicts the next scanpath fixation as a discrete location in an image grid with dimensions $H \times W$. We now present the architecture of each model in detail.

Fixation Prediction from High-Level Features In this model, we utilized the high-level features retrieved from the ImageNet-trained VGG16 model [19, 20] with dimensions $H \times W \times Ch$, and it is composed of the following sections:

- **Input Transformation**: To condition the image feature maps on the task, we perform an element-wise multiplication of these inputs. In addition, depending on its format, the task encoding may be transmitted through a Fully Connected (FC) Layer with Ch units and a tanh activation, followed by a Dropout Layer with a rate of $r_{Dropout}$ in order to prevent overfitting.

- **Recurrent Section**: This portion mainly consists of a ConvLSTM layer with F filters (dimensionality of its output), a kernel size of K x K, a stride of S, and a left and right padding of P. The ConvLSTM has a tanh activation, and the recurrent step utilizes a hard sigmoid activation [1]. Subsequently, to prevent overfitting we perform batch normalization, where the features are normalized with the batch mean and variance. During inference, the features are normalized with a moving mean and variance.

- **Output Section**: We perform a flattening operation to each temporal slice of the input with the help of a Time Distributed wrapper. The flattened array is then fed to a FC layer and has $H \times W$ units and a softmax activation function.

Fixation Prediction from Panoptic Features To compute these new features we resorted to the Panoptic FPN model in [18]. The belief map computes a combination of high and low resolution belief maps:

$$B(t) = M_t \odot H + (1 - M_t) \odot L, \tag{1}$$

where H and L are the belief maps for the high and low resolution images, respectively, and M_t is the binary fixation mask at time step t, of size $H \times W$, where every element is set to 0 except the grid cells at an euclidean distance shorter than r from the current fixation point.

To duplicate the belief maps in [16], the task encoding is a one-hot encoding with dimensions $H \times W \times Cl$, where each row of the axis Cl corresponds to an

[1] a piece-wise linear approximation of the sigmoid function, for faster computation.

object class and the two-dimensional map $H \times W$ is all set to one for the target class and zero for the others. This input is subsequently transmitted to the Input Transformation stage, where it is concatenated with the image feature maps.

The recurrent section of the model is composed of d ConvLSTM layers, each followed by a Batch Normalization layer. Every ConvLSTM is constructed with the same hyper-parameters: each one has F_{LSTM} filters with a kernel size of $K_{LSTM} \times K_{LSTM}$, a stride of S_{LSTM}, a padding of P_{LSTM}, a tanh activation function and a hard sigmoid activation during the recurrent step.

Finally, in the output section, we conducted experiments over two different setups. The first one is composed of a 3d-Convolutional layer with a sigmoid activation followed by a time distributed flattening operation. The second setup is comprised of the same 3d-Convolutional layer, but with a ReLu activation, and a flattening layer followed by a FC layer with softmax activation.

3.2 Target Detection Module

In the last stage of the pipeline, the model evaluates at each time-step, if the fixation point coincides with the location of the target object. To detect the target, we rely on VGG16 architecture, and develop 18 binary classifiers, one for each task. In addition, as a baseline, we utilize a complete VGG16 trained on the ImageNet dataset to perform classification on our data.

To fine-tune the already pre-trained VGG16 model, we substituted its classification layers, with three fully connected layers, FC_i with $i \in \{1, 2, 3\}$, each with U_i units, where FC_1 and FC_2 were followed by a ReLu activation function and FC_3 was followed by a sigmoid action function. During training, only the parameters of these last FC layers were updated.

3.3 Dual Task Model

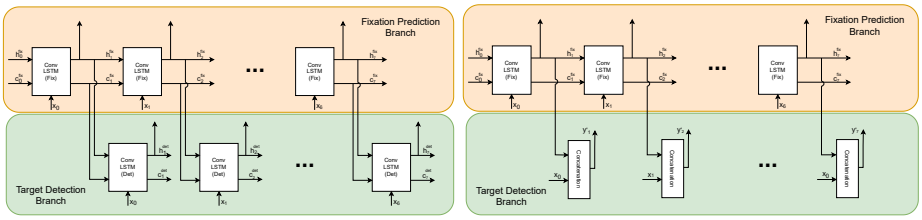
We aim to both predict the fixation point and localize the target object, by sharing the internal states on two LSTMs branches. We consider three different architectures: A, B and C. All models receive as input the high-level feature maps, with dimensions $H \times W \times Ch$ and a one-hot task encoding array, of size Cl, which are then aggregated. Similar to Section 3.1, the grouping of both of these inputs is accomplished by passing the task encoding through a FC layer with Ch units and tanh activation, and conducting an element-wise multiplication with the foveated image's feature maps. After this shared module, the models branch off to complete each specific task using the following architectures:

- **Architecture A (fixation-first)** After performing the input transformation, where we aggregate the feature maps and task encoding, the array x_t is fed to two ConvLSTM layers. Then, following each iteration of the fixation prediction recurrent module, its internal states h_t^{fix} and c_t^{fix} are passed to the detection branch as the internal states, h_{t-1}^{det} and c_{t-1}^{det}, of the preceding time step, as illustrated in figure 1a. In the first branch, a temporal flattening operation is performed to h^{fix}, followed by an output layer consisting of a FC layer with softmax activation. In the second we classify each temporal slice of h^{det} by

employing the same structure of three FC layers $FC_i \in \{1, 2, 3\}$, with U_i units, respectively, where the first two layers have a ReLu activation while the output layer has a sigmoid activation.

- **Architecture B (detection-first)** Similar to the previous architecture, each task branch employs ConvLSTM layer. The sole difference is that we now conduct the iterations of the detection module first, and send the internal states h_t^{det} and c_t^{det} to the fixation prediction module for the preceding time step $t-1$.

- **Architecture C** The fixation prediction is a copy from architecture A. In difference, the target detection branch no longer has a ConvLSTM layer. Instead, at each time step t, the combined input x_t computed by the shared module is concatenated with the output of the ConvLSTM layer, h_{t+1}^{fix} of the fixation prediction task, as illustrated in figure 1b. Finally, this concatenation is followed by the same three FC layers utilized by the previous architectures.



(a) Architecture A. **(b)** Architecture C.

Fig. 1: Information flow across the fixation prediction and target detection branch in architectures A, on the left, and C, on the right.

4 Implementation

4.1 Dataset

We use the COCO-Search18 dataset [15]. This dataset consists of 6,202 images from the Microsoft COCO dataset [22], evenly split between target-present and target-absent images, of 18 target categories [2], with eye movement recordings from 10 individuals. As humans were able to fixate the target object within their first six saccades 99% of the time, fixation sequences with length greater than that were discarded. Additionally, the sequences were padded with a repeated value of the last fixation point to achieve a fixed length of 7, including the initial center fixation. This was done to replicate the procedure of a similar work [21], where participants were instructed to fixate their gaze on the target object, once they found them, during search tasks. To train the fixation prediction module and the dual task model, we used a random dataset split of 70% train, 10%

[2] bottle, bowl, car, chair, analogue clock, cup, fork, keyboard, knife, laptop, microwave, mouse, oven, potted plant, sink, stop sign, toilet and tv.

validate and 20% test over each class category and all images were resized to 320 x 512 which resulted in feature maps with 10×16 spacial dimensions.

4.2 Training

During the training phase, all our models were optimized with the Adam algorithm [23] and a learning rate of $lr = 0.001$, for a maximum of 100 epochs with an early stopping mechanism activated when the validation loss stops improving after a duration of 5 epochs. Additionally, every dropout is performed with $r_{Dropout} = 0.5$ and we use a batch size of 256 in every module apart from the fixation prediction performed with high-level features.

Fixation Prediction from High-Level Features We estimate the weights and bias parameters that minimize the loss between the predicted output $\hat{y}$ and the ground truth label y, with the cross entropy function computed for every fixation time step t for every sequence s of each mini-batch:

$$L_{CE} = -\sum_{s=1}^{S}\sum_{t=0}^{T}\sum_{i=1}^{H\times W} y_i * log(\hat{y}_i), \tag{2}$$

where S corresponds to the batch size, T to the sequence length which is set to 6 (in addition to the initial fixation point at $t = 0$) and $H \times W$ to the output size which is set to 160. We set with $F = 5$ filters, a kernel size of $K = 4$ and a stride of $S = 2$, and varied the batch size between 32, 64, 128 and 256. We conduct an ablation study over these additional hyper-parameters and settings: (i) Fovea size: We use the same real-time foveation system as in [13], considering three fovea sizes: 50, 75 and 100 pixels. (ii) Task encoding: We consider two representations. The first is a one-hot encoding array of size 18. The second is a normalized heat map of fixations made during the observations of that same task, compiled exclusively with training data. (iii) Ground truth function: We consider both a one-hot encoding representation of the ground-truth label and a two dimensional Gaussian function with the mean set to the cell coordinates of the actual fixation location and the variance set to 1.

Fixation Prediction from Panoptic Features We want to minimize the loss function in equation 2. Additionally, the feature maps used have dimension $10 \times 16 \times 134$ in order to replicate the scale of our grid shaped output. The configuration of the ConvLSTM: $F_{LSTM} = 10$ filters with square kernels of size $K_{LSTM} = 3$, a stride of $S_{LSTM} = 1$ and a padding of $P_{LSTM} = 1$ to maintain the features spatial resolution. In the output section, $F_{Conv} = 1$ for the 3d-Convolutional layer to have a kernel size of $K_{Conv} = 2$, a stride of $S_{Conv} = 1$ and padding of $P_{Conv} = 1$. The second setup of this section is configured to have a Fully Connected layer with 160 units.

For this model, we additionally varied the depth of the recurrent section with $d \in \{1, 3, 5\}$, and altered the structure of the output section to utilize both

a sigmoid and softmax as the final activation function. Concerning the data representation, we once again evaluated the impact of having a one-hot or a Gaussian ground truth encoding, and explored several belief maps settings: we varied the radius r of the mask, M_t, with values $r \in \{1, 2, 3\}$ (each to emulate a corresponding fovea size of 50, 75 and 100); and experimented with a cumulative mask configuration, M'_t, where the binary mask utilized in equation 1, in addition to the information of the current time step, accumulates the high acuity knowledge of all previous time steps. All panoptic feature maps were computed with a low resolution map L extracted from a blurred input image with a Gaussian filter with radius $\sigma = 2$.

Target Detection The binary classifiers were implemented with each fully connected layer having $U_1 = 512$, $U_2 = 256$ and $U_3 = 1$ units, a dropout rate of $r_{Dropout} = 0.5$, and we varied the fovea size between 50, 75 and 100 pixels. They were trained with a loss function defined as:

$$L_{BCE} = -\frac{1}{N}\sum_{i=1}^{N} y_i \cdot log(\hat{y}_i) + (1 - y_i) \cdot log(1 - \hat{y}_i) \tag{3}$$

Dual Task In this case, the ConvLSTM layers are configured with $F = 5$ filters of size $K = 4$ to execute the convolutional operations with stride $S = 2$, while the fully connected layers of the detection branch are configured with $U_1 = 64$, $U_2 = 32$ and $U_3 = 1$ units. The loss function of the dual task L_{Dual} is:

$$L_{Dual} = w_{fix} \cdot L_{fix} + (1 - w_{fix}) \cdot L_{det}, \tag{4}$$

where L_{fix} and L_{det} correspond to the loss of the fixation and detection prediction, respectively. L_{fix} corresponds to the categorical cross entropy, like in (2), while L_{det} is a weighted binary cross entropy as follows in (5):

$$w = y \cdot w_1 + (1 - y) \cdot w_0; \quad L_{det} = w \cdot [y * log(\hat{y}) + (1 - y) \cdot log(1 - \hat{y})]. \tag{5}$$

Since the target is absent in half of the images and appears in a limited section of the scanpath sequence, (5) includes sample-based weights. Due to the high imbalance of the detection data we add the weights $w_1 = 1.6$ and $w_0 = 0.7$. In the case of w_1 we compute the multiplicative inverse of the ratio of positive detections on the total number of detections, and dividing it by 2. In the case of w_0, the inverse ratio of negative detections divided by 2.

To determine the optimal configuration for each model's architecture, an ablation study was done over the fovea size (50, 75 and 100 pixels) and the degree of importance w_{fix} (0.10, 0.25, 0.50, 0.75, 0.90) in Eq. (4).

4.3 Prediction

During the testing phase, the single and dual task models aim to predict a scanpath sequence of fixed length $l = 7$, based on the training data, and the

fixation point at $t = 0$ as the center cell of the discretized grid. We apply the beam search algorithm, which selects the best m fixation points at each time step. Then, the selected points are appended to the sequences they were generated from, while saving the target detection prediction in the case of the dual task model. In the next time step, the model runs for each of these m predicted sequences, to select the next best m predictions. In our experiments, $m = 20$.

In regards to the target presence detector, all models were deployed on the scanpaths produced by the highest performing scanpath predictor. The baseline classifier was tested similarly to the binary models, but cropping the images to 224×224. Due to the mismtaching classes between the datasets, we adapt the ImageNet classes to our targets by grouping some sibling sub-classes [3], and remove the non-exisiting classes in ImageNet. The target is present when the ground-truth class has the highest classification score and as absent otherwise.

5 Results

5.1 Two Stage-Pipeline

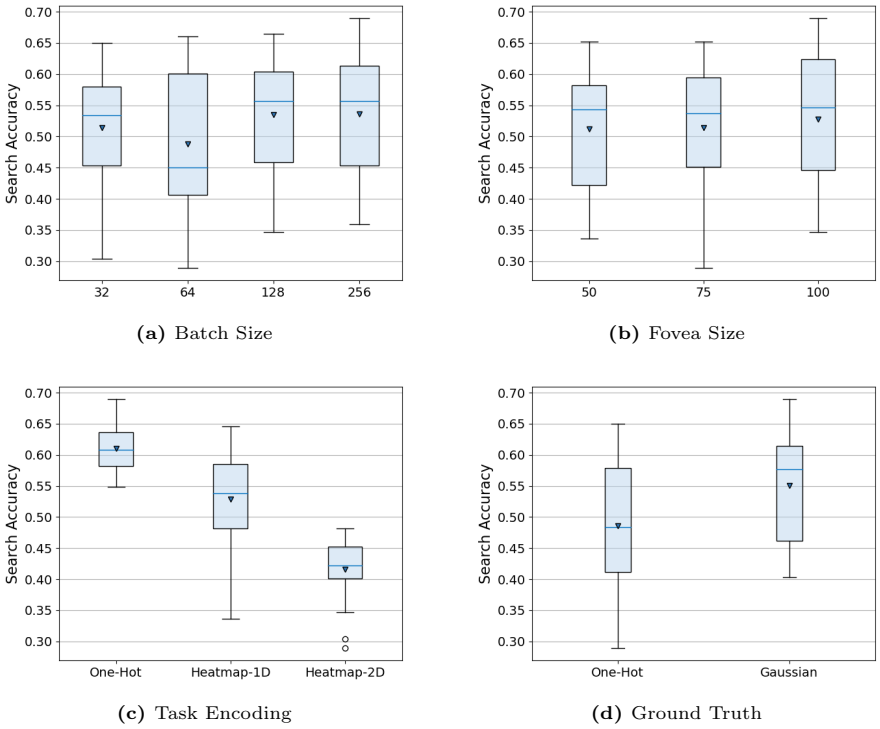
Fixation Prediction The evaluation metrics include: (i) **Search Accuracy** which is computed as the ratio of sequences in which a fixation point selects a grid cell that intersects the target's bounding box, (ii) **Target Fixation Cumulative Probability (TFP)**, which is plotted in figure 3, and presents the search accuracy attained by each time step. On the TFP, we compute the **TFP - Area Under Curve (TFP-AUC)** and the **Probability Mismatch**, which is the sum of absolute differences between the model's TFP and the human's observable data. Finally, (iii) the **Scanpath Ratio** as the ratio between the sum of euclidean distances between each fixation point and the distance from the initial fixation to the center of the target's bounding box.

Through the ablation study we conducted for the two stage pipeline, we found that the high-level features scanpath predictor achieves top search accuracy scores (0.69) when using a one-hot task encoding and a Gaussian ground-truth, as seen in figure 2, where the search scores are depicted in box-plots grouped by each training setting.

Figure 3 shows the TFP curve of several highest model configurations, as well as human search behavior, and a random scanpath baseline model[4]. We noticed: (i) A decrease in performance across time for all models through the slopes of each function, (ii) all the models but the random one were able to detect the most of the targets by the second fixation step and (iii) our models barely detected any new targets on the last four fixation points. The right side of Figure 3 shows that fixations converge across time steps, but seems to be a behavior copycat from the training set.

[3] e.g. the task bowl corresponds to the joint sub-classes mixing bowl and soup bowl.

[4] We select a human sequence randomly from the train split for the same search target class on the testing set.



(a) Batch Size **(b)** Fovea Size

(c) Task Encoding **(d)** Ground Truth

Fig. 2: Search accuracy box-plots of the single task High-level features' models, grouped by training settings, with the mean values (triangles) and outliers (circles).

Regarding the ablation study of fixation prediction with the panoptic features shown in Figure 4, the highest search accuracy score of 0.686 was attained with a cumulative mask of radius $r = 1$. We note that using a sigmoid activation improves the model's results because the ground truth is Gaussian. In contrast, when a final softmax activation is utilized, the model turns the scores of the last hidden layers into class probabilities. Due to the ground truth encoding, there is a saturation of the loss when every grid cell is considered, as opposed to only examining the probability of the true class in a one-hot encoding configuration, resulting in the model's poor performance.

By comparing the high-level vs. panoptic features on figure 3, we see that the models with high-level feature maps fixate targets much sooner. However, the panoptic features lead to a similar search accuracy. The panoptic-based model is much less efficient as the scanpaths travel a much greater distance, as seen in figure 3, leading to a a scanpath ratio score of only 0.463.

Target Detection For the target detection task we used **accuracy**, **precision** and **recall** as metrics. The fine-tuned classifiers with foveation radius of 50 pixels had the maximum performance for all measures, with a mean accuracy, precision

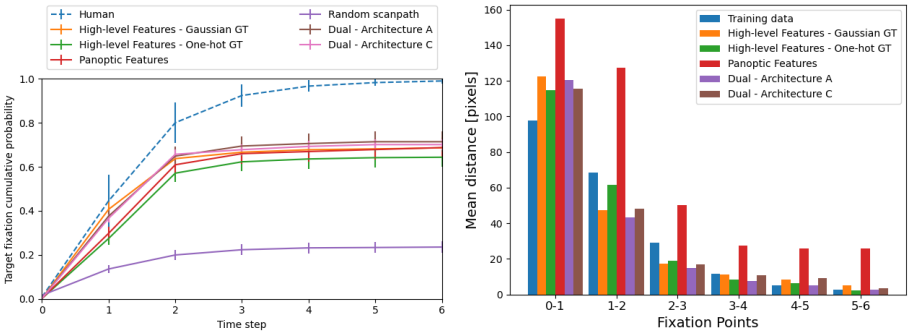


Fig. 3: Left side: Search accuracy per model along scanpaths, with means and standard errors computed over target classes. The Human TFP refers to the human behavior observed in the entire COCO-Search18 dataset. The remaining TFP curves were computed for the test data split. Right side: Euclidean distances between fixation points.

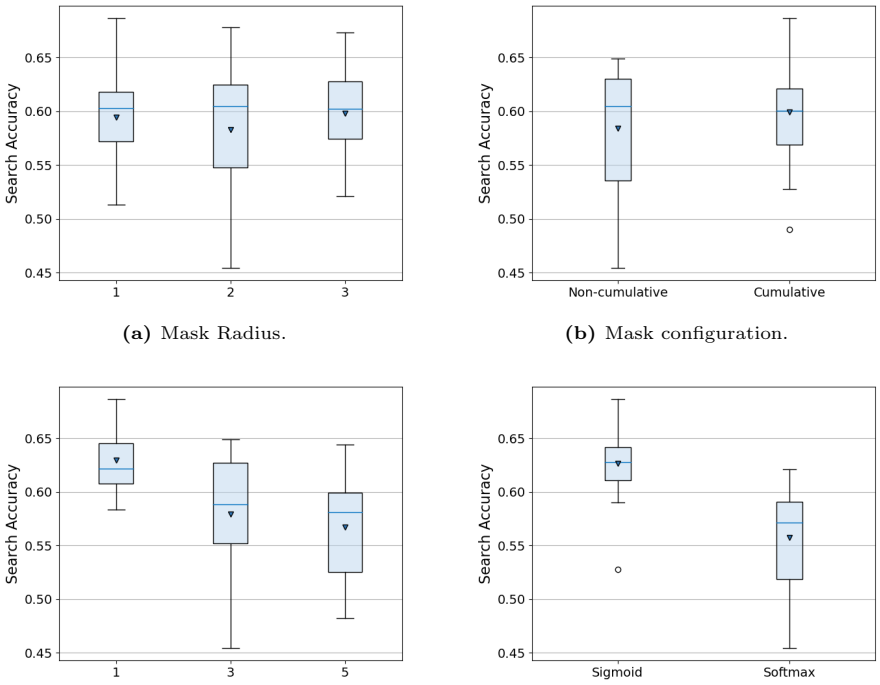
Table 1: Performance evaluation of best performing models (rows) based on Fixation Prediction metrics (columns). The ↑ indicates higher is better, and ↓ lower is better.

	Search Accuracy ↑	TFP-AUC ↑	Probability Mismatch ↓	Scanpath Ratio ↑
Human	0.990	5.200	-	0.862
High-Level Features - One-hot GT	0.650	3.068	1.727	0.753
High-Level Features - Gaussian GT	0.690	3.413	1.360	0.727
Panoptic Features	0.686	3.259	1.514	0.463
Dual - Architecture A	0.719	3.496	1.263	0.808
Dual - Architecture C	0.701	3.446	1.320	0.791
IRL	N/A	4.509	0.987	0.826
BC-LSTM	N/A	1.702	3.497	0.406
Random Scanpath	0.235	1.150	3.858	-

and recall of 82.1%, 86.9% and 75.4%, respectively, whereas the configuration of 75 pixels achieved the lowest accuracy and recall. In turn, the baseline pre-trained model has a very large variance across metrics for the vast majority of the classes, as seen in figure 5b. Performance increases slightly with larger foveation radius, reaching the highest accuracy, precision and recall scores (64.0%, 85.7% and 34.3%) for the 100 pixels fovea size.

5.2 Dual Task

Figure 6a shows that architecture A outperforms more than half of the models of architecture C on search accuracy (top value of 73.4% with a fovea size of 100 pixels and $w_{fix} = 0.75$). Regarding detection performance, the dual task model with architecture C, a fovea size of 50, and $w_{fix} = 0.9$ achieved the highest



(a) Mask Radius.

(b) Mask configuration.

(c) Depth of recurrent section.

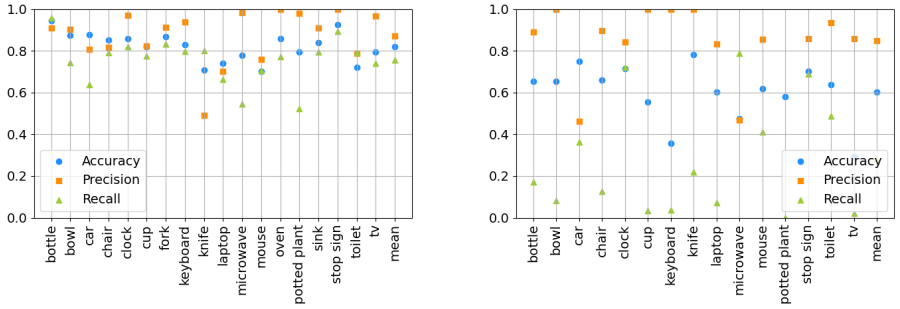
(d) Activation Function.

Fig. 4: Search accuracy box-plots of the single task Panoptic features' models grouped by their training settings, with the mean values (trinagles) and outliers (circles).

target presence detection rate of 68.7%. In Figure 6c, considering the quartiles and upper limit of its performance, the architecture of design C is deemed to be the most effective. Regarding the weight of the fixation loss, we can also see that models trained with bigger values obtained a larger interquartile range than models trained with smaller values.

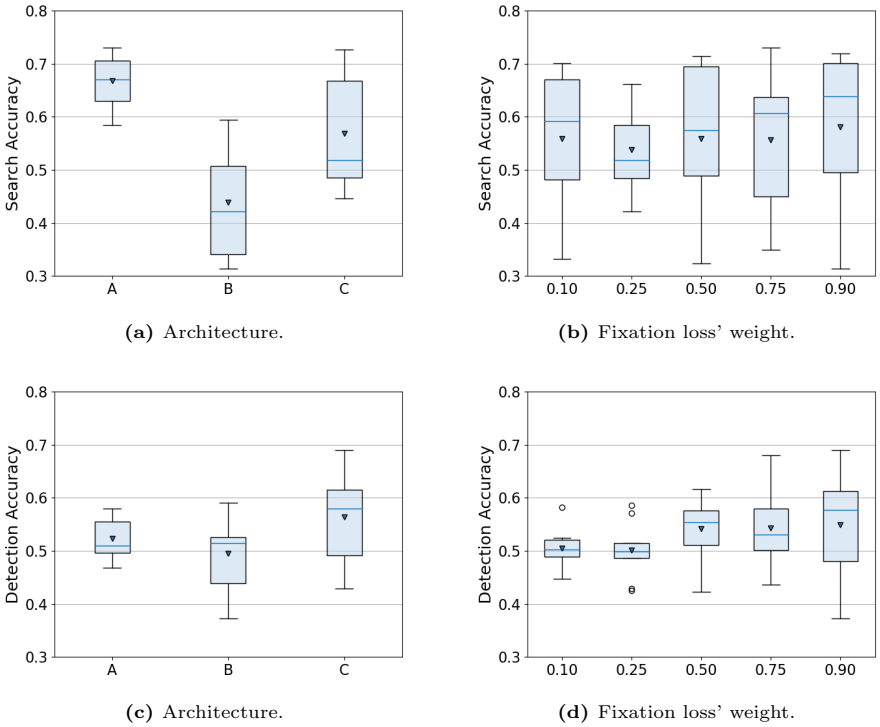
In addition, note that the best scanpath prediction model achieved a detection accuracy of 49.7% while the best target presence predictor achieved a search accuracy of 63.9%. To have a single value for evaluation, we also considered the average of both metrics. The majority of the time, design A earned a higher score than design C, while design B ranked the lowest. Regarding the remaining parameters, we observe that a bigger fovea radius led to higher average scores, and a higher fixation loss' weight resulted in a better top score, with the exception of setting $w_{fix} = 0.25$. The model configured with architecture C, a fovea size of 75 pixels, and $w_{fix} = 0.9$ achieved the top score of 67.7% with search and detection accuracies of 70.1% and 65.3%, respectively.

Finally, the dual approach led to higher search accuracy, as seen in Table 1, resulting in a higher TFP-AUC score of 3.496 and 3.446 and a lower probability mismatch of 1.263 and 1.320 for the overall best models with architectures A



(a) Fine-Tuned Classifier **(b)** Baseline Classifier

Fig. 5: Performances of the fine-tuned and baseline target detectors in terms of accuracy, precision and recall, for the fovea size setting of 50 pixels on the left side and 100 pixels on the right side.



(a) Architecture. **(b)** Fixation loss' weight.

(c) Architecture. **(d)** Fixation loss' weight.

Fig. 6: Box-plots of the Search accuracy, on the top, and Detection accuracy, on the bottom, grouped by configuration, with mean values (triangles) and outliers (circles).

and C, respectively. In addition, the two models exhibit a higher search efficiency with scanpath ratios of 0.808 and 0.791, respectively.

6 Conclusions and Future Work

We present two methods for predicting the presence or absence of a target in an image with foveated context: (i) a two-stage pipeline and (ii) a dual task model. In the first one, the fixation prediction module produced the best results with high-level feature maps, both in terms of search accuracy and search efficiency, when compared to panoptic features. In addition, we found that a Gaussian ground truth label encoding, enhanced search accuracy. This novel representation captures the spatial structure of the problem, encouraging both the exact discretized human fixation positions as well as attempts to cells near these locations. Two classifiers performed the target presence of the two-stage pipeline model, where the fine-tuned classifiers for multiple binary tasks performed better than a pre-trained VGG-16.

The final contribution of this work is a dual-task model that executes both tasks concurrently while enabling information sharing between them by executing a common input transformation and establishing linking channels throughout each task branch. This multi-task approach improved search precision when the task prediction branch initiated the predictions, i.e. in designs A and C. However, the former suffered a reduction in detection accuracy, whilst the latter achieved the maximum score when compared to our baseline method. Finally, we found that the use of a recurrent layer biases the model towards temporal patterns of target detection of the dual-task. An alternative solution would be for the task branch to generate the input image simultaneously, simulating an encoder-decoder, so as to require the model to maintain its knowledge of high-level features in its hidden states.

Future research should also investigate a visual transformer-based design, as it has shown promising results in similar image classification and goal-directed search tasks.

Acknowledgements. Work partially supported by the LARSyS - FCT Project [UIDB/50009/2020], the H2020 FET-Open project Reconstructing the Past: Artificial Intelligence and Robotics Meet Cultural Heritage (RePAIR) under EU grant agreement 964854, the Lisbon Ellis Unit (LUMLIS)

References

1. Borji, A. and Itti, L., 2012. State-of-the-art in visual attention modeling. IEEE transactions on pattern analysis and machine intelligence, 35(1), pp.185-207.
2. Bandera, C. and Scott, P.D., 1989, November. Foveal machine vision systems. In Conference Proceedings., IEEE International Conference on Systems, Man and Cybernetics (pp. 596-599). IEEE.
3. LeCun, Y., Bottou, L., Bengio, Y. and Haffner, P., 1998. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), pp.2278-2324.
4. Krizhevsky, A., Sutskever, I. and Hinton, G.E., 2017. Imagenet classification with deep convolutional neural networks. Communications of the ACM, 60(6), pp.84-90.
5. Fukushima, K., 1988. Neocognitron: A hierarchical neural network capable of visual pattern recognition. Neural networks, 1(2), pp.119-130.

6. Akbas, E. and Eckstein, M.P., 2017. Object detection through search with a foveated visual system. PLoS computational biology, 13(10), p.e1005743.
7. James, W., 1890. The principles of psychology, Vol. 1. Henry Holt and Co.
8. Corbetta, M. and Shulman, G.L., 2002. Control of goal-directed and stimulus-driven attention in the brain. Nature reviews neuroscience, 3(3), pp.201-215.
9. Yarbus, A.L., 2013. Eye movements and vision. Springer.
10. Ngo, T. and Manjunath, B.S., 2017, September. Saccade gaze prediction using a recurrent neural network. In 2017 IEEE International Conference on Image Processing (ICIP) (pp. 3435-3439). IEEE.
11. Graves, A., 2012. Long short-term memory. Supervised sequence labelling with recurrent neural networks, pp.37-45.
12. Kreiman, G. and Zhang, M., 2018. Finding any Waldo: zero-shot invariant and efficient visual search.
13. Nunes, A., Figueiredo, R. and Moreno, P., 2020, June. Learning to search for objects in images from human gaze sequences. In International Conference on Image Analysis and Recognition (pp. 280-292). Springer, Cham.
14. Shi, X., Chen, Z., Wang, H., Yeung, D.Y., Wong, W.K. and Woo, W.C., 2015. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. Advances in neural information processing systems, 28.
15. Chen, Y., Yang, Z., Ahn, S., Samaras, D., Hoai, M., and Zelinsky, G. (2021). COCO-Search18 Fixation Dataset for Predicting Goal-directed Attention Control. Scientific Reports, 11 (1), 1-11, 2021.
16. Yang, Z., Huang, L., Chen, Y., Wei, Z., Ahn, S., Zelinsky, G., Samaras, D., and Hoai, M. (2020). Predicting Goal-directed Human Attention Using Inverse Reinforcement Learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 193-202).
17. Kirillov, A., He, K., Girshick, R., Rother, C. and Dollár, P., 2019. Panoptic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 9404-9413).
18. Kirillov, A., Girshick, R., He, K. and Dollár, P., 2019. Panoptic feature pyramid networks. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 6399-6408).
19. Simonyan, K. and Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.
20. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K. and Fei-Fei, L., 2009, June. Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition (pp. 248-255). Ieee.
21. B. Cabarrão, Learning to search for objects in foveal images using deep learning, Master's thesis, Universidade de Lisboa - Instituto Superior Técnico, 2022
22. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P. and Zitnick, C.L., 2014, September. Microsoft coco: Common objects in context. In European conference on computer vision (pp. 740-755). Springer, Cham.
23. Kingma, D.P. and Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.