Linköping University | Department of Computer and Information Science
Master's thesis, 30 ECTS | Datateknik
2017 | LIU-IDA/LITH-EX-A--2017/001--SE

# Learning to Search in High-dimensional Signals

– with a subtitle

---

*En himla bra svensk titel*

**Oskar Lundin**

Supervisor : Sourabh Balgi
Examiner : Jose M. Peña

External supervisor : Fredrik Bissmarck

Linköpings universitet
SE–581 83 Linköping
+46 13 28 10 00 , www.liu.se

**Abstract**

The abstract resides in file `Abstract.tex`. Here you should write a short summary of your work.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Pellentesque in massa suscipit, congue massa in, pharetra lacus. Donec nec felis tempor, suscipit metus molestie, consectetur orci. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Curabitur fermentum, augue non ullamcorper tempus, ex urna suscipit lorem, eu consectetur ligula orci quis ex. Phasellus imperdiet dolor at luctus tempor. Curabitur nisi enim, porta ut gravida nec, feugiat fermentum purus. Donec hendrerit justo metus. In ultrices malesuada erat id scelerisque. Sed sapien nisi, feugiat in ligula vitae, condimentum accumsan nisi. Nunc sit amet est leo. Quisque hendrerit, libero ut viverra aliquet, neque mi vestibulum mauris, a tincidunt nulla lacus vitae nunc. Cras eros ex, tincidunt ac porta et, vulputate ut lectus. Curabitur ultricies faucibus turpis, ac placerat sem sollicitudin at. Ut libero odio, eleifend in urna non, varius imperdiet diam. Aenean lacinia dapibus mauris. Sed posuere imperdiet ipsum a fermentum.

Nulla lobortis enim ac magna rhoncus, nec condimentum erat aliquam. Nullam laoreet interdum lacus, ac rutrum eros dictum vel. Cras lobortis egestas lectus, id varius turpis rhoncus et. Nam vitae auctor ligula, et fermentum turpis. Morbi neque tellus, dignissim a cursus sed, tempus eu sapien. Morbi volutpat convallis mauris, a euismod dui egestas sit amet. Nullam a volutpat mauris. Fusce sed ipsum lectus. In feugiat, velit eu fermentum efficitur, mi ex eleifend ante, eget scelerisque sem turpis nec augue.

Vestibulum posuere nibh ut iaculis semper. Ut diam justo, interdum quis felis ac, posuere fermentum ex. Fusce tincidunt vel nunc non semper. Sed ultrices suscipit dui, vel lacinia lorem euismod quis. Etiam pellentesque vitae sem eu bibendum. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Pellentesque scelerisque congue ullamcorper. Sed vehicula sodales velit a scelerisque. Pellentesque dignissim lectus ipsum, quis consectetur tellus rhoncus a.

Nunc placerat ut lectus vel ornare. Sed nec dictum enim. Donec imperdiet, ipsum ut facilisis blandit, lacus nisi maximus ex, sed semper nisl metus eget leo. Nunc efficitur risus ac risus placerat, vel ullamcorper felis interdum. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Duis vitae felis vel nibh sodales fringilla. Donec semper eleifend sem quis ornare. Proin et leo ut dolor consectetur vehicula. Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Nunc dignissim interdum orci, sit amet pretium nibh consectetur sagittis. Aenean a eros id risus aliquam placerat nec ut lectus. Curabitur at quam in nisi sodales imperdiet in at erat. Praesent euismod pulvinar imperdiet. Nam auctor mattis nisi in efficitur. Quisque non cursus ipsum, consequat vehicula justo. Fusce varius metus et nulla rutrum scelerisque. Praesent molestie elementum nulla a consequat. In at facilisis nisi, convallis molestie sapien. Cras id ullamcorper purus. Sed at lectus sit amet dolor finibus suscipit vel et purus. Sed odio ipsum, dictum vel justo sit amet, interdum dictum justo. Quisque euismod quam magna, at dignissim eros varius in. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

# Acknowledgments

Acknowledgments.tex

# Contents

# List of Figures

# Notation

| | |
|---|---|
| $x$ | vector |
| $X$ | matrix or set |
| $\mathbb{X}$ | index set |

# 1 Introduction

In this thesis project, the problem of searching for targets in unknown but familiar environments is addressed. This chapter presents the motivation behind the project, the research questions that are addressed, and the delimitations.

## 1.1 Motivation

The ability to visually search for targets in an environment is crucial to many parts of our daily lives. We are constantly looking for things, be it the right book in the bookshelf, a certain keyword in an article or blueberries in the forest. In many cases, it is important that this search is efficient and fast. Animals need to quickly identify predators, and drivers need to be able to search for pedestrians crossing the road they are driving on.

While searching for targets is often seemingly effortless to humans, it is a complex process. How humans and animals search for things has been extensively studied in neuroscience and neurobiology [2, 5, 4]. In the computer vision field, there has been several attempts to mimic the way humans search in machines []. It is of great interest to automate visual search. Applications range from search and rescue to ...

Problematically, it is difficult to manually create search algorithms. The appearance and distribution of targets in an environment varies greatly, and may be subtle. If one could instead learn the underlying from a limited set of sample environments and generalize to unseen similar environments this problem would be circumvented.

In many real-world visual search scenarios the field-of-view is limited. This means that the search process is split into two steps: directing the field of view, and locating targets within the view. Much work has been focused on latter, locating targets within the field of view []. Often, only a fraction of the environment is visible. In these cases where to move the field of view becomes an important decision.

The characteristics of the searched environment can often be used to find targets quicker. For example, if one is foraging for blueberries it makes sense to search the ground rather than the trees. Similarly, if one is searching a satellite image for boats it is reasonable to focus on ocean shores. If you see a railroad track or the wake of a boat you can usually follow it to find a vehicle.

The exact characteristics of the environment need not be constant - forests with blueberries can vary greatly in appearance and boats can be found in all of the seven seas. In many cases, the environment is familiar in that it has characteristics that are similar to previously seen environments. Humans are able to generalize in such cases.

This work tries to address these issues, focusing on strategic scans of larger environments where the field of view is small relative to the environment. This is a problem that has been less studied in the literature than visual search in smaller environments. There are other factors that become increasingly important. The field-of-view of the observer is often limited, and she has to move it efficiently to find the target.

## 1.2 Aim

The aim of this thesis is to implement and evaluate an autonomous agent that intelligently searches its environment for targets. The agent should learn common characteristics of environments and utilize this knowledge to search for targets in new environments more effectively. Furthermore, the agent should be able to

A specific instance of the visual search problem is considered, where the environment is searched by a pan-tilt camera fixed in place. The camera has a limited view of the environent. Automating this task is of interest for multiple reasons. Manually controlling a camera may be costly, and the performance of a human operator may be suboptimal. Crucial to the problem is generalization.

## 1.3 Research questions

This thesis will address the following questions:

1. How can a learning agent that does efficient visual search in familiar environments be implemented?
2. How can a simulator that tests the ability of an agent to solve the presented problem be implemented?
3. How can a learning agent generalize to unseen but familiar environments?
4. How does memory affect the agent's ability to search an environment?
5. How does the learning agent compare to common non-learning methods?
6. How does the learning agent compare to an exhaustive search of the environment, frontier-based algorithm, and a human searcher?

## 1.4 Delimitations

This thesis will be focused on the behavioral aspects of the presented problem. To train and test agents, a simplified environment will be used. This will test the desired characteristics of the agent as presented above, but will not simulate realistic environments.

# 2 Theory

This chapter introduces relevant theory and related work

## 2.1 Visual Search

Visual search is a perceptual task which involves seeking out targets among distractors. Eckstein (2011) [2] identifies four factors that limit performance of visual search in animals:

- Foveated vision.
- Variability in visual environment and uncertainty about target parameters.
- Stochasticity of neural processing.
- Limitations of covert attention and memory.

The brain utilizes a set of strategies to optimize visual search performance:

- Calculation of the visibility of different regions (saliency).
- Knowledge about the visual properties of the environment, including targets, distractors and noise.

There are a set of statistical regularities that reduce uncertainty of the target location:

- Target probabilities varying across locations and predictive cues.
- Contextual cuing

Interestingly, studies in humans have shown

- We do not use memory during visual search
- We have easier to differentiate unknown distractors from targets (or vice versa?)

An alternative model of visual search is *guided search* by Wolfe (2021).
Wolfe also presents a simulation of the model. Simulates some mechanics of the search

### 2.1.1 Active Vision

Much of current research in computer vision studies problems with passive observers where images are passively sampled. The problem considered in this project contains an *active vision* [1] system. The observer (agent) can manipulate the viewpoint of the camera in order to investigate the environment and get better information from it. This closer to human perceptual activity which is both exploring and searching. A study of basic problems of vision shows that active vision systems have several computational advantages over passive ones for common perceptual tasks [1].

Caicedo and Lazebnik (2015) [**aol**] propose to use deep reinforcement learning for active object localization in images where the object to be localized is fully visible. An agent is trained to successively improve a bounding box using translating and scaling transformations. They use a reward signal that is proportional to how well the current box covers the target object. An action that improves the region is rewarded with +1, and -1 otherwise. Without this quantization, the difference was small enough to confuse the agent. Binary rewards communicate more clearly which transformations keep the object inside the box and which take the box away from the target. When there is no action that improves the bounding box, the agent may select a trigger action (which would be the only action that does not give a negative reward) which resets the box. This way the agent may select additional bounding boxes. Each trigger modifies the environment by marking it so that the agent may learn to not select the same region twice. This is referred to as an inhibition-of-return mechanism, and is widely used in visual attention models [**16 in aol**]. This method has a few shortcomings for the problem considered in this project. The object may not be visible in the initial frame so the agent cannot act in the same way.

## 2.2 Reinforcement Learning

Reinforcement learning (RL) is a subfield of machine learning concerned with learning from interaction how to achieve a goal. An *agent* and its *environment* interact continually over discrete time steps. At each time step the agent selects some *action* that updates the state of the environment, and gives it a *reward*. The agent selects actions using a stochastic *policy* with the goal of maximizing the *return* which is usually defined as the discounted sum of future rewards.

### 2.2.1 Markov Decision Process

The RL setup is usually formalized as a (finite) Markov decision process (MDP).

The problem of learning from interaction to achieve a goal is usually framed as a (finite) Markov Decision Process (MDP). For regular MDPs it is assumed that the learning agent has access to some representation of the underlying *state* of the environment which it uses to select *actions*. For many problems this is not true. A partially observable Markov decision process (POMDP) is a generalization of an MDP in which it is assumed that the environment has some well defined underlying latent state, but the agent only perceives a partial *observation* of it from the environment.

A POMDP is formally defined as a 7-tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{O}, \mathcal{R}, \mathcal{T}, \Omega, \gamma \rangle$, where

- $\mathcal{S}$ is a finite set of states,
- $\mathcal{A}$ is a finite set of actions,
- $\mathcal{T}$ is a set of conditional state transition probabilities,
- $\mathcal{R} : S \times A \rightarrow \mathbb{R}$ is a reward function,
- $\Omega$ is a finite set of observations,

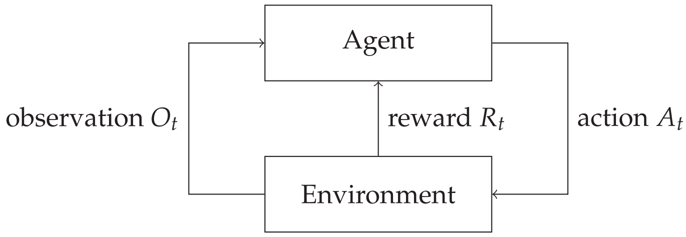


Figure 2.1: Partially observable Markov decision process.

- $\mathcal{O}$ is a set of conditional observation probabilities, and
- $\gamma \in [0, 1]$ is a discount factor.

The agent interacts with the environment at discrete time steps $t = 0, 1, 2, \ldots T$. At each time step $t$, the agent receives an observation of the environment's state $O_t \in \Omega$ and selects some action $A_t \in \mathcal{A}$. In the next time step the agent receives a reward

action $a \in \mathcal{A}$ which causes the environment to transition to state $s'$ with probability $\mathcal{T}(s'|s, a)$. It receives an observation $o \in \Omega$ with probability $\mathcal{O}(o|s', a)$, as well as a reward $r$ given by $\mathcal{R}(s, a)$.

This interaction is repeated until the end of the episode at time step $T$. The goal of the agent is to maximize the *discounted return*, defined as the discounted sum of future rewards $G_t \doteq \sum_{k=t+1}^{T} \gamma^{k-t-1} R_k$ where $\gamma$ reflects the uncertainty of the environment.

Planning in a POMDP is undecidable, and solving them is often computationally intractable. Approximate solutions are more common.

### 2.2.2 Policies and Value Functions

Most RL algorithms estimate both a *value function* that tells the agent how good it is to be in a given state, and a

### 2.2.3 Policy Optimization

This work will focus on policy optimization algorithms.

### 2.2.4 Actor Critic Models

### 2.2.5 Exploration and Exploitation

### 2.2.6 Generalization

Kobbe et al. (2020) [] study generalization in RL. They introduce a benchmark of procedurally generated i.i.d. environments, and find that this is essential to

## 2.3 Automating Visual Search

## 2.4 Deep Reinforcement Learning

Ghesu et al. () [3] use XXX

# 3 Method

In this chapter, the method used is described. Section 3.1 formalizes the problem solved. Section 3.2 details the environment used to train and test an agent. Section 3.3 describes the algorithm used to train the agent.

## 3.1 Problem Statement

The problem of finding an optimal sequence of actions to find can be cast as a partially observable Markov decision process (POMDP). The environment's state is described by an $n$-dimensional matrix $S$ whose members are drawn from some unknown distribution. In the environment, there are $N$ targets $\mathbb{T}$.

The agent observes a window $\mathbb{W}$ of this matrix. The actions available to the agent change the members of the window $\mathbb{W}$ but not the number of elements.

Every camera move is associated with a cost, and it is therefore important that the targets are identified as quickly as possible.

The aim of this project is to learn to search efficiently in the visual domain with a pan-tilt-zoom camera. To make the problem more manageable, it is approximated as follows. The environment state is an RGB image of shape $(3, H, W)$. The window is fixed to $(3, 64, 64)$ and can be moved row-wise and column-wise in the image. This emulates the pan-tilt behaviour of a camera.

## 3.2 Environment

To train a learning agent, an environment for search was implemented. The searched environment is approximated by an RGB image with dimensions $(3, H, W)$. The behaviour of a pan-tilt-zoom camera is approximated by giving the At any given time, only a $(3, 64, 64)$ subregion of the image is observable to the agent.

The environments to be searched are drawn from a distribution, with varying but similar appearance, target locations and appearances. For all environments, the appearance correlates to the probability of targets.

To incentivize finding targets quickly, the reward signal is set to -1 for each time step. Since viewing a window twice is redundant, such actions are punished by setting the reward to -2.

If the agent selects the trigger action when a target overlaps with the window, the reward is set to 5. When all targets have been triggered, or when 1000 time steps have passed, the episode ends.

## 3.3 Algorithm

The

The agent is trained with reinforcement learning

## 3.4 Implementation

The environment is implemented with Gym [**gym**], and the agent is implemented with PyTorch [**pytorch**].

## 3.5 Experiments

The first environment was used to determine a good observation space. By just observing the current window, the agent can never learn a suitable policy to solve the problem. This is because it cannot distinguish between equal windows at different locations. Experiments are run with several additional observation types: window position, .... Results of these experiments are presented for this environment only.

The agent was trained using the algorithm described in Section 3.3 for 100 million time steps in all three environments. Hyperparameters are tuned with random search separately for each environment. For all experiments, the average return per episode is reported together with the theoretically optimal reward (obtained with an optimal path).

Additionally, experiments to eveluate the generalization capability of the agent were conducted. These were conducted on the procedurally generated terrain environment following the approach suggested in [**procgen**]. During training, the seed pool size was fixed to various sizes to limit the training set size. The agent was trained for varying number of timesteps and then evaluated on the full distribution of environments.

All experiments are conducted on an Intel Core i9-10900X CPU and an NVIDIA GeForce RTX 2080 Ti GPU.

### 3.5.1 Memory

# 4 Results

This chapter presents the results. Note that the results are presented factually, striving for objectivity as far as possible. The results shall not be analyzed, discussed or evaluated. This is left for the discussion chapter.

In case the method chapter has been divided into subheadings such as pre-study, implementation and evaluation, the result chapter should have the same sub-headings. This gives a clear structure and makes the chapter easier to write.

In case results are presented from a process (e.g. an implementation process), the main decisions made during the process must be clearly presented and justified. Normally, alternative attempts, etc, have already been described in the theory chapter, making it possible to refer to it as part of the justification.

# 5 Discussion

This chapter contains the following sub-headings.

## 5.1 Results

Are there anything in the results that stand out and need be analyzed and commented on? How do the results relate to the material covered in the theory chapter? What does the theory imply about the meaning of the results? For example, what does it mean that a certain system got a certain numeric value in a usability evaluation; how good or bad is it? Is there something in the results that is unexpected based on the literature review, or is everything as one would theoretically expect?

## 5.2 Method

This is where the applied method is discussed and criticized. Taking a self-critical stance to the method used is an important part of the scientific approach.

A study is rarely perfect. There are almost always things one could have done differently if the study could be repeated or with extra resources. Go through the most important limitations with your method and discuss potential consequences for the results. Connect back to the method theory presented in the theory chapter. Refer explicitly to relevant sources.

The discussion shall also demonstrate an awareness of methodological concepts such as replicability, reliability, and validity. The concept of replicability has already been discussed in the Method chapter (3). Reliability is a term for whether one can expect to get the same results if a study is repeated with the same method. A study with a high degree of reliability has a large probability of leading to similar results if repeated. The concept of validity is, somewhat simplified, concerned with whether a performed measurement actually measures what one thinks is being measured. A study with a high degree of validity thus has a high level of credibility. A discussion of these concepts must be transferred to the actual context of the study.

The method discussion shall also contain a paragraph of source criticism. This is where the authors' point of view on the use and selection of sources is described.

In certain contexts it may be the case that the most relevant information for the study is not to be found in scientific literature but rather with individual software developers and open source projects. It must then be clearly stated that efforts have been made to gain access to this information, e.g. by direct communication with developers and/or through discussion forums, etc. Efforts must also be made to indicate the lack of relevant research literature. The precise manner of such investigations must be clearly specified in a method section. The paragraph on source criticism must critically discuss these approaches.

Usually however, there are always relevant related research. If not about the actual research questions, there is certainly important information about the domain under study.

## 5.3 The work in a wider context

There must be a section discussing ethical and societal aspects related to the work. This is important for the authors to demonstrate a professional maturity and also for achieving the education goals. If the work, for some reason, completely lacks a connection to ethical or societal aspects this must be explicitly stated and justified in the section Delimitations in the introduction chapter.

In the discussion chapter, one must explicitly refer to sources relevant to the discussion.

# 6 Conclusion

This chapter contains a summarization of the purpose and the research questions. To what extent has the aim been achieved, and what are the answers to the research questions?

The consequences for the target audience (and possibly for researchers and practitioners) must also be described. There should be a section on future work where ideas for continued work are described. If the conclusion chapter contains such a section, the ideas described therein must be concrete and well thought through.

# Bibliography


[1] John Aloimonos, Isaac Weiss, and Amit Bandyopadhyay. "Active vision". In: *International Journal of Computer Vision* 1.4 (Jan. 1988). 705 citations (Crossref) [2022-02-07], pp. 333–356. ISSN: 0920-5691, 1573-1405. DOI: 10/cn4mdc. URL: http://link.springer.com/10.1007/BF00133571 (visited on 02/07/2022).

[2] M. P. Eckstein. "Visual search: A retrospective". In: *Journal of Vision* 11.5 (Dec. 30, 2011). 207 citations (Crossref) [2022-02-28], pp. 14–14. ISSN: 1534-7362. DOI: 10.1167/11.5.14. URL: http://jov.arvojournals.org/Article.aspx?doi=10.1167/11.5.14 (visited on 02/22/2022).

[3] Florin-Cristian Ghesu, Bogdan Georgescu, Yefeng Zheng, Sasa Grbic, Andreas Maier, Joachim Hornegger, and Dorin Comaniciu. "Multi-Scale Deep Reinforcement Learning for Real-Time 3D-Landmark Detection in CT Scans". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41.1 (Jan. 1, 2019), pp. 176–189. ISSN: 0162-8828, 2160-9292, 1939-3539. DOI: 10.1109/TPAMI.2017.2782687. URL: https://ieeexplore.ieee.org/document/8187667/ (visited on 02/03/2022).

[4] Jeremy M. Wolfe. "Guided Search 6.0: An updated model of visual search". In: *Psychonomic Bulletin & Review* 28.4 (Aug. 2021). 29 citations (Crossref) [2022-03-02], pp. 1060–1092. ISSN: 1531-5320. DOI: 10.3758/s13423-020-01859-9.

[5] Jeremy M. Wolfe. "Visual search". In: *Current biology : CB* 20.8 (Apr. 27, 2010). 64 citations (Crossref) [2022-03-02] Publisher: NIH Public Access, R346. DOI: 10.1016/j.cub.2010.02.016. URL: https://www.ncbi.nlm.nih.gov/labs/pmc/articles/PMC5678963/ (visited on 03/02/2022).