

Universität des Saarlandes
Fakultät für Mathematik und Informatik
Fachrichtung Informatik

Bachelorarbeit

Visual Feature Extraction with Convolutional Neural Networks
for Search Target Inference

vorgelegt von

Sven Oliver Stauden

am 09.04.2018

Begutachtet von:

Dr. Daniel Sonntag

Prof. Dr. Dr. h. c. mult. Wolfgang Wahlster

Erklärung

Ich erkläre hiermit, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Statement

I hereby confirm that I have written this thesis on my own and that I have not used any other media or materials than the ones referred to in this thesis

Einverständniserklärung

Ich bin damit einverstanden, dass meine (bestandene) Arbeit in beiden Versionen in die Bibliothek der Informatik aufgenommen und damit veröffentlicht wird.

Declaration of Consent

I agree to make both versions of my thesis (with a passing grade) accessible to the public by having them added to the library of the Computer Science Department.

Saarbrücken, _____
(Datum/Date)

(Unterschrift/Signature)

Contents

Declaration	iii
Abstract	vi
1 Introduction	1
1.1 Human Gaze and Attention	4
1.1.1 The Human Eye	4
1.1.2 Modeling Visual Attention	6
1.1.3 Visual search and Search Target Inference	8
1.2 Machine Learning Background	10
1.2.1 The Learning Process	11
1.2.2 Neural Networks	11
1.2.3 Convolutional Neural Networks	13
2 Related Work	16
2.1 Visual Search Target Inference	16
2.2 CNN Features for the Representation of Visual Data	18
3 Datasets	21
3.1 VIU Dataset	21
3.2 Amazon Book Cover Dataset	23
4 CNN Feature Extraction	26
5 Feature Space Analysis & Spatial Target Inference	29
5.1 Spatial Feature Relations	29
5.1.1 Fixation Cluster Analysis	29
5.1.2 Direct Relation between Fixations & Target	33
5.1.3 Temporal Relation between Fixations & Target	35
5.2 Spatial Inference	37
5.2.1 Fixation Aggregation Methods	37
5.2.2 Aggregation Target Relation	40
5.2.3 Search Target Inference by Vector Similarity	42
5.2.4 Target Inference by learned distance metric	49

6	Bag of Neural Network Features	53
6.1	Closed World Inference by Sattar et al.	54
6.2	Bag of Words Adaption using CNN Features	56
6.3	Semantic Supported Inference	58
7	Conclusion	62
	References	64
	Literature	64
	Online sources	66

Abstract

Human gaze reveals valuable insights to attention and cognitive processes. A person looking for an object in a scene or an image, which is referred to as visual search, compares the visual input from his eyes with an abstraction of this object in his mind. If these representations coincide, the person considers the object as found. However, the way the brain abstracts visual data, to be able to perform such a search-target mapping, is difficult to reconstruct. As convolutional neural networks get trained to extract task relevant features from visual data, this work introduces and analyzes the possibilities to use a pre-trained network to encode human gaze data for search target inference.

The general goal is to exploit the high information concentration of these encodings in order to reveal relations between human gaze behavior during visual search and features of the searched objects. Therefore, different strategies to process gaze encodings as well as the application to an existing target inference approach get presented and evaluated. Further, with the involvement of an intelligent image segmentation procedure, the beneficial impacts of respecting semantical relations between objects in the search space get shown.

The findings of this work are not only relevant for intelligent gaze-based human computer interaction systems, but can also be applied to other user attention inference approaches.

Chapter 1

Introduction

The human visual system performs sophisticated and important tasks in almost every situation. With more than one million nerve fibers, visual signals get carried from one eye over a part of the brain called lateral geniculate nucleus [15] to the visual cortex [30]. The strong evolved linkage of the visual system to the human brain, underlines its general importance. For humans, the distinction of colors, brightness, moving and stationary objects, as well as three dimensional behaviors played an important role due to the high variability of life circumstances, already in the early years of mankind.

During waking hours, our eyes are almost constantly active performing different tasks to perceive, often unconsciously, information about the environment. One of these tasks, in which eyes and the neural system strongly work together, is the visual search. During the process of analyzing the perceptible surrounding, in order to find a certain object, the attention is driven by the motivation of finding.

Early studies showed, that regions, which have been focused during search, are not random [2, 24–27]. Figure 1.1 showing an overview of multiple car images gives an example for the occurrence and effect of visual search. For the task "Find the image with a red Smart!", a user usually does not require to look at each displayed thumbnail separately but finds the target image after a few moments. This is possible because the attention gets automatically attracted by red objects and later by features defining the significant shape of a Smart model which reduce the search time enormously. From 2009 to 2012, Microsoft's search engine Bing made use of the benefits of visual search by displaying query results of an image search in an overview mode as showed in [28] to improve search time performance and user experience.

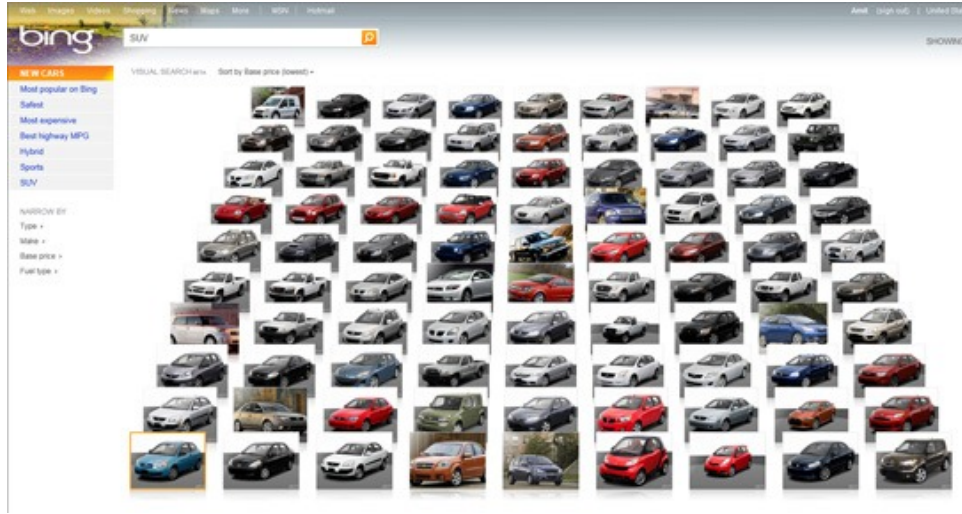


Figure 1.1: Screenshot of Bing's "Visual Search" option (2009-2012) that allowed users to get an overview of multiple images. Due to the task driven gaze behavior, finding a "red Smart" for example is possible without considering each image individually.

The locations that attract the cognitive system during the seeking process seem to relate in a certain way to the searched target object [2, 24, 27]. General search target inference, tries to identify such relations by analyzing the search process in order to gain information about the target object. This can be helpful e.g. to make intelligent systems follow and understand a user's intentions in order to support him in his tasks. To find a relation between gaze behavior and target, visual features at the considered locations in the image and at the searched object have to get considered. Because visual features appear in different forms like color, shape, pattern, etc., it is challenging to find a proper way to encode them.

This work presents and analyzes two major approaches that use human gaze data recorded during a visual search in order to predict the searched target objects. Facing the problem of accounting visual features, these methods use trained artificial neural networks for extracting information from visual data.

Using data from eye tracking devices that captured a person's gaze, prior studies performed search target inference using different methods for feature extraction. Considering color histograms [2], SIFT features [5] or Bag of visual Word encodings [17] already delivered promising results.

The approaches presented in this work extract visual features with convolutional neural networks, that have actually been trained for object classification on images. Considering two datasets, multiple experiments test the

possibilities and performances of these systems as beneficial alternative to existing methods.

Usually, convolutional neural networks (short: CNNs) get trained to extract relevant features from input data, find existing correlations and finally apply a certain task on them. Exploiting the network’s ability to highlight important image features might open new doors for search target inference. Therefore, this work introduces the two target inference concepts of Spatial Target Inference and an adaption of the popular Bag of Words encoding, both using CNN features.

For Spatial Inference, the general representation of the extracted feature data in their corresponding vector space gets studied under several aspects. Investigating the spatial structures of encoded gaze and target representations clarifies existing correlations which can be used to setup useful prediction models.

The Bag of Words approach is aligned to the work of Sattar et al. [17] who performed search target inference on a finite set of target alternatives. This idea gets extended by using CNN features. Further, an additional neural network gets involved which delivers a semantic segmentation of the search image in order to increase the applicability of the presented algorithm. Moreover, this approach clarifies the general impact of the semantic context of a search image.

For a better understanding, basic concepts of the human visual system as well as a brief introduction to machine learning and neural networks are provided in the subsequent sections. After presenting prior research related to search target inference and CNN features, the main part of this work is structured into three parts: The first section explains the general feature extraction process with convolutional neural networks, which gets used for all subsequent approaches. Afterwards, feature encodings from gaze data of two different datasets get extracted to study structures and relations in the CNN feature space. These relations will then be used to setup target inference models, that consider distance measures of feature vectors. The Bag of Words approach by Sattar et al. [17] gets explained in detail subsequently, to compare it with the novel methods using CNN features and semantic segmentation.

1.1 Human Gaze and Attention

When intercepting information from the sense organs the cognitive system decides within milliseconds whether a stimulus is important and has to be further observed or can be ignored. This procedure is strongly required as otherwise the brain would have to process an overload of data. However, even if one does not seem to be concentrated on a certain task, general disturbing information, that are not necessarily needed, get blurred out by the brain [3, 7].

The perception of visual stimuli while performing a specific task differs significantly from the behavior in situations without the need for explicit attention. This difference can be measured by analyzing the movements of the eyes, which are mostly controlled by the cognitive system. For this reason, eye behavior during an attention driven situation may reveal information about the ongoing task [16, 26].

For target inference, it is essential that visual search is highly attention and task driven. In the following, basic concepts of the human visual system as well as findings of gaze-attention relations get introduced.

1.1.1 The Human Eye

The human eye consists of several components that are required to perform different actions in order to deliver important visual information about the current environment. With individual adaptations of these components, the visual system captures stimuli that get processed by the neural system to objects, relation understandings, three dimensional vision, etc.

A visual stimulus, which initially is nothing different than incoming light from the pupil through the lens, gets absorbed by the light sensitive coat at the inner side of the eye ball, called retina (see figure 1.2). It consists of cells called rods and cones which convert light into an actual stimulus signal. Rods, the more sensitive receptors that can already be triggered by single photons, are responsible for vision in low light situations but only deliver blurry and weak colored information. In contrast, visual perceptions of well lighted conditions are the product of cones. Therefore, a higher amount of photons is needed for activation but a more detailed color vision results [31]. The approximately 120 million rods and 6 million cones are not equally distributed on the retina [18]. The point located almost directly opposite the lens, called fovea centralis (short: fovea), does not contain any single rod but has the highest concentration of cones. For this reason, the best visual acuity is reached at this spot.

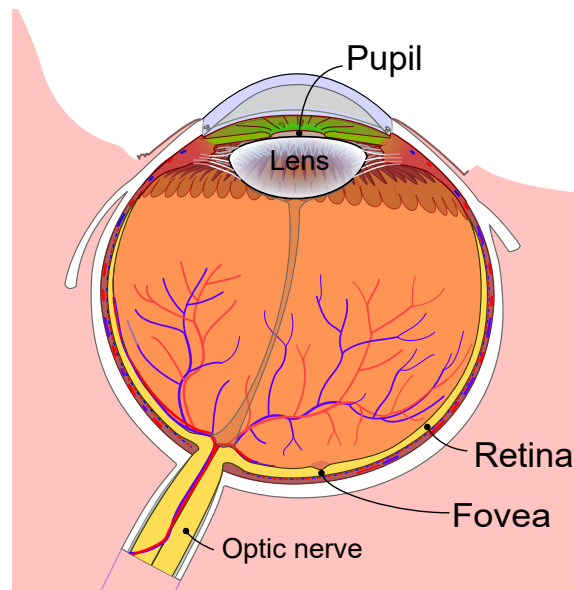


Figure 1.2: Simplified model of a human eye. The pupil and the lens regulate the incoming light that gets projected onto the retina. The fovea is the most sensitive spot on which stimuli of focused objects get projected.

Summarized, sharp vision for almost all human tasks is related to the correct projection of incoming light onto the fovea (see figure 1.2). Correct angle, light absorption and refraction, unconsciously controlled by flexible components of the eye, lead to unblurred visual perceptions of selected objects. Stimuli that reach this point are associated to lie in the visual and cognitive focus of an observer [25].

Usually, the attention is not restricted to one single object or spot, but it assembles from information of multiple impacts. In order to process all relevant information, the human eye is able to perform various movements for different requirements.

As introduced before, visual and cognitive focus is achieved by directing eye components to a state, so that the light coming from the desired object gets sharply projected onto the fovea. When the projection is in place, it seems that the eyes stay in a static position for a very short moment. These situations are called fixations (figure 1.3 left).

Analyzing one single location does not deliver sufficient information about the surrounding environment. Fixating multiple objects successively requires to change the point of focus. This happens with a ballistic [9] and simultaneous [4] eye movement, which is called saccade (figure 1.3 left). Due to the high velocity, the eyes do not deliver a sharp but blurry image, called saccadic suppression. A saccade only lasts for a very short time and ends up

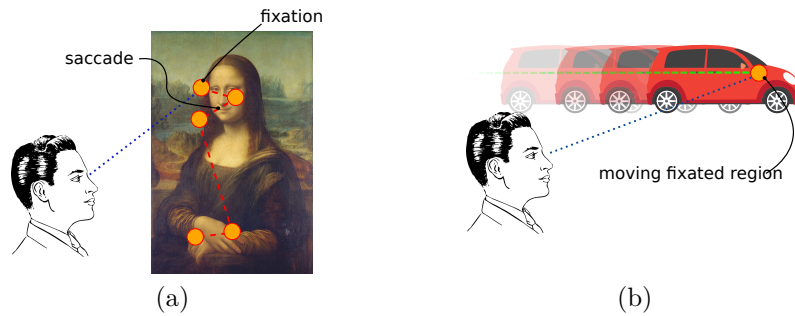


Figure 1.3: (a) A saccade is the eye movement between two fixations when considering motionless objects. (b) The behavior of tracking a moving object is called "smooth pursuit".

with a fixation when the eyes stop at a certain point [9].

Often, nature also requires to focus on moving objects. The human visual system evolved the so called pursuit shift movement (also called smooth pursuit movement), where the eyes are constantly fixated to an object while it is moving [9] (figure 1.3 right). This type of eye movement is important for attention measurement in moving scenes but less for static images. For these, the analysis of fixations delivers information about locations which attracted the observer and therefore where his attention has been paid.

The process of capturing a person's gaze, is called eye tracking, and is nowadays usually performed with eye tracking devices. Out of the resulting data, fixations reveal the objects or regions a person has been focused. Analyzing these, might disclose interesting information about the person's attention and intention.

1.1.2 Modeling Visual Attention

Much effort has already been spent on the research of human behavior in various situations. Regarding to the observation of a test person's eye movements during a certain task, the important fact came up, that eyes behave more or less predictably and follow a non-random choice of fixations [24, 25]. Eye movements are highly related to the activities a person is currently performing. These gaze impacts are condition depending and usually get specified as "top-down". In contrast, visual features that attract the human focus naturally without the influence of any specific task are called "bottom up" features.

Richard Gregory [8] explains bottom-up processing as stimulus driven consideration of visual input. Specific sensory information reaching the eye can affect the attention automatically. Bottom-up investigations consider the general probability of an image part to be fixated by the human gaze. This

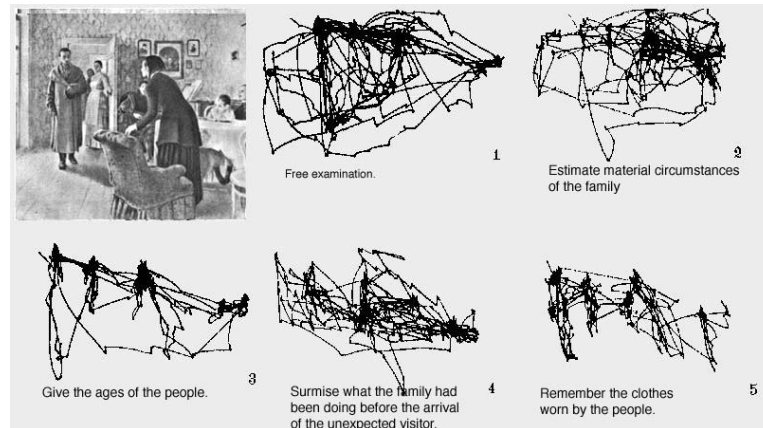


Figure 1.4: Yarbus [25] showed that people observe same images differently, depending on the task. The fixated locations reveal the most attracting image regions.

type of saliency does not depend on a certain task the viewer tries to achieve but on low-level image features and natural interests in certain objects. Human faces and written letters often provide concentrations of informational load which makes them more attractive to consider than objects that seem to be uninteresting. A person considering a natural scene without any tasks or presumptions (free viewing task) recognizes or at least perceives objects containing bottom-up features first [1].

When, in contrast, an observer looks at an image with a certain presetting, the points of consideration relate to the intention of the person and one speaks of top-down processing. In 1967, Yarbus [25] used eye tracking techniques to analyze scan paths of test persons which had to perform different tasks that have been related to shown images (figure 1.4). In the experiments, the characterization of the asked information for every task was different. Image spots that seem to reveal the highest gain on information for the currently intended task, are very likely to get fixated. One says that these locations contain top-down features that unnoticeably attract a test person's gaze [1, 24, 26].

Summed up, in most cases, the human attention is task driven. Attention directs eye movements by fixating the locations that seem to provide important information for the task. This work analyzes and makes use of these theoretical relations for visual search as task option which get further described in the following section.

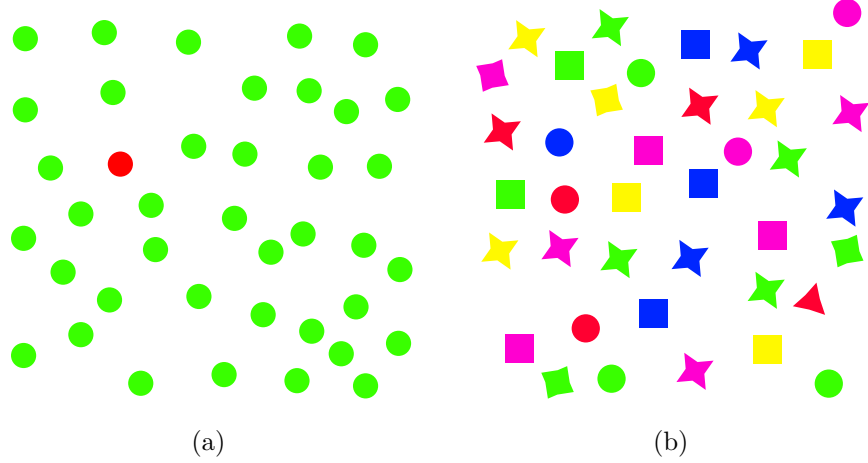


Figure 1.5: (a) Finding the red circle is very easy as the color is an outstanding feature the gaze directly gets attracted by. (b) For finding the only triangle in the image, considering the color feature does not bring any benefit but the consideration of shapes.

1.1.3 Visual search and Search Target Inference

The process of finding a certain object by analyzing visual data is designated as visual search task and underlies, like almost all tasks, an effective amount of attention and therefore describes a top-down process [26]. By only considering data captured during the search process, search target inference aims to reconstruct information about the target object. As stated in the previous section, fixations during top-down processing do follow specific rules and in this case are likely to relate to the task-defining and attention-shaping target object.

The assumption, that locations at fixated regions and at the target object contain related or even similar visual features, gets also reflected in the discussed "red smart" example where one common feature of fixations and target may be the red color (see figure 1.1).

The main problem of feature consideration and comparison is, that the feature type can not be generally specified. Searching for a red circle among green distractor items, the feature of color directly leads the gaze to the target (see figure 1.5 a). In contrast, when searching for a triangle among different colored and shaped objects, the color feature does not support the search at all unlike concentrating on the objects' contours (see figure 1.5 b). Obviously, gaze and target features depend on the task but also on further impacts like the target itself and its semantical context. Depending on the type of the target object, the relevance for various areas in the search im-



Figure 1.6: (a) Google Street View screenshot. About half of the image shows sky and the other half displays road. (b) Searching for an airplane restricts the search space to the semantically more attractive region (blue) as airplanes are used to be located in the sky. The red area of the image illustrates the less interesting part where less fixations during the search will occur.

age might decrease, so that the search space can actually be restricted. For example, as airplanes usually are located in the sky, for the task of finding an airplane in a photography, non-sky segments in the search image get less relevant (see figure 1.6). Therefore, also the impact of semantic information can be exploited to gain information about the search target which gets further analyzed in section 6.3.

Persons conducting the search tasks might apply different strategies or get attracted by diverse features due to various experiences or relations to the target. Therefore, it is likely that an inference strategy cannot be applied to all persons in the same way. For this reason, it makes sense to consider the fixation behavior of observers separately (within-user inference) [17].

One additional factor, which is especially important for planning and conducting experiments on visual search, is the search target representation in the observer's mind. Showing an image of the target before the search, highlights almost all features relating to the reference image (figure 1.7 a). The search is then highly influenced by the appearance of the shown object. This can be an advantage, when the target really resembles the reference. Otherwise, the clue image can lead to erroneous assumptions of the target and can complicate the search. For this work this "image driven" approach contrasts with the so called "cue driven" visual search task, where observers are instructed to look for an object for which a certain keyword was given like in figure 1.7 b). Unlike the "image driven" task, the "cue driven" search allows some room for interpretation, fantasy and experience of the observer.

The presented diversity of visual features leads to the general problem to find an appropriate and representative encoding. Search target inference as described in the following approaches, necessarily requires persistent feature encodings. Because simple feature definitions for inference are neither clear



Figure 1.7: (a) In image driven search, the observing person has seen the target and so also recognized its visual features which can be compared to occurring features in the search image. (b) In cue driven search, a person only gets an idea of the target object. Visual features that relate to this target are constructed by the person.

nor sufficient, using machine generated feature encodings might deliver useful results. Deep Learning models (so called Neural Networks) are able to learn recognizing features of given data that are relevant [20].

1.2 Machine Learning Background

Machine learning is one important field of artificial intelligence and data science. Many applications rely on structures and relations in data collections which are usually hard for humans to recognize. Machine learning models learn these relations to later apply them on unseen data and predict certain properties. In general, data X relates to a certain property (often called label) y so that $f(X) = y$, where f describes the relation between data and property. Machine learning tries to find a good approximation \hat{f} for f by minimizing the difference $y - \hat{f}(x)$ numerically over multiple iterations.

The interest of this work lies in the consideration of the relation between fixation of human gaze data and a searched target object.

This section provides a basic background of relevant the machine learning fields for this work. The general pipeline to train a predictive model, the concept of classification and the idea of neural networks get explained briefly.

1.2.1 The Learning Process

Training and applying models to data collections requires to process a pipeline of multiple steps, which partially also have been conducted for this work.

First of all, the considered data have to be created or collected. The resulting information get stored to simple collections. Afterwards, the data have to be adapted to the considered problem. Missing, not needed or erroneous data samples get handled and if needed, additional information can be included. Data that describe a single object instance are called "samples" while the actual data values of a sample are called "components" or "features". In general, an additional property belongs to each data sample, called label. Usually, the task of a machine learning model is to predict a label y for a given data sample x .

Training a model happens by optimizing a hypothesis function \hat{f} . Therefore, the data samples with known label values get distributed to two distinct sets - one for training and one for testing. Samples from the training set get used to adapt the hypothesis function \hat{f} so that the cost value $c = \hat{f}(x_{train}) - y_{train}$ gets minimized. After training, \hat{f} should imitate the relation between data samples and labels well.

To test, whether the approximation generalizes, labels of samples from the test set get predicted. The accuracy of these predictions describe the general performance of the model. Labels, that do not correlate to data samples, cannot be predicted as there exists no relation f . In these cases, the model accuracy is close to the probability of predicting the labels by chance. Depending on the test performance, the resulting model can be used to predict label values for new data.

The data for this work preliminarily consist of annotated images and eye tracking captures which have been already collected in previous studies (see 3). The later introduced methods aim to map information of the search target or the target itself to the representation of the corresponding gaze fixations.

1.2.2 Neural Networks

Neural networks are architectures that belong to a sub topic of machine learning called deep learning. Neural networks are inspired by the setup of a natural brain consisting of connected neurons that process electric stimuli. Implementations of artificial neural networks reach outstanding performances for machine learning tasks.

Neural networks are organized in layers. One layer can be seen as matrix or tensor containing numerical values called units. Each unit of one layer

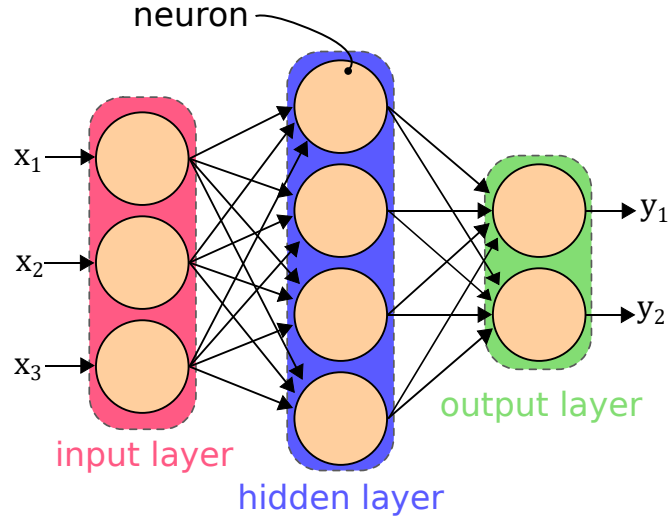


Figure 1.8: A visualization of a two layer fully connected neural network. Each layer consists of several units while each unit is connected to all neurons of the subsequent layer. The network processes three dimensional input data through a hidden layer with four neurons and returns a two dimensional result vector in the output layer.

is connected to units of the neighboring layer. There are mainly three layer types to be distinguished: input layer, hidden layers and output layer (see figure 1.8).

Actually, neural networks are nothing different than multiple function applications to the values of the vector in the input layer. Processing the data from one to the subsequent layer, is performed by conducting a matrix multiplication of the input vector with a layer specific weight matrix θ_i . After applying a so called activation function a_i to the result, the process gets repeated with the weight matrix of the next layer until the last layer is reached. During training a neural network, the values of the weight matrices get adapted so that the network returns the wished outputs after a certain training time.

Simplified, general neural networks are machine learning models approximating data relations with a hypothesis function \hat{f} so that:

$$\hat{f}(x) = a_n(\dots(a_1(a_0(x \cdot \theta_0) \cdot \theta_1), \dots) \cdot \theta_n) \stackrel{!}{\approx} y = f(x)$$

with input data x , activation functions a_i and weight matrices θ_i .

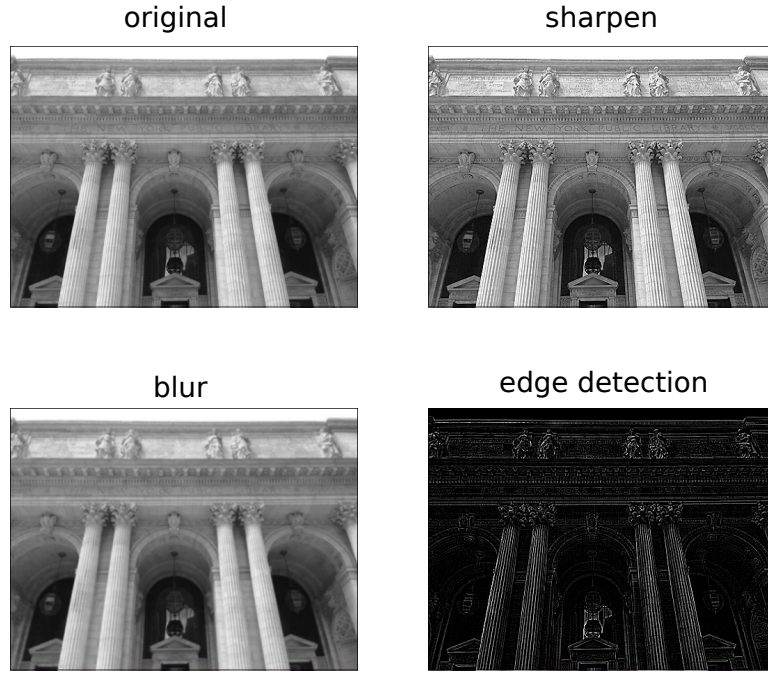


Figure 1.9: Resulting outputs of an image convolved with different filter functions. Depending on the filter, different visual features get highlighted.

1.2.3 Convolutional Neural Networks

There are different types of neural networks that get distinguished by the way of processing data. The neural network which was described in the previous sections is called "Fully Connected" because all units are linked to their neighbors in each layer. For image data, it makes sense, to highlight visual components, that are important for the learning task first. This can be accomplished by applying a convolution with a specific filter.

In general, a convolution is an operator which takes two functions f and g as input and returns the function $f * g$ of the following form:

$$(f * g)(x) := \int_{-\infty}^{\infty} f(r)g(x - r) dr$$

In image processing convolutions are often performed with f as function that represents the pixel color values at a certain position of an image and g as a so called kernel or filter. Intuitively, the filter, which is actually a n -dimensional weight matrix, gets laid over each image pixel. Accumulating the products of the pixel values of the neighbor pixels and their respective weightings delivers the pixel values of the resulting convolved image. Depending on the weights in the filter, specific image features like edges, color

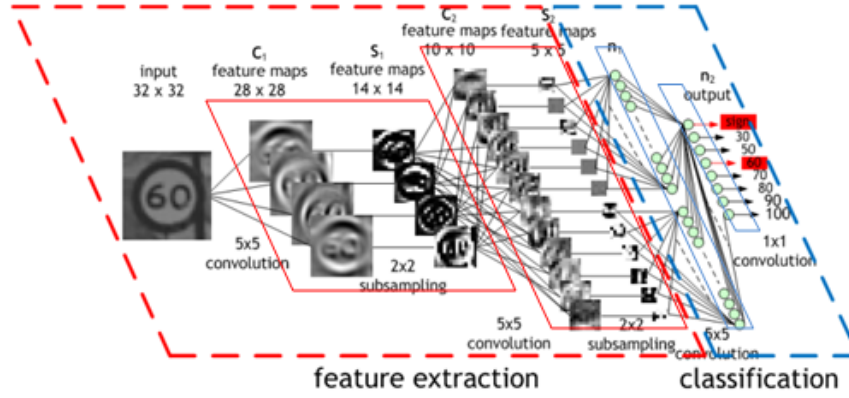


Figure 1.10: CNNs extract visual features by applying multiple convolutions in the first layers. Afterwards, relations of the resulting features get learned by fully connected layers for classification. [29]

segments etc. can be extracted and highlighted as demonstrated in figure 1.9.

In a convolutional neural network, image data matrices get processed through the layer structure, while multiple convolutions get applied to the data as activation functions. The weights of the filters get learned during the training process, so the network gets optimized to extract image features, that are relevant for the objective. Usually, early layers of a trained CNN perform data convolutions to first extract important visual features that get processed afterwards through fully connected layers to optimize the outputs (see figure 1.10).

Summary

Eye movements are not processed randomly but get highly influenced by subconscious cognitive processes which result from the ongoing, concentration demanding task. In visual search, fixations reveal the positions in the visual field, that disclosed the highest attraction and therefore are likely to contain features that relate to the search target. Gaining enough information about the target, may allow to infer the position or even the object itself. To reveal these information, the relevant visual features need to be extracted. This work analyzes the utility and useful involvement of image representations created by a convolutional neural network which has already learned how to extract relevant features but for a different purpose.

Analyzing data correlations and measuring the performances of prediction models, give insights, in which form CNN features are able to extract and represent target related features. Findings of this work can be applied to various fields like machine learning, computer vision but also human computer interaction. The ability of machines to "understand" what a user is looking for or interested in could improve intelligent user interfaces but also user supporting and interacting systems.

Chapter 2

Related Work

Search target inference with eye tracking data as well as the concept of extracting visual features with convolutional neural networks have already been considered before. The most relevant studies on which this thesis is aligned get introduced in this chapter.

2.1 Visual Search Target Inference

As mentioned in the introduction, Wolfe [24] showed that data of eye movements during a visual search on images is influenced by the target representation in the observer's mind. Zelinsky et al. [27] has proved, that this impact can be used to extract information about the target. In their work, the concept of "behavioral encoding" describes the ability of inferring human intentions out of observed conducts like gaze traces. With eye tracking hardware, several participants were instructed to find a specific object out of various distractors in a shown image. Only by considering the search trace, a human and also a machine learning model should predict the target object, assuming that the observer fixated target similar objects. The trained model considered extracted SIFT features [14] and local color histograms around fixated image regions and could preform above chance results. This revealed the general possibility of search target inference using gaze information.

The opportunity to decode gaze-target correlations is essential for this thesis. Unlike Zelinsky et al., this work considers the limits and benefits of CNN encodings as alternative to SIFT vectorization and color histograms.

The object consideration of Zelinsky et al. was restricted on a predefined set of data. Borji et al. [2] went a step further by considering unclassified local patterns during visual search on binary QR-code like images (see figure 2.1 a). Participants had to find a specific 3×3 block pattern in a generated search image. With an introduced similarity ranking algorithm called "pattern voting", structures in the search image, that provided a certain degree

of similarity to the fixated patterns, got selected as potential search target. Due to the simplicity of the binary image patterns, no additional feature extraction method besides the pattern voting similarity measure was needed. The resulting off-chance performance shows that fixated features, in this case pattern structures, relate to the search target and also are suitable for target inference.

This work adapts the idea of considering a similarity measure between fixations and search target. Therefore, the spatial inference concept gets introduced, which considers methods that infer target information from feature space structures. Aside from new approaches using CNN features, an adaptation of Borji's pattern voting algorithm gets introduced in section 5.2.1.

In the work of Sattar et al.[17], human search behavior got analyzed on aligned collages showing multiple images. The goal for participants was to find one specific image, while fixations during the search were recorded. Using RGB data for setting up a Bag of Words vectorization, fixation sequences got encoded and utilized to train a prediction model. Out of a limited set of target candidates, the resulting model stated which image was most likely the searched target for a considered fixation sequence.

Further, Sattar et al. introduced the concepts of "closed world" and "open world" for target inference. The settings distinguish from the data used for training the predicting models. Closed world means, that during the model training, samples with all possible labels get considered. For an open world setting, only samples with labels of a subset get used for training, while the performance gets tested on data with the unconsidered labels. This more challenging approach focuses on the general applicability of the model. The experiments of the closed world setting delivered promising results, while for the more difficult open world setting, models predicted significantly off-chance only for feature rich images.

In chapter 6, the implementation of Sattar et al.'s approach gets described in detail, as this work adapts the idea of using a Bag of Words vectorization for target inference. With a re-implementation, prediction performances of models using the RGB based Bag of Words encoding get directly compared to the novel approach processing CNN features. Parts of the publicly available collage dataset (see section 3.2) get used for these and other performance measures.

In a follow-up work, Sattar et al. [16] combined the idea of using gaze information and CNN-based features to infer the category of a user's search target in collages (see: figure 2.1) containing images of the DeepFashion dataset [13]. CNN activations of entire images combined with fixation density maps got applied to a global pooling operation which delivered encodings that got fed to a machine learning model. Finally, this model was able to categorize the search behavior to trained classes like "floral", "knit", etc.



Figure 2.1: (a) Binary pattern search image as used by Borji et al. in [2] with human gaze scan path seeking for the displayed 3x3 pattern. (b) Visual scan path conducted on a collage containing images of the fashion data set for the target attribute "floral" [16].

Using pre-trained neural networks as feature extractor to analyze gaze behavior, is generally also the concept of this work. To handle the large amount extracted of CNN features, some introduced methods for spatial inference also apply global average pooling. In contrast to [16], in this work, CNN features always get extracted around fixated spots in the search image. This allows to apply the methods on all kinds of images and does not require to manage visual objects individually.

2.2 CNN Features for the Representation of Visual Data

In [21], Sharif et al. consider image representations that result from hidden layers of a pre-trained CNN called OverFeat [19]. Using these for training machine learning models to perform different tasks like scene recognition and object detection delivered promising performances.

As the high information concentration in CNN features allows the application to different purposes, the intention of this work is to combine CNN features with gaze information and setup target inference models.

Using a similar network architecture, Donahue et al. [6] visualizes image data in the feature space of the hidden layer activations. The usually high dimensionality of image representations from CNNs get reduced to two dimensions. Plotting the images, using the reduced features as position measure, visualizes how the CNN arranges the data semantically. Images with similar content get placed closer together than samples showing semantically different objects. (see figure 2.2).

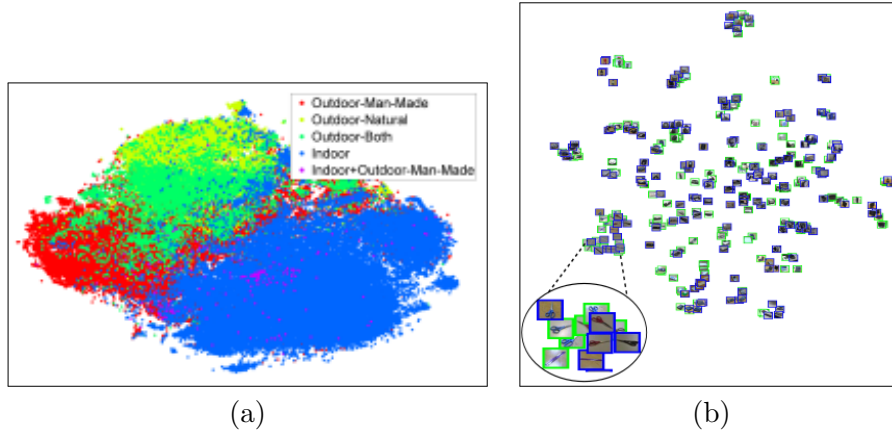


Figure 2.2: (a) Two dimensional visualization of image representations from the 6th hidden layer of the DeCaf neural network. Semantically similar images from the same class are located closely and setup clusters [6]. (b) CNN feature reduction as position coordinates. Original images from two different data domains (green and blue) have been processed through a CNN. The feature vector of the 6th hidden layer has been reduced to two dimension, which setup x and y coordinates of an image's position [6].

The clustering ability of layer activations for visualization purposes was also used by Jiang et al. [10]. For an interactive machine learning and fine tuning framework, several feature representations from hidden layers got concatenated and clustered. The purity of class occurrences in clusters indicated the ability of a prediction model to separate data correctly.

The cluster formations resulting from the CNN encoding process are essential for spatial target inference. Considering vector differences as similarity measure for visual features, could allow to derive feature analogies in fixations that might also appear in the searched target.

Summary

Several studies have shown that during a visual search, humans fixate locations with visual features that correlate to the searched target object. Different extraction methods, are able to encode these feature relations and allow an automatic search target inference.

The feature extraction ability of CNNs can be exploited for multiple purposes. The arrangement of image feature representations in the vector space respect semantically similarities of the data. Analyzing and using these spatial relations for similarity measuring and finally for target inference are the intension of chapter 5. Extending the idea of Sattar et al.[17] by introducing a CNN based Bag of Words approach gets covered in chapter 6.

Chapter 3

Datasets

The proposed methods in the subsequent chapters rely on eye tracking data of test participants conducting various visual search tasks. Therefore, Koehler et al.'s VIU dataset [11] and data samples of the already introduced work by Sattar et al.[17] got adapted to the needs of the implementations. A general description of the data as well as statistics and differences get introduced in the following.

3.1 VIU Dataset

For a collection of 800 images à 450×450 pixels, Koehler et al.[11] captured eye movement data from 100 test persons. The images, which display a variety of in- and outdoor photographs (see figure 3.1 a) got presented to the participants on a LCD monitor for two seconds. During that time, a tower-mounted Eyelink 1000 system recored the gaze behavior of the current test person.

To perform different investigations, the participants got distributed into three groups, each instructed to perform another task. Considering the images without any further instruction was the task for 22 test persons. In such free viewing situations, the gaze gets attracted by bottom-up features that reveal information about the saliency of images. The second group, consisting of 20 test persons, got instructed to allocate the object with the highest saliency in an image. Participants had to state, whether the most salient object, which in their eyes stood out for any reason, was on the left or on the right half of the image. Similar to the free viewing task, bottom-up features dominated the fixation attractions, but this time, the result gets more or less annotated.

The last task stated as "cued object search" was conducted by 38 participants and produced data which are relevant for this work. In each test iteration, a written word (see figure 3.1) b) got displayed for 1000ms. Within the subsequent 2000ms, in which the actual image was shown, the test person had



Figure 3.1: (a) Four sample images with natural scenes of the VIU dataset by Koehler et al.[11] with the gaze scan path of one user and the ground truth region (red rectangles). (b) Search target terms that were shown to participants for the cued object search.

to find the cued object. By pressing a button afterwards, the participant stated, whether he was able to find the target. The asked object was present in only 400 of the 800 images. Due to timing constraints each test person only observed 400 images while half of them contained the searched target. The later introduced models aim to predict the location of the target or the target as object. Samples not containing the seared object are not usable for this task. Therefore, the finally considered data consists of 400 images, each observed by 19 participants.

The fixed search time of 2000ms leads to a general problem, as it may happen, that the target gets found earlier. The fixations created afterwards are therefore not driven by the task anymore and consequently do not belong to the search sequence. Hence fixations that got captured inside the target area as well as fixations appearing after the target finding do not belong to the search process and therefore get ignored (see figure 3.2).

Averagely, each user considers a single image with 7.9 fixations, whereby 4.3 (maximally 15) belong to the search process which indicates that the search tasks are generally easy to accomplish.

For this work, each image got manually annotated with bounding boxes around the target objects which get denoted as ground truth. The probability of predicting the correct search target region by chance depends on the size of the ground truth area which differs for each image. Averagely, this area covers 17.43% of an image in the VIU dataset.



Figure 3.2: (a) Search sequence of fixations for the target "guitar". (b) Fixations from a) annotated depending on the stage of the visual search they appeared. For the search target inference models, only search fixations get used.

As explained in figure 1.7, the visual search conducted in these experiments is obviously cue driven. Reading the target describing word highlights an imaginary representation of the target object which guides the observer's gaze to related visual features (see figure 3.1 b). Further, images of the VIU dataset, displaying natural situations, provide semantic relations between the visible objects which also influence the search behavior like in figure 1.6.

3.2 Amazon Book Cover Dataset

In the inference approach already mentioned in 2, Sattar et al.[17] created Bag of Words encodings to train classification models to predict the targets. Therefore, visual search got conducted on collages that displayed multiple images aligned in a grid pattern with a small margin in between (See figure 3.3 a).

Fixation data from different images types have been collected during the experiments. 78 similar designed O'Reilly book covers, 84 book covers from Amazon with various colored illustrations and 78 greyscaled mugshots showing the faces of different persons setup three distinct collage datasets.

In each iteration, participants were instructed to find one specific image in the current shown and randomly generated collage, while their gaze got tracked with a stationary Tobii TX300 eye tracking device. Before the search, for a maximum of 10s the target image got displayed which should be found within maximally 20s in each of the subsequent shown collages. By pressing



Figure 3.3: (a) Example of one generated image collage showing various book covers. Observers were asked to find one specific cover. (b) The five target cover alternatives. The prediction models had to state which of these five covers got actually searched by a participant.

a key, the test person indicated that he found the search target. This work concentrates on the dataset containing the Amazon book covers, as it seems to provide a higher variety of visual features in each single image. 100 collages displaying 6×14 book covers got inspected by six test persons. Each collage got mapped to one of five target book covers (see figure 3.3 b) which was instructed to get found, resulting 20 collages for each target.

Because all book covers got displayed equally sized, the ground truth area of each collage takes the same proportion of 1.22%. Participants fixated a single collage averagely 14.47 (maximally 91) times. Due to the high amount of distractors, the large search space and missing semantical context between objects, this high amount of fixations can be explained with the difficulty of this search task.

In contrast to the VIU dataset, the search of this dataset is "image driven", as fixations are guided by the feature memorization of the shown target images. Therefore, the focused locations might reveal target features that are less biased by further impacts.

	VIU	Amazon Book Covers
images	400	100
observers	19	6
type	scene photographs	book cover collages
prediction chance	$\approx 17.43\%$	1.22%
avg fixations/image	4.3	14.47

Table 3.1: Direct comparison of the used datasets VIU by Koehler et al. [11] and Amazon book covers by Sattar et al. [17]

Summary

The two introduced datasets are suitable to analyze performance and applicability of the introduced methods. Covering different aspects of visual search, benefits and detriments can be revealed. Table 3.1 summarizes the most important characteristics of the datasets.

Chapter 4

CNN Feature Extraction

This chapter describes the procedure of visual feature extraction using a CNN, which gets studied in the subsequent experiments. The motivation behind this approach lies in the ability of neural networks to optimize the feature extraction process for a concrete task. The application of these extractions to other tasks, like introduced in chapter 2, may offers remarkable opportunities.

Defining the vector spaces \mathbb{I} containing image data and \mathbb{F} holding visual feature representation. A general feature extractor is a function $\Phi : \mathbb{I} \rightarrow \mathbb{F}$ transforming visual inputs to feature encodings. Studies mentioned in chapter 2 introduced multiple implementations of Φ . In this work, the method Φ_{CNN} extracts visual features by processing image data through a trained CNN and returning the activation of a selected network layer.

The CNN architecture used in this thesis was developed by Krizhevsky et al. [12] and is often called AlexNet. Input images get processed through five convolution- and three fully connected stages. The result of the network predicts the visible content of the input image (see figure 4.1).

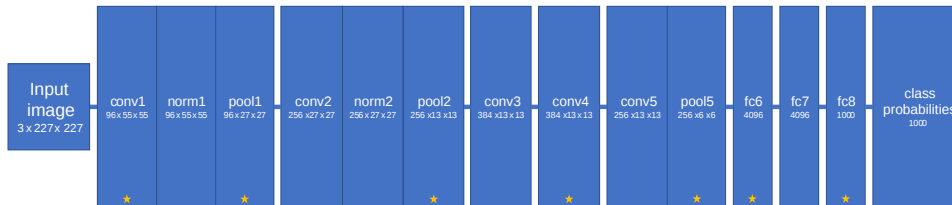


Figure 4.1: Simplified visualization of the AlexNet architecture. For each layer, channels × height × width state the shape of the image representations. Layers with a star get considered for feature extraction in this work.

AlexNet was adapted and trained on the popular ImageNet dataset containing about 14 million images. The output layer of the network holds probability values of 1000 selected object classes, describing how likely the corresponding object is visible in the input image. Deeper layers are higher biased by the classification task while early layers perform more general feature extractions (see figure 1.10). In the following experiments, features extracted from the layers conv1, pool1, pool2, conv4, pool5, fc6 and fc8, representing different stages of abstraction, get used and analyzed.

The general RGB encoding of images describes the color intensity of red, green and blue for each pixel in a range between 0 and 255. This three-channel layout gets reflected in the image data vector space, wherefore it holds that $\mathbb{I} = \mathbb{R}^{h \times w \times 3}$, with h and w as the height and width of images.

The convolutional layers of AlexNet produce multi-channel representations of the input image, which do not separate colors but various extracted features. Depending on the application, the resulting tensors, get either flattened to one dimensional array or processed by a so called global average pooling.

The one-dimensional flattening rearranges the feature components so that $\mathbb{F} = \mathbb{R}^n$ with $n = c \cdot w \cdot h$ while h and w state width and height of a channel window and c denotes the amount of produced channels. For most of the considered layers, the resulting feature space is extremely high dimensional. Feature vectors from the conv2 layer for example, consist of $256 \cdot 27 \cdot 27 = 186624$ components. Processing features for many fixation data would require enormous disc space and further alters the training time of prediction models.

Global average pooling (short: GAP), also used in [16] overcomes this problem by only considering the average value of each channel (see figure 4.2). Conv2 representation then only contain 256 components. To avoid misunderstandings, encodings of this type will be denoted with Φ_{CNN}^{GAP} .

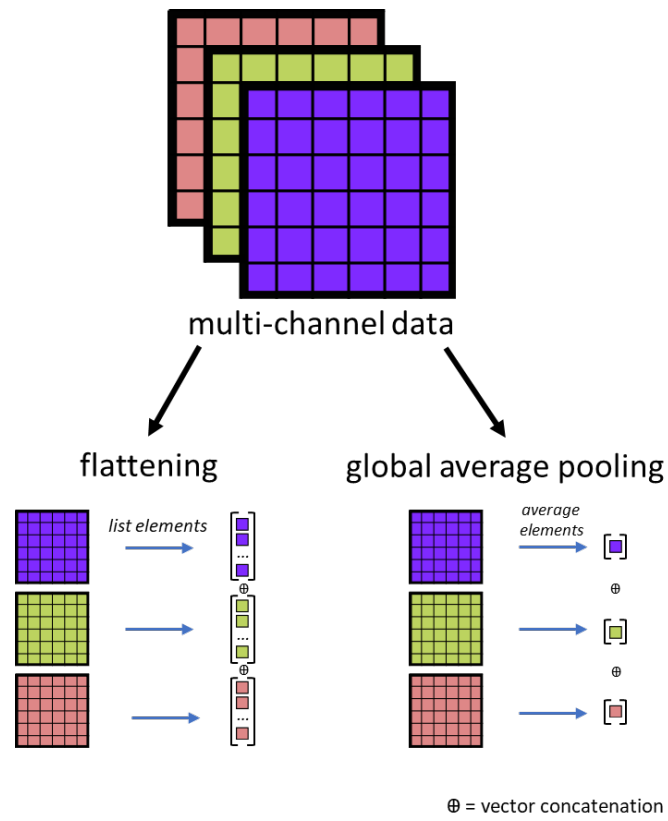


Figure 4.2: Differences between flattening and global average pooling for handling multi channel outputs from hidden CNN layers.

Chapter 5

Feature Space Analysis & Spatial Target Inference

This chapter describes the concept of spatial inference and analyzes the vector spaces of different CNN features. Spatial relations between fixations and target as well as methods to encode entire search sequences, so called aggregations, get presented and studied. Finally, the findings will be applied to two approaches that use vector distances as similarity measurements to predict targets from search fixations.

5.1 Spatial Feature Relations

During visual search, gaze attracting visual features are likely to be related to features defining the target. So called fixation patches, which are squared rectangles with a fixation in the center, get disjointed from the search images like in figure 5.1 and processed through a feature extractor Φ_{CNN} . The arrangements of the resulting encodings get analyzed in their vector spaces \mathbb{F} to study relations among fixations, between single fixations and their target and finally between entire fixation sequences and the target.

5.1.1 Fixation Cluster Analysis

Two vectors located close to each other in their vector space are likely to provide certain similarities. Data samples appearing in a concentrated structure hold low distances to each other and form so called clusters. Cluster distributions allow predictive models to classify data samples based on their feature space position. Data concentrations need to provide a certain degree of density to be significantly distinguishable from samples of random distributions. (see figure 5.2).

In order to analyze how search fixation features relate to each other, the idea is to measure the cluster densities of fixation patch encodings. Assum-



Figure 5.1: Creation of fixation patches by cropping out squared boxes from the search image around each fixation point.

ing that specific visual features attract the participant's gaze during search, fixations should provide particular similarities. Depending on the encoding, these similarities get reflected in the density measurement $\rho \in \mathbb{R}$ of the clusters.

As control scale, fixations at randomly generated locations in the search images, get further taken into account. Because the generated "fake" fixations do not rely on any attraction pattern, the resulting clusters should provide a less dense structure than concentrations consisting of human fixations.

Experiment

For each search image I , squared patches $p_i^I \in \mathbb{I}$ with a fixed size (VIU: 45px, Amazon Book Covers: 80px) get cropped around each fixation. Using different network layers as extractor, visual features of these fixation patches get encoded by $\Phi_{CNN}(p_i^I)$. Afterwards, the pairwise vector distance between all fixation encodings from the same image I get averaged. As density measure, the expected averaged distances using the euclidean as well as the cosine distance, get computed and compared:

$$\rho = \mathbb{E}_I[\mathbb{E}_{p_i^I \neq p_j^I}(\text{dist}(\Phi_{CNN}(p_i^I), \Phi_{CNN}(p_j^I)))] \quad (5.1)$$

Assuming that similar feature vectors provide low variances, the average component-wise variance of fixation patch encodings gets considered as third density measure. Therefore, for all encodings related to the same image, the variances of all vector components c get computed separately. The values of the resulting variance vector get averaged to a scalar. The expected scalar

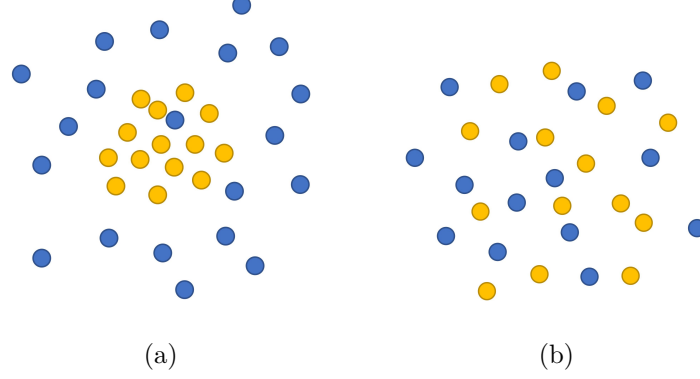


Figure 5.2: Assuming that yellow points are considered data while blue points represent noise in the feature space. (a) Dense concentrations of data differ from random noise and therefore contain characteristic features. (b) If the data does not form any concentration, its distribution is hard distinguishable from noise.

value from all images form the final component-wise variance density measure.

The relative differences between the cluster densities ρ_h of human fixation features and ρ_g for generated fixation clusters get indicated by the ratio $\frac{\rho_g - \rho_h}{\rho_h}$. When human fixation clusters provide a high density in contrast to generated fixations, this values is high. For a ratio of low magnitude, human and random fixations are almost equally distributed like in figure 5.2 (b).

Results

Figure 5.3 summarizes the density ratios for all dataset, extraction layer and density measure approach combinations.

For the VIU dataset, the pairwise cosine distance delivered for all network features positive ratios. Except for layer fc8, the euclidean and the component variance quantify human fixation clusters almost equally or even less dense than faked fixation representation clusters. The highest human fixation data concentrations got achieved by layer pool1 with a 22.6% ratio and by layer fc8 with a 33.5% ratio, both using cosine distance. For fc8, the euclidean distance with 13% and and the component variance with 22.5% delivered the highest non-cosine measurements.

The density comparisons for the Amazon book cover dataset generally are lower. All euclidean distance and component variance measurements indi-

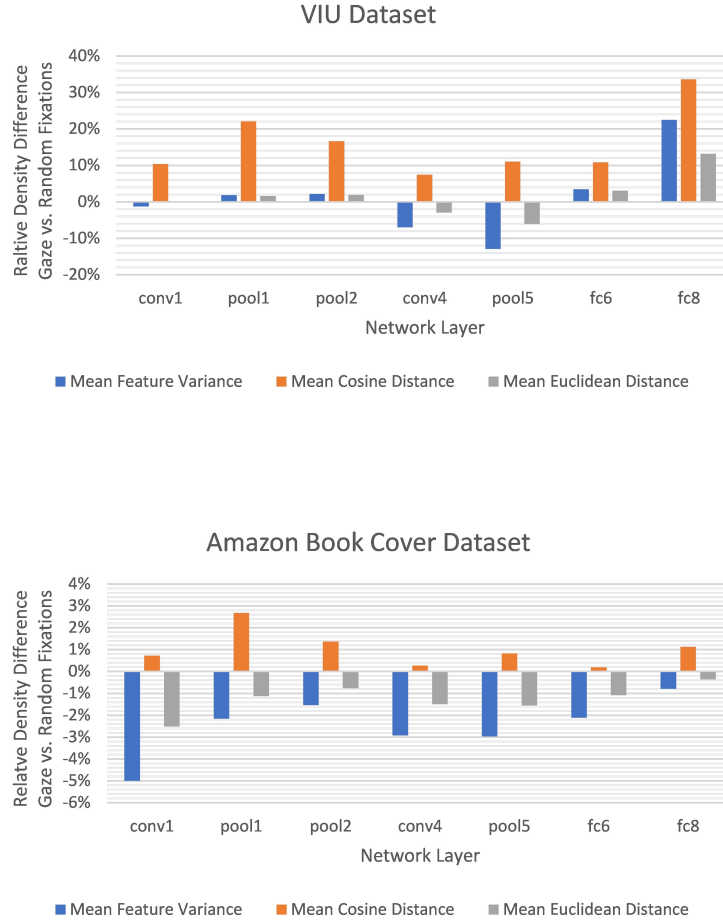


Figure 5.3: Relative differences of density measurements between human and random generated fixation encodings using different layers as feature extractor for the VIU and the Amazon book cover dataset. The values state how much lower (denser) the human fixation encodings are in contrast to clusters of randomly generated fixations. Positive values state a higher density of the gaze cluster.

cate a higher density for randomly generated fixations. Conv1 with euclidean distance reaches the lowest ratio of -5%. Of all positive cosine distance measurements, pool1 achieves with 2.7% the most dense human fixation representations.

Interpretation

The general lower ratio values for the Amazon book cover dataset are likely to result from the difficulty of the search task. The large image collages might need to be explored with less target related features. Further, VIU images generally provide less variety, which alters the probability of random generated fixations to lie on target similar features.

The euclidean distance and the component variance measurements delivered similar weak results. The high dimensionality of the feature representations lead to high absolute distances in the vector space. Considering vector angles with the cosine distance instead seems to be a more appropriate similarity measure.

The cosine distance approves the hypothesis that CNN encodings of gaze fixations provide certain characteristics which get reflected in the density of their spatial arrangement. This makes them distinguishable from random distributions and underlines the existence of a mutual relation. Despite the layers pool1, pool2 and fc8 seem to provide the most promising feature extractions, results of the other layers will still get taken into consideration for the following experiments.

5.1.2 Direct Relation between Fixations & Target

The previous section revealed, that CNN encodings of search target fixations provide certain spatial similarities to each other which can be measured by the cosine distance. This section analyzes dependencies between extracted features around single fixations and at the corresponding target object.

The comparison of human and randomly generated fixations discloses the actual gain of information by the CNN feature encodings. Assuming that target similar features attracted the participant's gaze during search, faked fixations should provide generally higher distances to the target encodings than recorded ones.

Experiment

Visual features at the target object get extracted, by computing $\Phi_{CNN}(t_I)$, with $t_I \in \mathbb{I}$ as the disjoint ground truth area of the search image I . For each image I , the feature similarity between an fixation patch $p_i^{(I)}$ and its target gets computed by:

$$dist(\Phi_{CNN}(t_I), \Phi_{CNN}(p_i^{(I)})) \quad (5.2)$$

Considering different layers as feature extractor, (5.2) gets applied to to all recored search fixations as well as to randomly generated fixations on

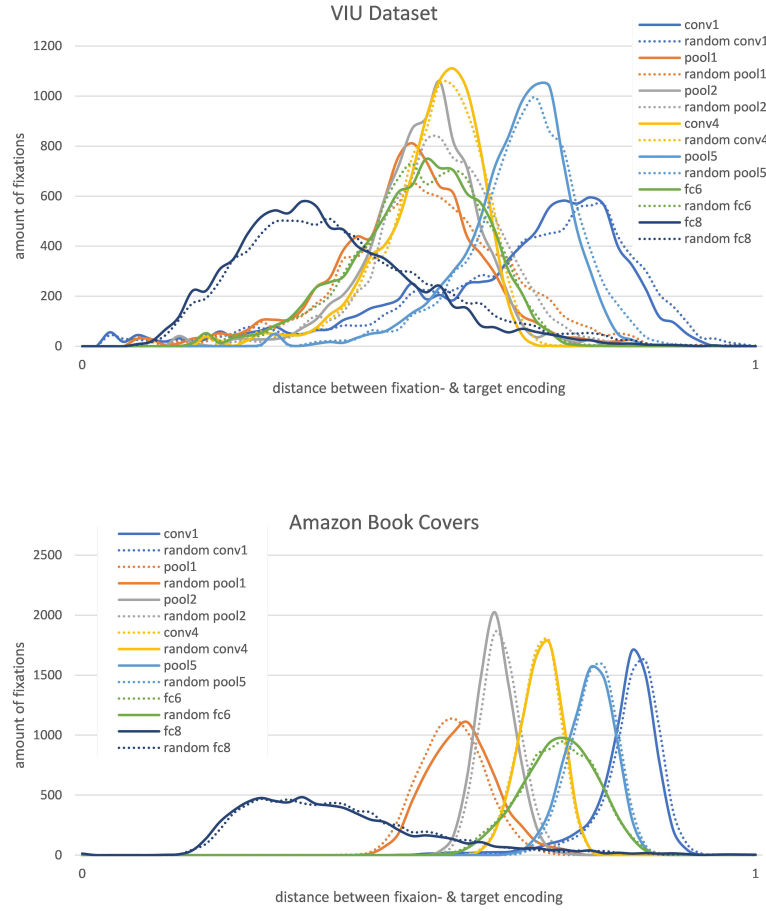


Figure 5.4: Histogram visualizations of distances between single fixation encodings and the encoding of their corresponding targets for the VIU and the Amazon book cover dataset. Solid lines show the data resulting from human fixations and dashed lines represent randomly generated fixations.

each search image. The squared fixation patches have fixed sizes of 45px for VIU images and 80px for Amazon book cover collages.

Results

Dependent on the extraction layer, different shaped normal distributions of fixation-target distances result as stated in figure 5.4. The distance distributions of human and randomly generated fixations are almost not distinguishable, regardless of the dataset and network layer. For VIU images as

well as for the book cover collages, layer fc8 provides generally the lowest target distances for captured and faked fixations.

Interpretation

The various distance distributions with different mean and variance values, dependent on the feature extractor, result from the different layer-specific output dimensionalities. Among all considered layers, fc8 extracts features with the lowest amount of components (1000) and therefore provides a higher probability of lower distances between encodings.

The almost identical distance distributions of human fixations and the not target-related, generated fixations indicate that CNN encodings, regardless on the considered layer, do not represent target similar features from search fixations well. Consequently, the spatial similarity consideration of a single fixation is not sufficient to infer useful information about the search target.

5.1.3 Temporal Relation between Fixations & Target

Besides considering visual features, search fixations can be further categorized depending on their appearance time in the search sequence. To get an overview of the image, more general and less target related locations might get fixated in the beginning of the search, which could even be bottom-up driven. Assuming that the gaze behavior adapts over time, the feature similarity between fixations and the target would increase with the search progress.

Section 5.1.2 showed, that the equally consideration of single fixations does not lead to useful fixation feature representations. To analyze the timing impact in visual search sequences, this section computes the average fixation target distances with respect to the search stage in which fixations got recorded.

Experiment

For each fixation, the index of occurrence $o \in \mathbb{N}$ gets determined and mapped in a reversed order. For the last fixation, before the user found the target object, it holds $o = 1$. Dependent on the sequence length, fixations captured at the beginning of the search provide higher occurrence indices. This notation allows to consider different sized fixation paths.

The target feature similarity gets computed by the distance 5.2 with squared patches (VIU: 45px, Amazon Book Covers: 80px) of each single fixation like in section 5.1.2. Afterwards, the measured distances of fixations with the same index o get averaged.

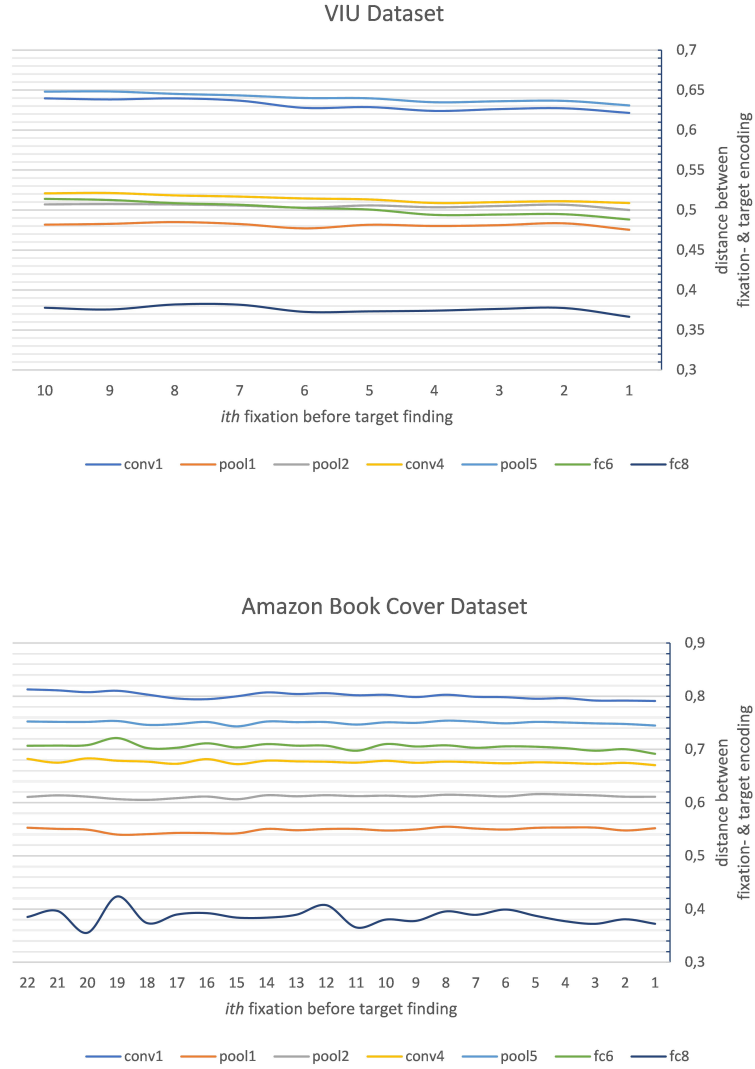


Figure 5.5: Distances of fixation encodings to their target dependent on the moment of appearance during the search for the VIU and the Amazon book cover dataset.

Results

Similar to 5.1.2, different feature extraction layers deliver generally different measuring value ranges but with similar behavioral patterns. The distance progress for fixations of both datasets get summarized in figure 5.5.

For the VIU dataset, target feature cosine distances continuously decrease for a about 1% over the fixations of the search sequence. The gaze data of the Amazon book cover does not show any remarkable development of the distances over the search time but generally provides more fluctuations.

Interpretation

The presented results show that there is only a very light decrease in distance the closer a fixation was recorded to the moment before finding the target object. Nevertheless, this development of maximally 1% is very slight. It seems, that the type of gaze attracting features does not change over the search sequence and the top-down processing remains almost constant from the beginning. Therefore, CNN encodings of fixations captured in different stages of the search can be weighted with equal importance.

Fluctuations of the distance measures for the Amazon Book Cover dataset, can be explained by the variety of search fixation sequence lengths.

5.2 Spatial Inference

The previous experiments showed that CNN features of single fixations do not provide spatial relations to features around the target. The existence of a certain spatial relation between multiple fixation encodings got indicated in section 5.1.1. Concluding from the timing considerations in section 5.1.3, all fixation encodings of a search sequence provide a similar relevance regarding the target relation. To respect all visual features fixated during a search sequence, this section introduces three aggregation methods that encode entire search sequences to single representations in \mathbb{F} using CNN features. After analyzing the distance relations between feature aggregations and the search target, the sequence encodings get used for two different spatial target inference approaches.

5.2.1 Fixation Aggregation Methods

In general visual search, after fixating a certain amount of gaze attracting spots in the search image, the participant finds the target object. The features around a single fixation which get extracted with the function $\Phi_{CNN} : \mathbb{I} \rightarrow \mathbb{F}$ are not sufficient to infer a similarity relation to target features in the vector space.

The idea of feature aggregation is to combine multiple fixation encodings with a function $\tilde{\Phi}_n : \mathbb{I}^n \rightarrow \mathbb{F}$ in order to respect all fixated features of a search sequence. The output space \mathbb{F} of $\tilde{\Phi}_n$ and $\tilde{\Phi}$ are equivalent which allows to analyze spatial relations between feature aggregations of fixation sequences and encodings of the search targets.

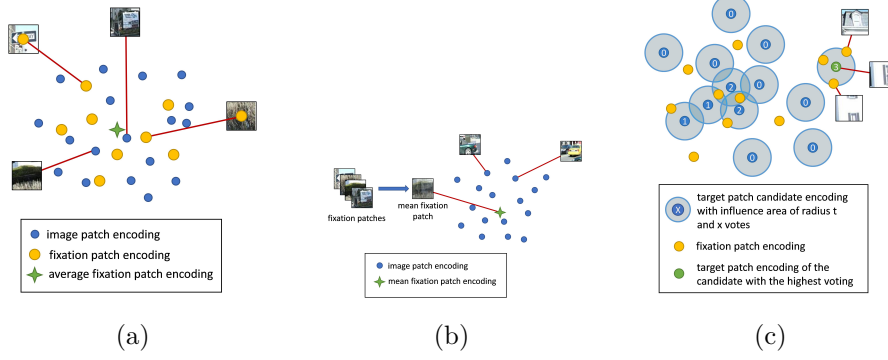


Figure 5.6: The three different aggregation methods visualized in a simplified feature space. (a) Representation Mean: The mean cluster center of the fixation patch encodings defines the aggregation (= green star). (b) Image Mean: First in image space the average of all fixation patches gets computed. Afterwards, its representation in feature space defines the aggregation. (c) An adapted version of Borji's pattern voting [1] counts fixation encodings around target alternatives. The candidate with the highest voting, represents the aggregation.

In the following, the aggregation methods "representation mean", "image mean" and "adapted pattern voting" which process CNN encodings of visual input data with different strategies, get introduced. Further, specifying a benchmark method "mean RGB histogram" which renounces the use of CNN encodings, allows to investigate benefits of CNN feature aggregations against classical approaches. All methods are based on extracting features from fixation patches as described in chapter 4.

Representation Mean

For each fixation patch $g_i \in \mathbb{I}$ resulting from a visual search trial, CNN features get extracted and simply averaged component-wise to a single encoding:

$$\tilde{\Phi}_n(g_1, \dots, g_n) = \frac{1}{n} \cdot \sum_{i=1}^n \Phi_{CNN}(g_i) \quad (5.3)$$

Spatially, the resulting feature aggregation vector is located in the center of the cluster consisting of the fixation feature encodings as visualized in figure 5.6 (a). Due to the mean computation, visual features that attracted the gaze more often have a higher impact on the aggregation. Fixation sequences with many differing features might lead to blurred and distorted representations that do not relate to the image space any more.

Image Mean

First, the mean RGB values for each pixel of all fixation patches get computed in image space. Extracting the visual features of the resulting patch with Φ_{CNN} delivers the image mean feature aggregation.

$$\tilde{\Phi}_n(g_1, \dots, g_n) = \Phi_{CNN}\left(\frac{1}{n} \cdot \sum_{i=1}^n g_i\right)$$

The averaging step in image space combines visual features differently. In contrast to the representation mean aggregation, the locality of the vector resulting from the image mean method is not known (see figure 5.6 b)

Adapted Pattern Voting

As already mentioned in chapter 2 Borji et al. [1] presented a voting based algorithm for target inference on simple image patterns. Adapting this approach to visual features in \mathbb{F} does not provide a direct aggregation representation, but rather a discrete similarity measure between fixation sequences and single elements from a limited set of alternatives $C \subsetneq \mathbb{I}$.

With Φ_{CNN} , feature encodings of each fixation patch g_i and of the preliminary defined target alternatives $c_s \in C$ get computed first. For each fixation, the target candidates whose encoding have a distance to the fixation encoding lower than a predefined threshold t , receive a +1-vote. After distributing votes for all n fixations of a search sequence, the candidate with the majority of votes $c^* \in C$ intends to contain the highest amount of fixation similar features among the alternatives. As this candidate can be seen as visual representation of the sequence, the final resulting feature vector is the CNN encoding of c^* :

$$v_t(x) = \begin{cases} 1, & \text{if } x \leq t \\ 0, & \text{else} \end{cases} \quad (5.4)$$

$$c^* = \arg \min_{c_s \in C} \sum_{i=1}^n v_t(\text{dist}(\Phi_{CNN}(g_i), \Phi_{CNN}(c_s))) \quad (5.5)$$

$$\tilde{\Phi}_n^C(g_1, \dots, g_n) = \Phi_{CNN}(c^*) \quad (5.6)$$

Visualized in figure 5.6, this approach counts fixation samples lying inside the high dimensional balls with radius t around the candidate encodings. The encoding with the highest count finally represents the resulting value. Due to the restriction on elements from C , the adapted pattern voting aggregation provides encodings that are comprehensible in image as well as in feature space.

Mean RGB Histogram

In [2], Borji et al. introduced search target inference on binary patterns (see chapter 2) as well as for virtual generated scenes. For these, fixation sequences got vectorized with concatenated red, green and blue color histograms of pixels in fixation patches. In the following, this approach gets considered as benchmark aggregation method not using CNN features.

Similar to the representation mean aggregation, fixation patch vectors simply get averaged but instead of using CNN encodings from Φ_{CNN} the $3 \times 256 = 768$ - dimensional RGB color histograms of the fixation patches get used as feature representations:

$$\tilde{\Phi}_n(g_1, \dots, g_n) = \frac{1}{n} \cdot \sum_{i=1}^n \Phi_{hist}(g_i)$$

5.2.2 Aggregation Target Relation

With distance measurements between fixation- and target representations, section 5.1.2 indicated that CNN feature encodings of single fixations do not reveal any significant benefit.

In this section, this consideration gets repeated by analyzing the distances between aggregation encodings of search fixation sequences and the feature encoding of the target. Assuming that fixation aggregations encode target related features of the whole search sequence, the measured distance distributions of human fixations should be distinguishable from distributions resulting from random generated data.

Experiment

Initially, squared fixation patches around each fixation g_i get cropped out with a fixed size (VIU: 45px, Amazon Book Covers: 80px). Afterwards, the distances between the feature aggregation of each search sequence $S = g_1, \dots, g_n$ and the simple CNN encoding of the corresponding target $t_S \in \mathbb{I}$ get computed:

$$dist(\Phi_{CNN}(t_S), \tilde{\Phi}_n(S)) \tag{5.7}$$

With different feature extraction layers for the introduced aggregation methods and for the target encoding process, (5.7) gets applied to all recorded search sequences and further to randomly generated fixations.

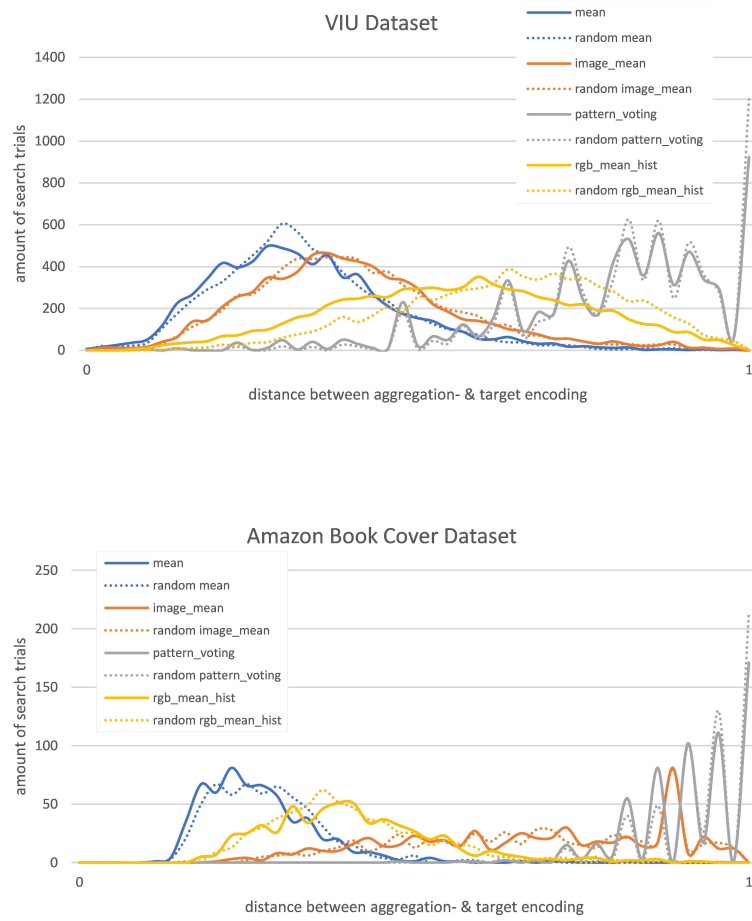


Figure 5.7: Distances of fixation aggregations to the respective target encoding for the VIU and the Amazon book cover dataset using layer fc8 as feature extractor.

Results

Generally, the differences between the distance distributions of human fixations and randomly generated fixations are very small, regardless of the extraction layer and the aggregation method. The distributions themselves differ in mean and variance, depending on layer and aggregation methods for both datasets similar to section 5.1.2. Representative for all applied layers, figure 5.7 displays the average distances between aggregations and their target encodings using layer fc8.

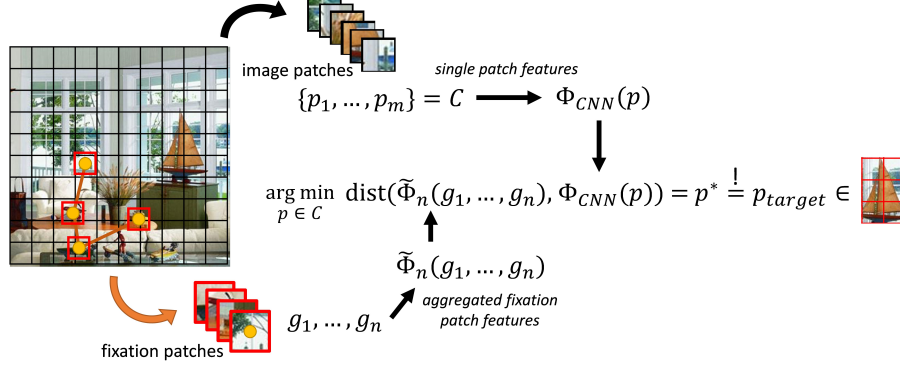


Figure 5.8: General pipeline of the spatial target inference algorithm. Feature encodings of all patches p_i in the image get compared to the fixation sequence aggregation. The most similar patch p^* is selected as target prediction.

Interpretation

Against the hypothesis, aggregations of human fixations which should contain target relating characteristics provide the same spatial similarity dependence as generated and unrelated fixations. Consequently, CNN feature aggregations do not generate fixation sequence representations that resemble the target features in the corresponding vector space.

Either, in all presented aggregation encodings, spatial relevant visual features get distorted or, CNN feature encodings of fixation sequences generally do not provide any direct spatial relation to features around the target object.

5.2.3 Search Target Inference by Vector Similarity

Section 5.2.2 concluded that search fixation aggregations do not relate directly to target feature encodings when using CNN features. Nevertheless, this section introduces and analyzes the spatial search target inference approach which uses the cosine vector distance as feature similarity measure between aggregation and target encoding. Goal of this approach is to locate the region in the search image which provides the highest of visual feature similarity to gaze attracting image spots.

Assuming that the target object is present in a search image $I \in \mathbb{I}$, target specific visual features $f_{target} \in \mathbb{F}$ are located in the respective ground truth area. Let $\mathbb{IP} \subsetneq \mathbb{I}$ be the set containing all image patches that can be gained by cropping I . For a search sequence of n fixations, the fixation

patches $g_i \in \mathbb{P}, 0 \leq i < n$ are intended to contain features similar to these of a not fixated patch $p_{target} \in \mathbb{P}$ showing the target object.

Considering the cosine distance of two feature encodings as similarity measure, the the feature aggregation resulting from $\tilde{\Phi}_n(g_1, \dots, g_n)$ might be located closely to the target patch feature encoding $\Phi_{CNN}(p_{target})$ so that:

$$\tilde{\Phi}_n(g_1, \dots, g_n) \approx \Phi(p_{target}) = f_{target}$$

As \mathbb{P} is an infinite set containing all possible image croppings of I , finding an appropriate p_{target} would be infeasible. Therefore, \mathbb{P} gets reduced to a subset $C \subsetneq \mathbb{P}$ containing image patches that result from subdividing the considered search image into non overlapping squared patches of fixed size. CNN encodings for all target candidate patches $p \in C$ get computed and compared to the fixation sequence aggregation of the search trial by measuring the cosine distance between the encodings. The patch $p^* \in C$ whose feature representation $\Phi(p^*)$ is closest to the aggregated sequence encoding is finally selected as target:

$$p^* = \arg \min_{p \in C} \left(dist(\tilde{\Phi}_n(g_1, \dots, g_n), \Phi(p)) \right) \quad (5.8)$$

Considering the distance measurements of all candidate patches, yields a discrete mapping function over the whole search image describing the feature similarity between each patch and the fixation sequence features.

The presented spatial inference approach, summarized in figure 5.8, does not require any learning process as it is based on the assumption that fixation-target feature similarities can be evaluated by distance in vector space.

Interactive Visualization Tool

In the context of this thesis, an interactive visualization tool has been developed which allows the user to understand the impacts of fixations to the patch prediction process.

By mouse clicks, gaze fixations and search sequences can be simulated on any selected search image. Choosing between different neural networks, extraction layers, patch sizes and aggregation methods provides the possibility to test and compare different prediction approaches. Figure 5.9 shows a screenshot of the application. The tool visualizes the image patch encoding similarities to the aggregation encoding of the user generated fixations by displaying a heatmap (see figure 5.10), which gets updated after each new added fixation. Green colored regions indicate a high similarity to the fixation sequence aggregation while red does the opposite.

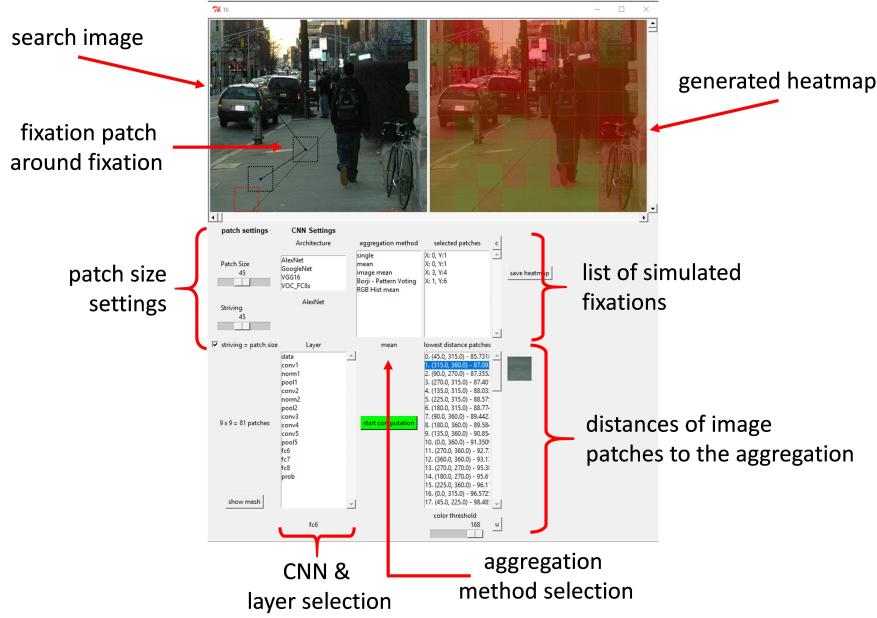


Figure 5.9: Annotated screenshot of the interaction tool. Users first chose an image, a CNN, feature extraction layer, aggregation method and set a wished size for fixation and image patches. Afterwards, by clicking on the search image, fixations can be simulated while the spatial inference gets conducted and visualized in real time as similarity heatmap.

Various resulting heatmaps approve the functionality of the low distance similarity concept as green marked regions often resemble fixated image spots (see figure 5.10). However, considering images from the annotated datasets, in the most cases, the target patches do not provide high feature similarities to fixated points which correlates with the findings of section 5.2.2.



Figure 5.10: Feature similarity heatmaps with corresponding fixation patches in the second row. Green regions indicate a high feature similarity between the image patches and the aggregation encoding of the fixation features.

Nevertheless, the visualization tool offers insights into the inference idea and allows a UI supported and responsive parameter adaption. Further, practical investigations can be easily conducted without the need of eye tracking hardware or dataset adaptations. For example tasks like, finding fixations that would lead to a correct prediction or testing the limits of certain approaches, benefit from the interactive real-time interface.

Spatial Search Target Inference Experiment

For different aggregation methods and extraction layers, the introduced spatial target inference approach gets conducted on the VIU and the Amazon book cover dataset. The image patch with the encoding providing the lowest distance to the fixation aggregation, gets selected as prediction. If the predicted patch overlaps the respective ground truth of the search image, the result is seen as correct.

To indicate the actual performances, all model accuracies get compared against two random measures. First, the statistical chance of predicting a patch in the ground truth has to be respected. When subdividing a search image into n patches, the statistical probability of selecting a target patch is $t \cdot \frac{1}{n}$ with the amount t of patches that overlap the ground truth area. This chance level depends on the selected patch size as well as on the average size of the ground truth area in all images.

Further, the prediction performance of randomly generated fixations contrasts the benefit of target relating features for the prediction process. Fixations, that should not relate to any target features might still lead to a



Figure 5.11: Target region prediction accuracies for the VIU dataset considering different patch sizes and CNN layers. Randomly generated and human fixations from the eye tracking get compared as well as the average chance probability.

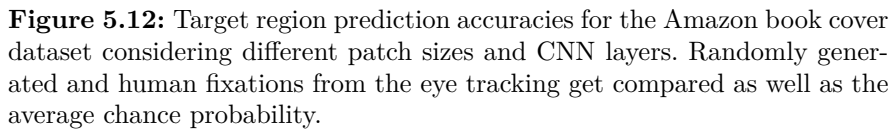
correct target prediction. However, performances in this cases do not result from a good prediction model but from an adverse visual feature distribution in the search image. Therefore this chance measure indicates the required quality of fixated features. High random fixation performances identify that many image spots provide target related features while low values indicate the need of fixating specific locations for a correct inference.

Results

Almost all prediction accuracies resulting from human fixations exceed the measurements of random generated fixations. Highest accuracies got always achieved for the largest considered patch size.

Considering the VIU dataset first, prediction results among different network layer extractors do just differ slightly. For the representation mean approach, all performances fall below the statistical chance level while the difference between prediction performance and random probability decreases with increasing patch size. The best prediction accuracy of 23% which is equivalent the the statistical chance probability got achieved with patch size 80 and pool5 as feature extractor.

In contrast, approaches using image mean aggregation generally lie above



All approaches using the pattern voting aggregation performed 3%-6% lower than chance. With 21% (chance: 23%) layer fc8 with patch size 80 delivered the best result for this category. With 29% accuracy (chance: 23%) the benchmark method using RGB histogram vectorizations and a patch size of 80px achieved the absolute best performance for the VIU dataset. Figure 5.11 summarizes all prediction results for images of the VIU dataset.

With an average performance of 4%, model variants using image mean aggre-

gation performed similar slightly above chance level, independent on patch size and layer choice. The highest accuracy of 6% got achieved by layer fc6 with path size 180px. Pattern voting based models, always achieved above chance level accuracies with the layers pool5, fc6 and fc8 for all patch sizes of maximally 5%. The RGB histogram based benchmark generally performed above statistical chance for human fixations but also for randomly generated samples. With almost 6% accuracy using a patch size of 180px, faked fixations even performed 1% better than human fixations.

Interpretation

In general, the prediction approaches using CNN feature aggregations do not deliver reliable results. The above chance performances of certain methods approve the concept of distance based inference but are still too vague for useful application.

Prediction results which do not exceed the statistical chance level indicate that feature similarity measures rather prevent target inference. In these cases, target features seem to differ and not resemble gaze attracting features.

The different accuracy ranges between the considered datasets result from the contrary difficulties of the search tasks. However, the target relation of book cover fixations gets better reflected in the encodings than for examples of the VIU dataset.

Another remarkable difference between the datasets is the performance variance considering different patch sizes. Performances of the VIU dataset increased with the respective patch size while for the Amazon book cover dataset, the patch size had a rather small impact on the prediction accuracies. This can be explained with the general structures of the images. Visual features in natural scenes like in the VIU dataset appear in larger segments while in image collages, features are restricted to always same sized areas. Therefore, extending the radius of considered features does not always benefit the prediction.

Summarized, certain approaches of direct spatial inference with CNN features predict slightly better than chance, but generally underly or do not outperform methods that do not use CNN features. Consequently, most fixation and target feature encodings do not directly correlate. Therefore, targets cannot get reliably inferred by just considering the locations of the fixation encodings in the vector space.

5.2.4 Target Inference by learned distance metric

The previous experiments as well as the heatmap visualizations clarified that CNN encodings of visual features at search fixations do not directly relate to target feature encodings. The vector distance might be a good feature similarity metric but the feature similarity itself does not seem to describe the relation between fixations and search target well. Therefore, this section introduces an approach which learns a distance metric between a fixation sequence and the target. The resulting model returns a probability value stating the likelihood that a certain patch contains the target of a given search fixation sequence. With fixation data of one test person for each dataset, models using different extraction layers, aggregation methods and patch sizes get trained and evaluated.

Experiment

For model training, the available data gets preprocessed and then adapted to a labeled training and test set.

A single data sample is set up by the combination of a fixation sequence aggregation with the feature encoding of a considered image patch. The corresponding label states whether the patch contains the search target of the search sequence or not. To keep dataset sizes and training time feasible, the global average pooling encoding of CNN features (see chapter 4) gets used in the following for all layers except fc6 and fc8.

Therefore, each search image I gets subdivided into n equally sized image patches $p_i \in \mathbb{P}, 0 \leq i < n$ like in section 5.2.3. Afterwards, the CNN feature encoding gets computed for each patch with Φ_{CNN}^{GAP} . The m fixation patches $g_j \in \mathbb{I}, 0 \leq j < m$ resulting from the fixation sequence of a single user get encoded with an aggregation method $\tilde{\Phi}_n^{GAP}$. For each image patch $p_i \in \mathbb{P}, 0 \leq i < n$, a single data sample gets created by concatenating the patch encoding $\Phi_{CNN}^{GAP}(p_i)$ with the fixation aggregation $\tilde{\Phi}_n^{GAP}(g_1, \dots, g_m)$. If p_i is overlapping the ground truth area of the search image I , the data sample receives the label 0, indicating a low distance between fixation features and target representation. If p_i does not contain the search target, the sample gets labeled with the maximum distance of 1. The dataset creation process gets summarized in algorithm 5.1.

Due to the naturally lower amount of image patches containing the search target, much more samples would receive the label 1 for not containing it which would bias the resulting model. To avoid this, the created datasets gets balanced by discarding randomly sampled data with label 1 until both labels occur equally often.

The model itself is a support vector machine (short: SVM). This architecture learns to reconstruct the label values out of the features of the

Algorithm 5.1: Generating a dataset out of fixation aggregations to learn a distance measure function.

```

1: CREATEDATASET( $C, F$ )
2:    $dataset \leftarrow \emptyset$ 
3:   for all image patch  $\in C$  do
4:      $x \leftarrow \Phi_{CNN}^{GAP}(image\ patch) \oplus \tilde{\Phi}_n(F)$ 
5:     if target  $\in$  image patch then
6:        $y \leftarrow 0$ 
7:     else
8:        $y \leftarrow 1$ 
9:     end if
10:     $dataset \leftarrow dataset + (x, y)$ 
11:  return  $dataset$ 
12: end for
13: end

```

corresponding data samples. In this case, it predicts for a data sample $x = \Phi_{CNN}^{GAP}(p) \oplus \tilde{\Phi}_n^{GAP}(F)$ whether the patch p contains the target which got searched in the fixation sequence F or not. Therefore, the chance probability of returning a correct prediction result is 50%.

For both datasets, models get trained and tested with fixations of one user. The performances of using different patch sizes, feature extraction layers and aggregation encodings get evaluated by a 10-fold cross validation.

Results

For both datasets, all prediction models performed significantly better than chance. Considering the results of VIU data samples, the performances of models using different parameters achieved similar accuracies of 52% - 64%. The best performance was delivered by layer fc6 using image mean aggregation on a patch size of 45px.

For models trained on the Amazon book cover dataset, the patch size seems to have a larger impact. The highest accuracy of 73% was achieved by combining layer pool2 with the representation mean aggregation as well as by using mean RGB-histogram aggregation without using CNN features, both with 80px patch size. Models using a patch size of 180px generally performed worse, with maximally 64% accuracy using layer pool5 and representation mean aggregation.

Figure 5.13 visualizes all measured prediction accuracies of the trained and evaluated models.



Figure 5.13: Prediction accuracies of distance learning models for VIU and Amazon book cover dataset for different patch sizes, extraction layers and aggregation methods.

Interpretation

Despite the book cover collages provide larger search images and therefore a more difficult search task than VIU images, models trained on fixations of the Amazon book cover dataset generally performed better.

One can say that encodings of fixated features of the Amazon book cover dataset are more related to the feature representations of their search target than encodings of fixations from the VIU dataset. This might result from the higher concentration needed for the visual search in image collages, which further do not rely on semantic context. Anyway, none of the presented methods delivered an outstanding performance but all prediction accuracies remained in a relatively small range. Further, models using CNN encodings did not perform much better than approaches using the RGB based benchmark. A model which did not use CNN encodings even achieved the best prediction results for the Amazon book cover dataset. This indicates that CNN feature encodings and aggregations might deliver target related representations but these encodings do not provide any real advantage over simpler color value vectorization methods.

Summary

For search target inference, visual features at locations fixated by the observing person relate to visual features of the search target. This chapter analyzed the idea, that this relation gets reflected in similarities of features which can be encoded by convolutional neural networks. Therefore a method got introduced which extracts activations from hidden layers of a trained CNN as visual feature representation.

Considering the vector space of the CNN feature encodings revealed that fixations of a visual search provide characteristics which makes them distinguishable from features without a search target relation.

Anyway, measuring feature similarity by computing the cosine distance of CNN encodings did neither reveal any temporal dependency nor any relation between fixation and target encodings. Also combining fixated features to an aggregation did not deliver any significant benefits.

Locating the region in the search image which provides the highest similarity to fixated points got introduced as similarity based target inference. Without the need of a learning process, this approach only performed around chance level.

Using a CNN feature based fixation-to-target metric, which got learned by a support vector machine delivered clearly above chance performances. Nevertheless, CNN encodings did not provide significant advantages over simple color based feature extraction methods.

In general, the introduced CNN feature encoding methods in combination with fixation data seem to provide too vague target revealing information. Fixation relations can be measured, but applicable models require more accuracy and stability.

Chapter 6

Bag of Neural Network Features

In the previous chapter, visual features got encoded by hidden CNN activations to infer search targets images. By the subdivision into image patches, spatial inference approaches analyzed all regions in the search image as potential targets.

Considering Sattar et al.'s distinction of open and closed world setting [17], spatial inference performs an open world inference as none of the models was specifically trained on a later predicted target. Because CNN features do not provide outstanding performances on the more challenging open world inference, this chapter analyzes by an adaption of the approach by Sattar et al.[17] the more restricted closed world target inference. Therefore, the general Bag of Words vectorization method, used in [17] gets modified to an approach using CNN features and introduced as Bag of Neural Network Features.

After explicitly describing the target inference approach by Sattar et al.[17] which uses an RGB based Bag of Words encoding, the implementation and evaluation of the adapted technique get considered.

To expand the applicability of the closed world approach to natural scenes, the use of semantic segmentation gets introduced and deployed in combination with Bag of Neural Network Features in the last section of this chapter. Further, the influence of respecting semantic information at search fixations for target inference gets measured.

6.1 Closed World Inference by Sattar et al.

A general Bag of Words (short: BoW) is a vectorization method which encodes data sequences to histogram representations. BoW encodings often get used for automatic text understanding and image classification.

To setup a BoW, a limited set of vectors (= code words) which represent remarkable and distinguishable features of the considered data, gets defined. Depending on the data, the code word generation can be performed by various methods. To encode a sequence of data, for each sample, the most similar codeword gets selected and noted in a histogram which counts the matchings for each codeword. The resulting histogram vector represents the final encoding for the sequence, which can be used for different purposes like model training.

In their work, Sattar et al.[17] used a BoW approach to encode fixation sequences of visual search trials on image collages, inter alia on images of the Amazon book cover dataset. Training a five-class SVM, fixation sequence encodings got used to predict the searched target image out of a set of five alternatives in a closed world setting.

Obviously, when finding the target image, the observer fixated it at least once. But these fixations in the ground truth do not belong to the search process anymore. Against this definition of a search fixation stated in figure 3.2, in [17] the last fixations of a sequence have always been taken into consideration even if they are located at the search target.

For this work, the approach of Sattar et al. got re-implemented for closed world inference. In the following experiments, the Bag of Words setup as well as the closed world target prediction gets described and evaluated with this re-implementation using the Amazon book cover dataset. To analyze the impact of the last fixations of a search sequence which are located in the target region, performances of models that in- and excluded these fixations, get compared.

Experiment

For the setup of a Bag of Words, which vectorizes search fixation sequences, the fixations on all search images in the training set get considered separately. First, the book cover in the image collage, which got focused (not necessarily fixated) by the observer gets determined for each single fixation. This gets achieved by choosing the book cover with the lowest Manhattan distance from the fixation point. When the fixation is located inside the bounds of a book cover, this step is trivial. Except the focused cover, all book covers in the search image get "hidden" by setting their RGB color

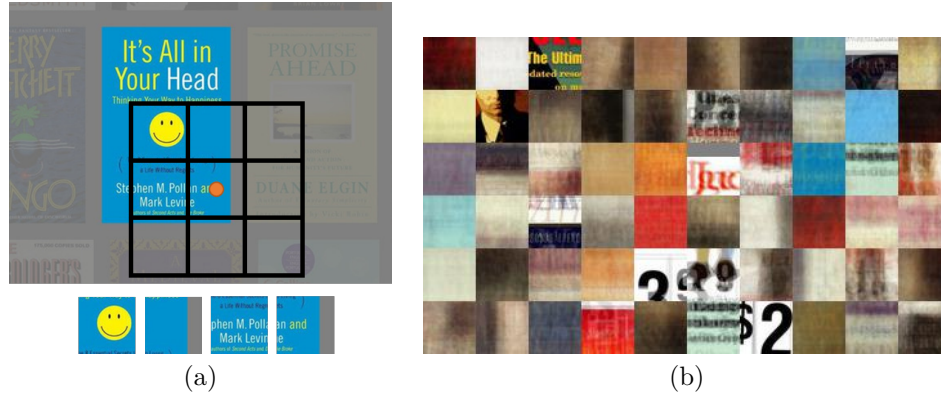


Figure 6.1: (a) Book cover patch selection. Except the selected cover, all other images get masked out. Totally nine patches get considered while patches that contain a certain amount of gray, get ignored. (b) Visualized code word vectors resulting from a k-means clustering ($k=60$) of fixation data from the Amazon book cover dataset. RGB colors get mixed from multiple fixation patches in the same cluster.

values to gray ($RGB = 128, 128, 128$) as visualized in figure 6.1 a.

To capture the visual features at the fixated location, the search image gets cropped to a squared image patch with the fixation point in the center. As eye tracking data are often slightly inaccurate, the eight directly surrounding fixations patches get extracted additionally as stated in figure 6.1 a. Patches that exceed the borders of the focused book cover, mainly consist of gray pixels and do not provide important information. These patches get sorted out, so that one single fixation maximally delivers nine patches.

For all available fixations these patches get collected and represented as flattened RGB vectors. With a preliminary defined code word amount k , a k-means clustering gets applied to all fixation patch vectors. The resulting k cluster centers form the code words (see figure 6.1 b) of the so called Bag of Visual Words.

For the actual fixation encoding, search sequences get represented as k sized histogram vectors. Therefore, each fixation of the considered sequence gets mapped to the most similar code word of the Bag of Words. The i th component of the resulting encoding vector states the amount of similarity mappings for the i th code word. The idea of this way of encoding is, that for some targets, participants fixate certain features more often than others. Counting, how often which feature got fixated, might provide enough information for a machine learning model to find out which target candidate was searched.

Finally, a five class support vector machine gets trained to predict the ID

of a target cover when getting the fixation sequence encoding as input. As different test persons might applied different strategies for the visual search, an individual Bag of Words as well as an individual prediction model get created for each participant (within-user consideration). The final model performance gets evaluated with fixation sequences of the test set which get encoded by the BoW generated in training.

The described process gets conducted on all available fixations as well as on fixation sequences ignoring the last fixations in the respective ground truth area.

Results

With this approach, Sattar et al.[17] as well as the re-implementation for this work achieved an average prediction accuracy over all users of 75% (chance: 20%) using a patch size of 41 pixels and a Bag of Visual Words of size $k = 60$.

The model trained with the same parameters but restricting on search fixations that do not lie in the ground truth area performed with an average accuracy of 36%.

Interpretation

The high performance decrease when ignoring the last fixations in the target area underlines the vast impact of these fixations on the prediction model. The above chance performance indicates that the color based Bag of visual Words encodings provide target correlations even when no target features got considered. Nevertheless, the resulting 36% prediction accuracy is too low for reliable real-time target inference.

6.2 Bag of Words Adaption using CNN Features

The method by Sattar et al. [17] described in the previous section encoded fixation sequences with a Bag of Visual Words approach which is based on the RGB data of fixation patches from the search image. As feature encodings introduced in chapter 4 might have the possibility to consider visual features which are not describable with simple RGB vectors, this section introduces the alternative approach, called Bag of Neural Network Features. The idea is to measure the prediction performance of the same target inference concept like in section 6.1, but replacing the consideration of RGB data from patches with the corresponding CNN feature extraction Φ_{CNN} .

Experiment

Leaving the patch extraction and filtering process from Sattar et al.'s approach untouched, for each fixation patch the encoding Φ_{CNN} gets computed and used for the k-means clustering to generate $k=60$ code word vectors.

For each fixation sequence, the code word histogram gets created with the similarity mapping of the respective CNN encodings of the fixation patches to the code words. Like, in the color based approach, the created histograms get used for training a five class SVM which predicts the target for search image collages of the Amazon book cover dataset.

For testing, unconsidered samples get encoded with the Bag of Neural Network Features generated in training. Prediction performances get measured for models with different layer extractors and that got trained on all available fixations as well as on sequences excluding finding fixations that do not belong the the search process.

Results

Taking all available fixations into account, layer fc6 as feature extractor with a Bag of Neural Network Features of size 60, performs best with an accuracy of 86%. Excluding finding fixations in the ground truth area, leads to a decrease of performance for all layer extractors. With an accuracy of 44%, layer fc8 performed best with only considering search fixations.

Figure 6.2 summarizes the results of the models using different layers and fixation sequences and further compares the performances against the RGB based approach from section 6.1.

Interpretation

Compared to the performance of the benchmark method by Sattar et al. which achieved under the same circumstances 10% less accuracy, fc6 features allow the SVM a better separation of classes than RGB values.

Like in section 6.1 also for CNN feature models, the last fixations in the ground truth provide many information about the target, as discarding them leads to a performance drop of averagely 50%. With 44% accuracy of a model inferring the target image without considering target fixations, CNN features still performed an improvement of 22% against the color based re-implementation of Sattar et al.

Summarized, one can say that Sattar et al.'s closed world target inference approach generally could be improved by the introduced Bag of Neural Network features method to encode search fixation sequences.

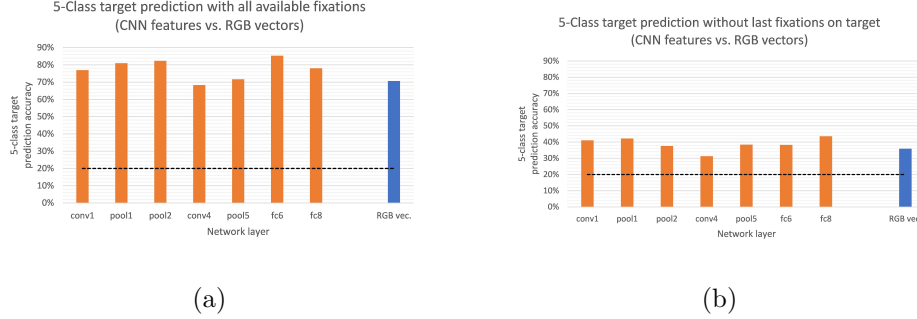


Figure 6.2: Accuracies of 5 class one-vs-all SVMs trained to predict the searched book cover by encoding fixation patches with a bag of neural network features. (a) displays the accuracies of a prediction model while all fixations that have been recorded during the experiment got used. The performances stated in (b) got achieved by models which do not took the last fixations into account, which are located inside the area of the target book cover, as these fixations do not belong to the search process anymore.

6.3 Semantic Supported Inference

In the previous section, the introduced Bag of Neural Network Features approach was used for a closed world search target inference task. The target candidate restriction to five alternatives allowed the prediction model to explicitly learn target specific feature correlations which led to a general good performance. Therefore, it was required that the same target got searched in multiple trials in order to provide a suitable training set.

For images of the VIU dataset showing natural scenes, this approach is not applicable as depending on the visible content, the target alternatives would have to differ for each image. Therefore, it does not make sense to restrict on certain object classes to train on. Nevertheless, one strong characteristic of natural scene images is the presence of a semantical organized structure, a so called context between visible objects as described in the beginning in figure 1.6.

In the following, a neural network called SegNet [23] gets used to extract additional context information from images of the VIU dataset. SegNet was trained on the SUN RGB-D dataset [22] managing 10000 images of indoor scenes to detect and classify visible objects in an image. Therefore, input images get segmented into multiple distinct areas of different semantic classes (see figure 6.3). The single class meanings of the resulting segments are not relevant for the following consideration unlike the segmentation and their distinction.

Instead of inferring the exact position of the search target, this approach



Figure 6.3: Semantic segmentation with SegNet. After processing the input image (a) through SegNet, each pixel gets mapped to a certain class (b). Segments with the same class are likely to provide similar semantic meanings.

aims to predict the segment class in which the target object might be located to reduce the search space. This concept allows to apply an adapted closed world target inference on any kind of natural images. Further, the semantical class around the search fixations, which might be related to the target segment class, gets included into the learning process. With this consideration, inter semantic dependencies like for example: "When a user looks for a car, streets get fixated often while buildings do not" might support the inference performance. To measure the actual benefit, the performances of models with and without semantical context consideration at the fixations get compared.

Experiment

First, each considered search image gets processed through the SegNet network resulting labeled segmentations. Afterwards, fixation patches of 45px get cropped for each image of the training set in order to set up a Bag of Neural Network Features of size 10 as introduced in section 6.2 which gets further used to encode all fixation sequences.

For the impact analysis of considering semantical context around fixations, the segment class which is dominant in a fixation patch gets mapped to the respective fixation in a sequence. Counting which class got fixated how often during a trial, creates a corresponding a histogram vector.

For the approach which ignores the fixation semantics, a single data sample consists of the Bag of Neural Network Features encoding. For the semantic respecting method, the Bag of Neural Network Features encodings get concatenated to respective semantic class histogram for each fixation sequence. In both cases, as prediction label, the class gets selected, which is dominant

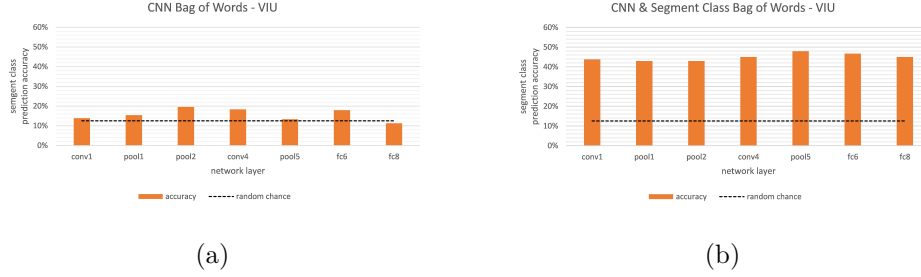


Figure 6.4: Accuracies of 8 class one-vs-all SVMs trained to predict the class of the segment containing the searched object. (a) shows the model performance using only a CNN feature bog of words encoding of fixation patches while (b) additionally considers the segment classes of fixations during search.

in the ground truth area.

For natural images it is obvious that several classes appear more often than others which would lead to an unbalanced dataset. Therefore, only the eight classes that contain the search target in most images get considered while images whose target is located in a segment of another class get ignored. To respect test person specific search behavior, for each participant an individual SVM gets trained. Using different layer extractors and both, either considering fixation semantics or not, allows to find the best feature representation and states the impact of semantic consideration around fixations.

Results

Figure 6.4 shows the prediction results of the described experiments. The prediction performances of models which do not consider the semantical context at fixated points provide an average accuracy of 15.6%. The best performance of 20% has been achieved with pool2 as extraction layer.

Considering the semantic class at the fixations, increases the the prediction accuracy for all layer extractors up to 48% using layer pool5. Predicting the target segment class out of eight alternatives provides a chance probability of 12.5%.

Interpretation

The low performance results of the approaches without the consideration of semantical context at fixations show, that none of the CNN feature extraction methods provides significant segment related information. This can result from the variety of visual features that might be located in segments

of the same class. Relations of these different domains seem to be too vague. However, the vast performance increase when respecting the segment classes at search fixations underlines the importance of semantic relations in target inference while visual features seem to play a minor role in this consideration.

Summary

In this chapter, the closed world target inference approach introduced by Sattar et al.[17] got considered and analyzed. The impact of fixations that appear at the end of a search sequence and that are located in the ground truth area of a search image got approved. As these fixations do not belong to the search process, the stated performances in [17] might be misleading.

By adapting the Bag of Words vectorization with CNN feature encodings, the target inference performance for the Amazon book cover dataset could generally be improved.

To apply the introduced approach to more variable natural images, a semantic segmentation model got involved to deliver trainable classes for the closed world inference which reduces the search space dependent on the search behavior. Further, the consideration of semantic classes at fixated regions delivered a vast performance boost to the prediction of the semantic segment in which the target is located.

Chapter 7

Conclusion

Convolutional neural networks generally deliver very precise image classification results because they are able to learn extracting relevant information from images to find existing correlations to considered object classes.

This work presented and evaluated multiple approaches to use extracted features from convolutional neural networks to perform a gaze based search target inference on image data. Visual feature encodings from the considered pre-trained CNN respect a higher amount of information than approaches from previous work and therefore are worth considering to be applied to the target inference concept.

By introducing the idea of spatial inference, relations between fixation encodings and target representations got analyzed in multiple experiments in order to find characteristic vector space structures that can be exploited for target inference. Under the assumption that fixations and search targets provide feature similarities which can be described by the encoding vector distance, neither a similarity based inference approach nor the explicit learning of a similarity measure delivered outperforming target inference predictions. This concludes that fixation target relations of visual search tasks underly more complex correlations which can just barely get represented with feature similarities.

The second consideration in this work described an adaption of the popular Bag of Words vectorization using CNN features. Applied to the closed world inference approach by Sattar et al. [17], the use of the so called Bag of Neural Network Features increased the inference performance by 20%.

Further, with the involvement of automatic segmentation, a way to conduct closed world inference on natural scenes got presented which does not predict the target object, but reduces the search space to support the finding process. Moreover, this approach revealed the high importance of respecting

the semantical context of fixated objects in search images.

Semantical dependencies in search tasks provide a high potential for future research to improve stability and precision of target inference systems. An application of this work's findings could be a usability consideration of real time search support. Live target predictions in combination with smart glasses might open various possibilities. Not restricting on search target inference, the achievements and approaches of this thesis can be adapted and applied to many further fields of supportive and gaze based human computer interaction.

References

Literature

- [1] Ali Borji. “Boosting bottom-up and top-down visual features for saliency estimation”. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2012), pp. 438–445 (cit. on pp. 7, 38, 39).
- [2] Ali Borji, Andreas Lennartz, and Marc Pomplun. “What do eyes reveal about the mind?. Algorithmic inference of search targets from fixations”. *Neurocomputing* 149.PB (2015), pp. 788–799. URL: <http://dx.doi.org/10.1016/j.neucom.2014.07.055> (cit. on pp. 1, 2, 16, 18, 40).
- [3] Donald Eric Broadbent. “Pereception and Communication” (1958) (cit. on p. 4).
- [4] Andreas Bulling et al. “Eye movement analysis for activity recognition”. *Proceedings of the 11th international conference on Ubiquitous computing* (2009), pp. 41–50. URL: <http://dx.doi.org/10.1145/1620545.1620552> (cit. on p. 5).
- [5] Stijn De Beugher et al. “Automatic analysis of eye-tracking data using object detection algorithms”. *UbiComp* November 2015 (2012), p. 677. URL: <http://dl.acm.org/citation.cfm?doid=2370216.2370363> (cit. on p. 2).
- [6] Jeff Donahue et al. “DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition”. *Icml* 32 (2014), pp. 647–655. arXiv: 1310.1531. URL: <http://arxiv.org/abs/1310.1531> (cit. on pp. 18, 19).
- [7] Jon Driver. “A selective review of selective attention research from the past century.” *British journal of psychology (London, England : 1953)* 92 Part 1 (2001), pp. 53–78. arXiv: arXiv:1011.1669v3. URL: <http://www.ncbi.nlm.nih.gov/pubmed/11802865> (cit. on p. 4).
- [8] R. L. Gregory. “Knowledge in perception and illusion”. *Philosophical Transactions of the Royal Society B: Biological Sciences* 352.1358

- (1997), pp. 1121–1127. URL: <http://rstb.royalsocietypublishing.org/cgi/doi/10.1098/rstb.1997.0095> (cit. on p. 6).
- [9] James E. Hoffman. “Visual attention and eye movements”. *Attention* (1998), pp. 119–153 (cit. on pp. 5, 6).
- [10] Biye Jiang and John Canny. “Interactive Machine Learning via a GPU-accelerated Toolkit”. *Proceedings of the 22nd International Conference on Intelligent User Interfaces - IUI '17* (2017), pp. 535–546. URL: <http://dl.acm.org/citation.cfm?doid=3025171.3025172> (cit. on p. 19).
- [11] K. Koehler et al. “What do saliency models predict?” *Journal of Vision* 14.3 (2014), pp. 14–14. URL: <http://jov.arvojournals.org/Article.aspx?doi=10.1167/14.3.14> (cit. on pp. 21, 22, 25).
- [12] Alex Krizhevsky, Ilya Sutskever, and Hinton Geoffrey E. “ImageNet Classification with Deep Convolutional Neural Networks”. *Advances in Neural Information Processing Systems 25 (NIPS2012)* (2012), pp. 1–9. arXiv: 1102.0183. URL: <https://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf> (cit. on p. 26).
- [13] Ziwei Liu et al. “DeepFashion: Powering Robust Clothes Recognition and Retrieval with Rich Annotations”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2016, pp. 1096–1104 (cit. on p. 17).
- [14] D.G. Lowe. “Object recognition from local scale-invariant features”. *Proceedings of the Seventh IEEE International Conference on Computer Vision* (1999), 1150–1157 vol.2. arXiv: 0112017 [cs]. URL: <http://ieeexplore.ieee.org/document/790410/> (cit. on p. 16).
- [15] John H. Reynolds, Jacqueline P. Gottlieb, and Sabine Kastner. “Chapter 46 - Attention”. In: *Fundamental Neuroscience (Fourth Edition)*. Ed. by Larry R. Squire et al. San Diego: Academic Press, 2013, pp. 989–1007. URL: <https://www.sciencedirect.com/science/article/pii/B9780123858702000469> (cit. on p. 1).
- [16] Hosnieh Sattar, Mario Fritz, and Andreas Bulling. “Visual Decoding of Targets During Visual Search From Human Eye Fixations” (2017), pp. 1–9. arXiv: 1706.05993. URL: <http://arxiv.org/abs/1706.05993> (cit. on pp. 4, 17, 18, 27).
- [17] Hosnieh Sattar et al. “Prediction of search targets from fixations in open-world settings”. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* 07-12-June (2015), pp. 981–990. arXiv: arXiv:1502.05137v3 (cit. on pp. 2, 3, 9, 17, 20, 21, 23, 25, 53, 54, 56, 61, 62).

- [18] D.L. Schacter, D. T. Gilbert, and D. M. Wegner. *Psychology (2nd Edition)*. New York: Worth, 2011 (cit. on p. 4).
- [19] Pierre Sermanet et al. “OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks” (2013). arXiv: 1312.6229. URL: <http://arxiv.org/abs/1312.6229> (cit. on p. 18).
- [20] Ali Sharif et al. “CNN Features off-the-shelf : an Astounding Baseline for Recognition” (). arXiv: 1403.6382 (cit. on p. 10).
- [21] Ali Sharif et al. “CNN Features off-the-shelf : an Astounding Baseline for Recognition” (2014), pp. 806–813. arXiv: 1403.6382. URL: https://www.cv-foundation.org/openaccess/content%7B%5C_%7Dcvpr%7B%5C_%7Dworkshops%7B%5C_%7D2014/W15/html/Razavian%7B%5C_%7DCNN%7B%5C_%7DFeatures%7B%5C_%7DOff-the-Shelf%7B%5C_%7D2014%7B%5C_%7DCVPR%7B%5C_%7Dpaper.html (cit. on p. 18).
- [22] “SUN RGB-D: A RGB-D scene understanding benchmark suite”. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition 07-12-June-2015* (2015), pp. 567–576 (cit. on p. 58).
- [23] Roberto Cipolla Vijay Badrinarayanan Alex Kendall. “SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation”. *CoRR* abs/1511.00561 (2015). arXiv: 1511.00561. URL: <http://arxiv.org/abs/1511.00561> (cit. on p. 58).
- [24] J. M. Wolfe. “Guided Search 2 . 0 A revised model of visual search”. *Psychonomic Bulletin & Review* 1.2 (1994), pp. 202–238. arXiv: NIHMS150003 (cit. on pp. 1, 2, 6, 7, 16).
- [25] Alfred L. Yarbus. “Eye movements and vision”. *Neuropsychologia* 6.4 (1967), p. 222 (cit. on pp. 1, 5–7).
- [26] Gregory J Zelinsky. “A Theory of Eye Movements during Target Acquisition”. *October* 115.4 (2009), pp. 787–835 (cit. on pp. 1, 4, 7, 8).
- [27] Gregory J Zelinsky, Y. Peng, and D. Samaras. “Eye can read your mind: decoding gaze fixations to reveal categorical search targets”. *Journal of Vision* 13.14 (2013), pp. 1–13. URL: <http://www.journalofvision.org/content/13/14/10.abstract?ct> (cit. on pp. 1, 2, 16).

Online sources

- [28] *Bing’s Visual Search decommissioned*. May 2012. URL: <http://www.liveside.net/2012/05/22/bings-visual-search-decommissioned/> (visited on 07/01/2018) (cit. on p. 1).

- [29] Tim Dettmers. *Deep Learning in a Nutshell: Core Concepts*. Nov. 2015. URL: <https://devblogs.nvidia.com/deep-learning-nutshell-core-concepts/> (visited on 04/06/2018) (cit. on p. 14).
- [30] Dr. Jun Lin and Dr. James Tsai. *The Optic Nerve And Its Visual Link To The Brain*. Mar. 2015. URL: <http://discoveryeye.org/optic-nerve-visual-link-brain/> (visited on 07/01/2018) (cit. on p. 1).
- [31] Edward S. Perkins and Hugh Davson. *Encyclopedia Britannica - Human eye*. Nov. 2017. URL: <https://www.britannica.com/science/human-eye> (visited on 07/01/2018) (cit. on p. 4).