Learning where to look for a target before seeing it

\begin{abstract}
A realistic system for object categorization should be able to find the target in possibly large images, independently of its position in visual space. Current solutions leverage this solution by processing the different hypothesis (classes) at all possible spatial configuration. However, this can be costly in terms of computing time especially without dedicated parallel hardware. %
We explore here a solution inspired by the anatomy of the human visual system, that is, the combination of a foveated sensor with the capacity of rapidly moving the center of fixation using saccades. Indeed, the position and category of objects in images are a priori independent and we hypothesize that the retinotopic map overlays a central area dedicated to object categorization and a peripheral area dedicated to
Using this hypothesis, we formalize this problem in a probabilistic setting which allows us to build two parallel but interactively connected system: a classical image classification algorithm assuming that gaze is centered on the object on one side and a system learning to infer the position of the target on the other. Until the classification is confident, the system performs a saccade to the most likely position in the image. Overall, the computational cost of this strategy is less than that in holistic methods. %
We tested this framework on a simple task of finding digits in a large, cluttered image. Results demonstrate that it is possible to correctly learn the position of a target from a given class, and this before actually seeing a foveated image of the target. We compare the results of this model with classical psychophysical results in visual search. This provides evidence of the importance of such strategies in computer vision and we highlight some predictions of our model.
\end{abstract}

Notes:

- differential processing
  - periphery : low resolution in space
    high in time
  - fovea : high spatial resolution
    low temporal

- saccadic suppression
  - Ziad Hafed : elevation of perceptual thresholds at the time of saccade
    → frequency tuning in SC

Notes existantes

2017-09-28

- manu → papier frontiers
- Coarse periphery
- Hermann grid?
- invariances

2018-03-15 sujet pierre

- def sujet / protocole
- def methode avec carte accuracy
- oral

2018-12-21

- outline paper
- Figures
- yaka

1 Introduction

1.1 Issue

1.2 State of the art

1.3 Outline

→ in the methods

1.3.1 Notations

- $\boldsymbol{x}$: visual field (image)
- $\boldsymbol{y}$: target category (categorical)
- $\boldsymbol{u}$: target position (real coordinates or categorical, retinocentric referential)

Generative model :

$$\boldsymbol{x} \sim P(X|\boldsymbol{y}, \boldsymbol{u})$$

Full inference (posterior):

$$P(Y, U|\boldsymbol{x}) \propto P(\boldsymbol{x}|Y, U)$$

Independence assumptions :

$$P(Y, U) = P(Y)P(U) \text{ \textit{(toujours vrai)}} \tag{1}$$

$$P(Y, U|X) = P(Y|X)P(U|X) \text{ \textit{(faux s'il y a plusieurs cibles)}} \tag{2}$$

Partial inference on object category:

$$P(Y|\boldsymbol{x}, \boldsymbol{u}) \propto P(\boldsymbol{x}|Y, \boldsymbol{u})$$

Partial inference on object position:

$$P(U|\boldsymbol{x}, \boldsymbol{y}) \propto P(\boldsymbol{x}|U, \boldsymbol{y})$$

Marginals:

- $P(Y|\boldsymbol{x}) = \int P(Y|\boldsymbol{x}, \boldsymbol{u})d\boldsymbol{u}$
- $P(U|\boldsymbol{x}) = \int P(U|\boldsymbol{x}, \boldsymbol{y})d\boldsymbol{y}$

1.3.2 What we did so far...

Consider a view $\boldsymbol{x}$ that contains a single target $\boldsymbol{y}$ at unknown retinocentric position $\boldsymbol{u}$. The brain needs to guess both $\boldsymbol{y}$ and $\boldsymbol{u}$ with limited computational resources.

We assume here that the brain adopts independence assumption (2), making a separation between the "Where" and the "What" pathways, forming separate (and cheaper) inferences :

- $p(Y|\boldsymbol{x})$
- $p(U|\boldsymbol{x})$

Another assumption is that the category $\boldsymbol{y}$ is *translationally invariant*: given a transformation $\mathcal{T}$,

$$\mathcal{T}(\boldsymbol{u}, \boldsymbol{y}) = (\mathcal{T}(\boldsymbol{u}), \boldsymbol{y})$$

Now, given $\boldsymbol{x}$ and the separation assumption, it is sensible to change the viewpoint to better estimate $\boldsymbol{y}$, because $\boldsymbol{y}$ is invariant to the viewpoint transformation.

This is where *active inference* comes into the play:

1 INTRODUCTION

1a Active Inference in Machine Learning

deep learning is passive / feed-forward

saccades → sparsity of the visual world / identity problem of figure 1

Saliency

essential comp.

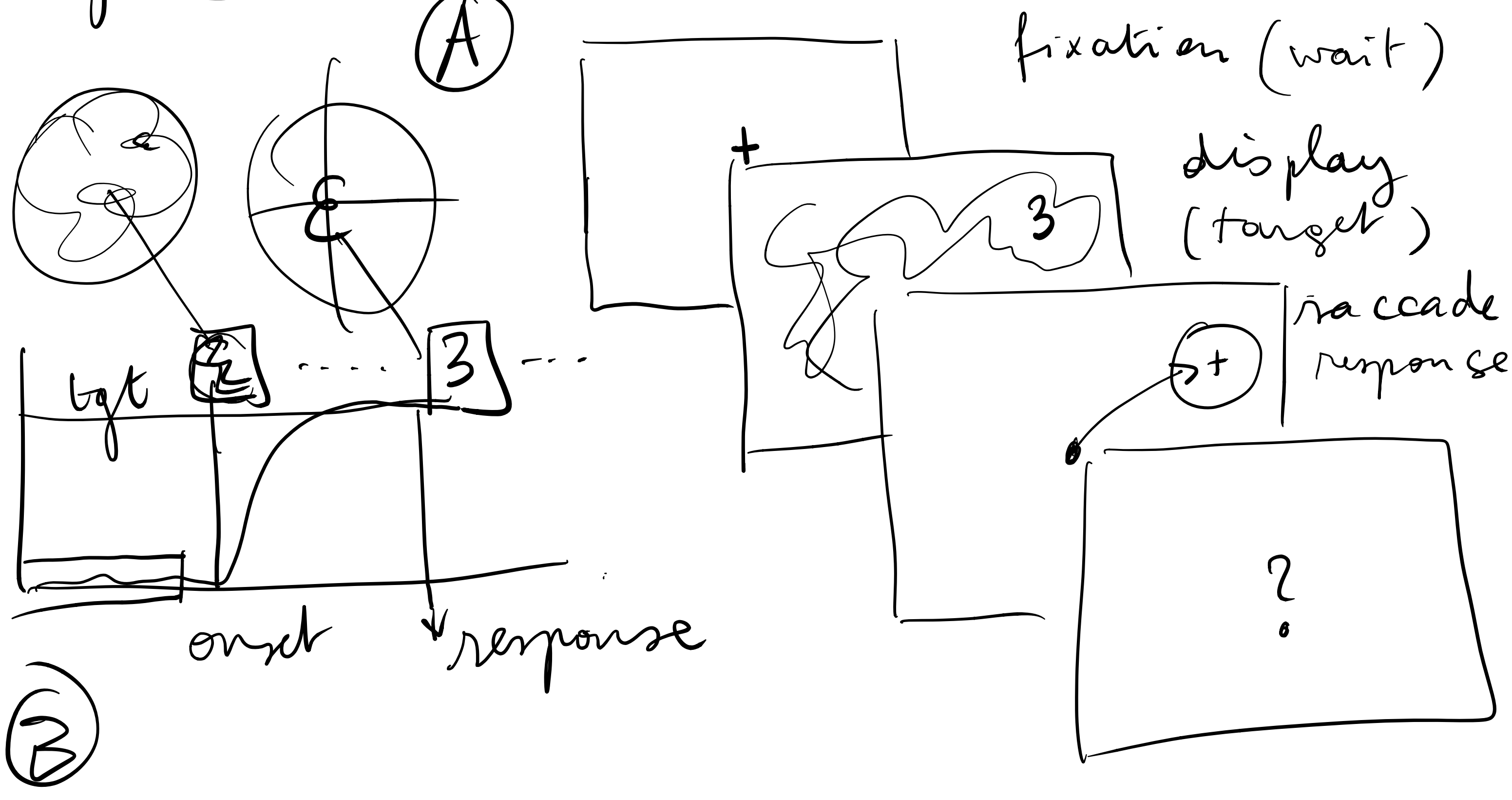
Bethge Deep Gaze → external view.

Robotics

→ identify gap.

→ fragmentation of studies between neuroscience, machine learning and robotics.

Figure 1:



(A) Problem setting: After a fixation period, an observer is presented with a luminous display which shows a target (here a digit) at a random position. The display is presented for a short period but enough to perform a saccade on the potential target. In particular, The configuration of the display is such that by adding clutter and reducing the size of the digit it may become necessary to perform a saccade to be able to see the digit.

Finally, the observer identifies the digit.

Ⓑ We show a prototypical trace of a saccadic eye movement to the target position. In particular, we show the fixation window used to ensure fixation during that window (green shaded area). Overlaid is a simulation of the retinotopic map at the onset of the display and after a (successful) saccade. This demonstrates that the position of the target has to be inferred from a degraded (sampled) image and that a correct identification is mediated by the action to the location of the target * before seeing it *.

1b State-of-the-art

Najemnik

Priebe

Friston . interoceptive representation

1c outline of the paper

- basaic hypothesis = invariance to position

- Consider that the true target is $\hat{\boldsymbol{y}}$
- Consider that the target current retinocentric position is $\boldsymbol{u}$
- Then, for any translation $\delta\boldsymbol{u}$, the future posterior on the true target is estimated by: $\mathbb{E}_{\boldsymbol{x}'\sim p(X|\hat{\boldsymbol{y}},\boldsymbol{u}+\delta\boldsymbol{u})} p(\hat{\boldsymbol{y}}|\boldsymbol{x}')$
- And the optimal translation is: $\underset{\delta\boldsymbol{u}}{\text{argmax}}\ \mathbb{E}_{\boldsymbol{x}'\sim p(X|\hat{\boldsymbol{y}},\boldsymbol{u}+\delta\boldsymbol{u})} p(\hat{\boldsymbol{y}}|\boldsymbol{x}')$

If now $\boldsymbol{u}$ is unknown and needs to be guessed from $\boldsymbol{x}$, the optimal translation is:

$$\underset{\delta\boldsymbol{u}}{\text{argmax}}\ \mathbb{E}_{\boldsymbol{u}\sim p(U|\boldsymbol{x})}\mathbb{E}_{\boldsymbol{x}'\sim p(X|\hat{\boldsymbol{y}},\boldsymbol{u}+\delta\boldsymbol{u})} p(\hat{\boldsymbol{y}}|\boldsymbol{x}')$$

with :

- $p(U|\boldsymbol{x})$ the inferred target position
- and $\mathbb{E}_{\boldsymbol{x}'\sim p(X|\hat{\boldsymbol{y}},\boldsymbol{u}+\delta\boldsymbol{u})} p(\hat{\boldsymbol{y}}|\boldsymbol{x}')$ the expected inference on the actual target.

1.3.3 Accuracy maps

In practice, it is computationally impossible to make exact guesses about the future observation $\boldsymbol{x}'$. Our second assumption is that instead of predicting future inferences on true target, the brain trains a *parametric accuracy map* by experience (trial and error).

In a model-based approach, the *accuracy maps* can be calculated using a parametric classifier :

- Given a training set $\{(x_1, u_1, y_1), ..., (x_n, u_n, y_n)\}$:
  - Train a classifier p_θ that estimates $p(Y|\boldsymbol{x})$.
- Then, for each class $\hat{\boldsymbol{y}}$, taking $\tilde{\boldsymbol{y}} \sim p_\theta(Y|\boldsymbol{x})$, *the classification rate* $r_\theta(\boldsymbol{u})$ *is an estimator of the posterior expectation :*

$$\begin{aligned} r_\theta(\boldsymbol{u}) &= \mathbb{E}_{\boldsymbol{x}\sim p(X|\hat{\boldsymbol{y}},\boldsymbol{u})}\mathbb{E}_{\tilde{\boldsymbol{y}}\sim p_\theta(Y|\boldsymbol{x})}\delta_{\hat{\boldsymbol{y}}=\tilde{\boldsymbol{y}}} \\ &= \mathbb{E}_{\boldsymbol{x}\sim p(X|\hat{\boldsymbol{y}},\boldsymbol{u})} p_\theta(\hat{\boldsymbol{y}}|\boldsymbol{x}) \\ &\simeq \mathbb{E}_{\boldsymbol{x}\sim p(X|\hat{\boldsymbol{y}},\boldsymbol{u})} p(\hat{\boldsymbol{y}}|\boldsymbol{x}) \end{aligned}$$

  that forms an *accuracy map* for each target position $\boldsymbol{u}$.

1.3.4 Parametric transformation (Colliculus?) map

One can now select $\delta\boldsymbol{u}$ with the parametric estimator:

$$\begin{aligned} \widehat{\delta\boldsymbol{u}} &\simeq \underset{\delta\boldsymbol{u}}{\text{argmax}}\ \mathbb{E}_{\boldsymbol{u}\sim p(U|\boldsymbol{x})} r_\theta(\boldsymbol{u}+\delta\boldsymbol{u}) \\ &= \underset{\delta\boldsymbol{u}}{\text{argmax}}\ Q(\delta\boldsymbol{u}|\boldsymbol{x}) \end{aligned}$$

with $Q(\delta\boldsymbol{u}|\boldsymbol{x})$ the *transformation* map, given the view $\boldsymbol{x}$ and the marginal posterior estimate $p(U|\boldsymbol{x})$.

It must be noticed that, given $\hat{\boldsymbol{u}} = \underset{\boldsymbol{u}}{\text{argmax}}\ r_\theta(\boldsymbol{u})$), the transformation map is maximal at $\delta\boldsymbol{u} = \hat{\boldsymbol{u}} - \boldsymbol{u}$. Each initial $\boldsymbol{u}$ provides a different transformation map, that is a shift of the original accuracy map (*Ergodic assumption??*).

We assume in the following that a parametric action value map Q_ψ can be trained on top of the parametric classifier p_θ and its accuracy map r_θ. The training set is $\{(\boldsymbol{x}_1, \boldsymbol{u}_1), ..., (\boldsymbol{x}_n, \boldsymbol{u}_n)\}$ and the accuracy map classifier learns to associate each $\boldsymbol{x}$ with its full transformation map $Q(.|\boldsymbol{x})$.

1.3.5 Algorithms

Once p_θ and Q_ψ are trained, the recognition algorithm is straightforward:

Single saccade algorithm:

```
1. Read the view $\boldsymbol{x}$
2. Choose $\delta \boldsymbol{u}$ according to $Q_\psi(.|\boldsymbol{x})$
3. Move the eye
4. Update the view $\boldsymbol{x}'$
5. Identify the target with $\tilde{\boldsymbol{y}} \sim p_\theta(Y|\boldsymbol{x}')$
```

Multi saccades algorithm:

```
1. $q(Y) \leftarrow$ uniform distribution
2. Read the view $\boldsymbol{x}$
3. Choose $\delta \boldsymbol{u}$ according to $Q_\psi(.|\boldsymbol{x})$
4. Repeat several times up to some posterior confidence threshold:
    (a) Move the eye
    (b) Read $\boldsymbol{x}$
    (c) $q(Y) \leftarrow q(Y) \times p_\theta(Y|\boldsymbol{x})$
    (d) normalize $q$
    (e) Choose $\delta \boldsymbol{u}$ according to $Q_\psi(.|\boldsymbol{x})$ (with some inhibition of return mechanism)
5. Identify the target with $\tilde{\boldsymbol{y}} \sim q(Y)$
```

2 Methods

2.1 Visual transformation

2.1.1 Wavelets

2.1.2 Log Gabor

2.2 Accuracy map

2.3 Network architecture

3 Results

https://github.com/laurentperrinet/WhereIsMyMNIST/blob/master/2018-11-13-Where%20recap%20(clut

4 Discussion

4.1 Summary

4.2 Limits

4.3 Perspectives

General case: Visual information gain maximization

Consider a view $\boldsymbol{x}$ generated from a target $\boldsymbol{y}$ viewed at retinocentric position $\boldsymbol{u}$.

Consider first that :

- The generative model $p(X|\boldsymbol{y},\boldsymbol{u})$ is known
- The retinocentric position $\boldsymbol{u}$ is known.
- The view $\boldsymbol{x}$ is known.
- The target category $\boldsymbol{y}$ is unknown.

The question comes how to choose the new retinocentric position $\boldsymbol{u}'$ in order to maximize the *mutual information* between $\boldsymbol{x}|\boldsymbol{u}$ (current view) and $\boldsymbol{x}'|\boldsymbol{u}'$ (future view).

In general, the visual Information Gain between two visual fields $\boldsymbol{x}|\boldsymbol{u}$ and $\boldsymbol{x}'|\boldsymbol{u}'$ is:

$$\mathrm{IG}(\boldsymbol{x}|\boldsymbol{u};\boldsymbol{x}'|\boldsymbol{u}') = -\log p(\boldsymbol{x}|\boldsymbol{u}) + \log p(\boldsymbol{x}|\boldsymbol{u},\boldsymbol{x}',\boldsymbol{u}')$$

Information Gain Lower Bound Consider now that given $\boldsymbol{x}$ and $\boldsymbol{u}$, the target category $\boldsymbol{y}$ can be *inferred* using Bayes rule, i.e.:

$$P(Y|\boldsymbol{x},\boldsymbol{u}) \propto P(\boldsymbol{x}|Y,\boldsymbol{u})$$

Then, it can be shown (see [**?**]) that :

$$\mathrm{IG}(\boldsymbol{x}|\boldsymbol{u};\boldsymbol{x}'|\boldsymbol{u}') \geq \mathbb{E}_{\boldsymbol{y}\sim p(Y|\boldsymbol{x},\boldsymbol{u})}\left[\log p(\boldsymbol{y}|\boldsymbol{x}',\boldsymbol{u}') - \log(\pi(\boldsymbol{y}))\right]$$

with $\pi(\boldsymbol{y})$ the prior over the $\boldsymbol{y}$'s . When the prior is uniform, the information gain lower bound (IGLB) simplifies to $\mathbb{E}_{\boldsymbol{y}\sim p(Y|\boldsymbol{x},\boldsymbol{u})}\left[\log p(\boldsymbol{y}|\boldsymbol{x}',\boldsymbol{u}')\right]+c$, with c a constant.

Predictive approach One can adopt a *predictive* approach to choose the new eye orientation $\boldsymbol{e}'$:

- First choose a new retinocentric position $\boldsymbol{u}'$ that will maximize the information gain.
- Then choose $\boldsymbol{e}'$ such that
$$\boldsymbol{z} - \boldsymbol{e}' = \boldsymbol{u}'$$
i.e.
$$\boldsymbol{e}' = \boldsymbol{e} + \boldsymbol{u} - \boldsymbol{u}'$$

The predictive approach needs three predictive steps:

- $p(Y|\boldsymbol{x},\boldsymbol{u})$ is the current posterior over the target category inferred from the current observation,
- $\boldsymbol{x}' \sim p(X|\boldsymbol{y},\boldsymbol{u}')$ is the predicted view generated by the model assuming that the target $\boldsymbol{y}$ is seen from from $\boldsymbol{u}'$,
- and $p(\boldsymbol{y}|\boldsymbol{x}',\boldsymbol{u}')$ is the predicted posterior for assumption $\boldsymbol{y}$, given $\boldsymbol{x}'$ and $\boldsymbol{u}'$.

Then the optimal new retinocentric position is:

$$\hat{\boldsymbol{u}}' = \underset{\boldsymbol{u}'}{\text{argmax}}\ \mathbb{E}_{\boldsymbol{y}\sim p(Y|\boldsymbol{x},\boldsymbol{u})}\left[\mathbb{E}_{\boldsymbol{x}'\sim p(X|\boldsymbol{y},\boldsymbol{u}')}\left[\log p(\boldsymbol{y}|\boldsymbol{x}',\boldsymbol{u}')\right]\right]$$

Taking $\delta\boldsymbol{e} = \boldsymbol{u} - \boldsymbol{u}'$, the optimal eye displacement is:

$$\widehat{\delta\boldsymbol{e}} = \underset{\delta\boldsymbol{e}}{\text{argmax}}\ \mathbb{E}_{\boldsymbol{y}\sim p(Y|\boldsymbol{x},\boldsymbol{u})}\left[\mathbb{E}_{\boldsymbol{x}'\sim p(X|\boldsymbol{y},\boldsymbol{u}-\delta\boldsymbol{e})}\left[\log p(\boldsymbol{y}|\boldsymbol{x}',\boldsymbol{u}-\delta\boldsymbol{e})\right]\right]$$

(TODO : Attention il faudrait à partir de maintenant une carte qui moyenne les log posteriors car l'espérance du log n'est pas égale au log de l'espérance, i.e. $r_\theta^{log}(\boldsymbol{u}|q) = \mathbb{E}_{\boldsymbol{y}\sim q(Y)}\left[\mathbb{E}_{\boldsymbol{x}\sim p(X|\boldsymbol{y},\boldsymbol{u})}\log p_\theta(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{u})\right]$*).*