UNIVERSITY OF TROMSØ UIT

FACULTY OF SCIENCE AND TECHNOLOGY
DEPARTMENT OF COMPUTER SCIENCE

Context-based image retrieval in Fronter learning environment

Jelena N. Larsen

INF-3900
Master's Thesis in Computer Science

February, 2011

Abstract

The Internet has become a natural medium for finding information and resources, and has probably become the most important tool in education and e-learning as well. Many educational institutions use on-line systems for uploading, creating and publishing educational content to students and pupils. Extended use of multimedia files, video, audio and image, as a part of the content is a growing trend and there is ever more a need to search for desired multimedia content. This causing challenges to both the on-line systems and its users.

To make multimedia content suited for search and retrieval it is imperative to organize and describe content well. Normally, users do not spend much of their time annotating and organizing content. Text-based search engines that are integrated into the educational on-line systems are normally not very suitable for search of multimedia content.

The specific concern of this project is to investigate and suggest solutions to how image context information can be collected and then used in image annotation and retrieval within educational content. The image context in this setting is defined as the image environment in which the image is used. For example it can be a collection of images or documents, the course(s), subtopic(s) and/or assignments where images are used as illustrations. The system that is designed and implemented in this thesis will use image context to describe the images content and its semantics and use this in its image retrieval. If there are multiple versions of an image it will also be used in the description and retrieval processes.

The evaluation of the system indicates that usage of the image context is very useful in describing the image content and its semantics. The system finds and retrieves more relevant images to the search than the integrated search engine in on-line educational portal. It has also good retrieval performance compared to the integrated one.

Acknowledgements

First of all I would like to thank my supervisor, Randi Karlsen, for supporting me in this work and for very constructive guidance throughout the entire period of the work with this thesis. Your availability, feedback, knowledge and sense of humour are very much appreciated.

I would also like to thank the teachers at the Department of Philosophy, Mariann Solberg and Roar Anfinsen, for their contribution to the test environment in this thesis. Thank you very much for spending your time with me and for your patient during the process.

I would also like to thank my employer and my colleges from U-VETT (The University's Centre for Flexible Education). Your encouragement and support means a lot. Special thanks to Mark Stenersen for his collaboration, discussions, and comments and laughs. He has also been a great help with English grammar.

Many thanks to my family who encouraged and supported me and has been patient with me throughout the entire period of my studies.

None of these people are to be blamed for any shortcomings in this thesis.

February, 2011
Jelena N. Larsen

Contents

List of tables and figures

Chapter 1

Introduction

This chapter is an introduction to the thesis. It will discuss the motivation behind the work, the problem and contribution and an overview of the approach. The last section of this chapter will include an overview of the organization of this thesis.

1.1 Motivation

Today, the Internet is a natural media for finding information and resources – it has become the most important tool in the conception of flexible education and e-Learning. Several educational institutions in Norway use the VLE (Virtual Learning Environment) systems to achieve these educational needs. VLE, in general, is the on-line educational system that provides a collection of tools for teaching, assessing, collaborating, publishing and general management. Most of these systems are open source and to some extent support the SCORM (Shareable Content Object Reference Model) [1] standard for uploading, launching and tracking courses.

Uploading files and creating content are the basic techniques used by teachers when publishing educational content to students- and pupils. Extended use of images as a part of the content is a growing trend. Images are used in almost all types of content and in almost all disciplines; medicine, biology, humanistic disciplines such as philosophy, history, language studies, etc. Over the years we have seen that image archives in online educational systems have grown dramatically causing great challenges to the systems and its users. Most challenging for users is the organization of image archives and consistent name principles as such that others could easily reuse them. Many users do not want to learn about file structures, organization and consistent naming principles. Some users name images by numbers, and some give images names that do not describe the content at all, for example, *05200.JPG* or *untitled.GIF*. As a result archives in educational systems become very messy and difficult to follow over time. It is very time consuming for users to click trough unstructured image archives to see if there are any relevant images. We also know that very little can be done in regards to change user's behaviour in structuring their image archives and learning them how to name images, but a lot of improvement could be made when it comes to image search techniques – techniques that will help users to find relevant images within the system in an easy and efficient way.

[1] http://www.scormsoft.com/scorm/overview

Most online educational systems offer simple text-based searching mechanism that is very efficient in searching for textual content. Normally, such search engines are the third part applications that are tightly integrated into the VLE. Solr[2], for example, is the search platform that is developed for text-search and rich document handling, and is tightly integrated with Fronter[3] VLE. But fully text-based search engines have a number of weaknesses specifically when it comes to indexing and searching images and other multimedia files such as video and audio.

In the past 20 years, research in multimedia information retrieval, especially with the focus on image retrieval, has resulted in many research and commercial image search engines. These systems can largely be divided in two categories: content-based (CBIR) and text-based (TBIR). CBIR systems focus on the content in an image such as shape, texture, colour, etc. TBIR uses textual content such as meta-data, manual annotations, and contextual information with an image. An overview and discussion of these approaches will follow in chapter 2. Google images[4] for example adopt both approaches: search and indexing visual content as face recognition, clip art, line drawing, colour, and search and indexing of textual information as image filename, anchor text, image caption, etc.

The integration of image search techniques within VLE systems would dramatically improve image searching results, user experience of the VLE, and ease the reuse of images in publishing educational content.

1.2 Fronter and DeStore

Fronter and DeStore are the systems that are used in this thesis.

Fronter is the VLE system that is used at the University of Tromsø for publishing and developing educational content. The test collection used in this thesis contains approximately 200 images and approximately 50 HTML documents where images are used. These are created in Fronter for the philosophy disciplines.

There is a growing trend of uploading and usage of large multimedia files such as video, audio and images as a part of the educational content. This causing great challenges to the centralized data storage in Fronter – as it has scaling problems that have effects on the user experience of the system. In addition to this – user data must be stored for an extended period of time, creating a continually expanding challenge for the service provider.

There is an ongoing project initiated by Fronter that will replace the centralized storage with the decentralized storage solution under centralized control – DeStore.

[2] http://lucene.apache.org/solr

[3] http://fronter.com

[4] http://www.google.com/imghp

In this thesis it is assumed that DeStore replaced the Fronter's centralized storage. Thus, the whole test collection in Fronter, i.e. all images and HTML files created in Fronter for the philosophy disciplines, are copied to DeStore. The system prototype implemented for this project uses this copy for automatic image annotation and search. The system is evaluated by measuring image retrieval performance in DeStore.

1.3 The goal and contribution

In this thesis I will study image retrieval based on the usage of context. Further I will focus on image search and retrieval in the VLE system used for the publishing educational content at University of Tromsø – Fronter. As a part of this work it has been made a study of how images are used in Fronter when developing educational content.

The goal of this thesis is to design, implement and evaluate the image search engine that will use context where images are referred in to automatically annotate images. The hypothesis one for this thesis is that context is useful in describing image content and its semantics. If an image is used within different contexts, e.g., about American Presidents, and about terror actions in the USA 9-11event, it will be annotated with the different contexts. The hypothesis two for this thesis is that gathering multiple contexts for an image will give to the system a better understanding of image content and its semantics and will enhance the systems retrieval performance.

The test environment for this thesis contains an image collection and the set of the HTML documents where images are used in; and was developed in Fronter for the philosophy disciplines at the University of Tromsø. Teachers at the Philosophy Department formulated the test queries for the collection, and defined sets with relevant images to each query in the test.

Evaluation of the implemented system for this thesis is based on the user-defined queries and user-defined relevant image sets. The systems retrieval performance has also been compared to Fronter's existing search system, Solr.

1.4 Approach

The automatic image annotation designed and implemented in this thesis can be divided into two parts. The first part consists of finding and extracting relevant information from the context images are used in. It is used in automatic image annotation. The second part is concerned about the processing query – retrieve all relevant images and do not retrieve non-relevant images to the query.

Based on the general observations, an image in an HTML document is typically semantically related to its context. This context might sometimes directly be used to illustrate some particular semantics of the image content, e.g., people, geographical places, buildings, etc. Sometimes it is related to the image subject or category, e.g., "*war and terror*", "*animals rights*", "*Sophists*", "*Painting*", etc. And sometimes images are used to illustrate the textual content. For example the image with the airplane draw by Leonardo da Vinci is used as an illustration to the history of philosophy.

The most important part of this approach is analysing the context and deciding what part of it might be relevant to the image. After the decision and definition of what parts of the context might be helpful in describing image is made, keywords from it are extracted, processed and used to automatically annotate the image. Extracted and processed keywords are further referred to as the index terms and used to build an index for an image. Each index term in the index is assigned a weight that indicates how important this terms in describing the image.

When the query is processed, the system uses the index to search for relevant images by matching index terms to the query terms. The similarities between the query and image are measured and images that the system finds relevant are retrieved and ranked.

This approach will be discussed in more detail in chapter 4.

1.5 Organization

The rest of this thesis is organized as follows:

Chapter 2 gives an overview of the background material for this thesis by representing the different image annotation techniques, review and discuss different approaches used for image annotations.

Chapter 3 gives an overview of Fronter and DeStore systems. Fronter is the VLE system that is used at the University of Tromsø for publishing and developing educational content. DeStore is the decentralized data storage system where the copy of the educational content and the Fronter image archive is stored.

Chapter 4 presents the approach of this thesis. It discusses, in more detail, the limitation and problems with existing approaches The definition and description of context relevant to image semantics in this thesis is given; and how it will be used to automatically annotate images.

Chapter 5 gives a detailed description of Fronter HTML editor and HTML design for the documents used for the test in this thesis. It gives also an overview of the system design and architecture and how the system handles queries.

Chapter 6 gives an overview of the implementation and more specific details.

Chapter 7 reports and evaluates retrieval results of the image engine system implemented in this thesis and compares results to the existing search engine system integrated in Fronter - Solr.

Chapter 8 concludes the system implemented for the project; discusses general experience with the work and possible future work.

List with the queries used in the test for this thesis are found in appendix A. Detail report with the retrieval results both for the implemented system and Solr is attached in appendix B.

Chapter 2

Background

This chapter represents the background material for this thesis. It will give an overview of image annotation techniques, represent and discuss different approaches used for image annotations.

2.1 Introduction

Over the last 10 years, uploading and usage of multimedia content such as video, audio and images, in different type of context is a growing trend. We upload videos to YouTube[5] and use it in promotion, education, private sharing, etc. We use Spotify[6] and Wimp[7] to stream and download music. We use Google Picasa[8] and Flickr[9] to manage, organize and edit our images and share them on the web with others. The challenge in publishing multimedia content is retrieving desired information/content. There are already a lot of commercial and research retrieval systems when it comes to multimedia retrieval. In this thesis I will focus on existing image retrieval techniques. But first a short overview of image annotation will be given to better understand the motivations behind different approaches within image retrieval.

2.2 Image annotation

Images can be associated with two kinds of information: visual and textual. Visual information is about colour, texture and objects that are illustrated in the image. Textual information can be divided into two categories: first category is the information that is "accidently" available within an image. This is meta-data such as size, resolution, date and time, location (can be available for photo taken by cameras with built-in GPS), etc., and text

[5] http://youtube.com

[6] http://www.spotify.com/no/

[7] http://wimp.no/site/web3/view.ftl?page=index

[8] http://picasaweb.google.com/home

[9] http://www.flickr.com/

in the document where image is used/embedded. Second category is textual information that is added by humans, or so called annotations. The goal of annotating is to assign images semantically meaningful information, for example image annotations as "*Vacation in Rhodos 2005, swimming park*".

2.2.1 Manual image annotation

Manual annotation is a very old-fashion tradition that began in the non-digital world: humans wrote associated texts to paper images or photos in books or albums. Manual annotation is a completely human oriented task. The advantage of this annotation approach is the precision of semantic level. At the same time manual annotations suffers from high cost, inefficiency and human subjectivity. For example annotation could be a moment of feeling at the time the image was taken. If a user bought a new car, and annotated taken picture at that moment with "*my lovely car*", maybe after a year the same user may feel different about the car and describe it as a "*boring ugly car*". Another example of user subjectivity is that different users can perceive the same image differently. Perception will often rely on peoples backgrounds, knowledge, maybe even work environments, family situations and so on. If you ask different people what the image in the figure 2.1 is about, you will get different answers. The anthropologist can associate the white person on the picture with his or her colleague who is on researcher tour. Some people might associate the same person with a missionary. But if you ask a girl from Gambia she would say that this is a typical tourist with little money, because she has a rucksack on her back [17].

Figure 2.1: Photo of Addis Ababa Main Street with Anne Britt Flemmen as the person with the rucksack [17].

In some cases it might even be difficult to describe the image content with words, e.g., painting of abstracts.

2.2.2 Automatic image annotation

Automatic annotation, also referred to as auto-annotation or linguistic indexing [3], is based on automatic adoption of textual information an image is available within by applying statistical classification methods. It could be metadata, text available on the same page as an image, image text and tag information. It seems a reasonable approach to indexing images, but it inherits the same weaknesses as manual annotation such as human subjectivity. In addition, the context the image is found within does not necessarily describe the images content, and will therefore result in a bad or irrelevant retrieve. Image collections such as private digital photo albums are poorly annotated if annotated at all and will result in an inadequate index.

2.2.3 Semi-automatic image annotation

Semiautomatic annotation is based on a combination of manual and automatic annotations. The standard starting point for this approach is a training database of images, where each image is annotated with a set of keywords (captured from the available context). The strategy is to create and refine annotations by encouraging users to provide feedback while examining retrieved results. Based on this feedback the system learns by automatically updating the association between the available keywords in the database and image feedback. The result of which is a set of updated keywords associated with each image added to the database [18].

2.3 Image retrieval approaches

Today, the biggest information repository - the World Wide Web (WWW) - is indexed and available for information retrieval on the web. Techniques for searching textual information have become very efficient and fast, and "*Googling*" became a recognized term for searching information on the web – even making its way into the Oxford Dictionary, but when it comes to image retrieval there are still a lot of challenges to be met. For example, how image retrieval systems extract and categorize content of an image? Is it possible to learn image retrieval systems to recognize and describe visual content of an image, and how to do it? Of course, the title of an image and its surrounding text might help to describe visual content and categorize it, but more often this technique is not good enough.

Another challenge lies in how users search for images. For example if one searches for "*Venezia*", Google Image will return many images of geographical places in Venezia, but also images of coffee/espresso machines, furniture, perfumes, clothes, cars, etc. All these images satisfies the search for the keyword "*Venezia*", but in a perfect world the system should intuitively know that Venezia is a geographical place, and thus, look only for images that illustrate this place. Another example is subject searches, for example "*War and ethics*". What kind of images would a user expect to be retrieved: covers to the books about ethics in war? Controversial photos containing dead children and women? Soldiers bearing guns? Propaganda images or the like? All these problems are featured in current image retrieval research.

In the next two subsections we will look at different image retrieval techniques, and how they attempt to resolve these challenges within image retrieval.

2.3.1 Text-based image retrieval (TBIR)

A lot of information about image content can be retrieved from the textual association of that image. It can be based on manual annotation that are keywords describing image content or an event. It can be text "accidently" available with an image like captions, subtitles, nearby text, anchor text, etc., or metadata available within the image. This text is used to index images by traditional text based techniques. In this case, similarities between images are measured by the text associated with the image. This approach is known as text based image retrieval and dates back to the late 1970s and the database management community. The main idea of this approach is that text in the document or a web page may somehow be related to the images that are embedded or referenced on the same page, as well as images pointing to this page, and people use natural language to express their queries.

Commercial search systems such Google Image Search and Yahoo! Image search extract keywords related to the images on a webpage and the image captions. These keywords can be found in many locations of the document, and systems that use a text-based approach will consider the importance of every keyword differently.

In WebSEER [12] keywords are extracted from filenames, image captions, alternative text, title attributes, hyperlinks and HTML titles. Words are weighted according to the probability that they contain useful information. For example words contained in the title tag of a HTML page have a lower weight than those in the alternative text, that is the ALT attribute of the IMG tag.

WebSEEK [13] uses Web URL addresses and HTML tags associated with the images and videos, from which it extracts key terms. These are used to classify images into subjects in the WebSEEK´s semantic ontology. Semantic ontology represents a hierarchy of image semantic classes and is constructed semi-automatically in that, initially, human assistance is required in the design of the basic classes. Then, periodically, additional candidate classes that are suggested by the system are verified with human assistance. According to the Chang, Smith and Meng [13], subject-based query is the most popular search method for images and videos, for example quires like "*War and ethic*", "*Social justice*". Unfortunately, WebSEEK allows only single word queries, and as a rule it is difficult to define a subject by just one word.

Diogenes [14] takes advantage of the full text and HTML structure of web pages. The system gives more significance to titles, words that are part of the image name or URL, and words that are enclosed in the same tags as the image or an alternative text. In addition to the special words, the body text of a web page is exploited, words relevant to the image is identified based on criteria such as frequency of occurrence, the word on the page related to the appearance of the word in the whole web, also known as the *tf*idf* (term frequency * inverse document frequency). For example, if a rare word appears frequently on a page then it should be very significant to that page.

In ImageRover [15] words appearing with specific HTML tags are given special importance by assigning a higher weight as compared to other words in the document. The system assigns different weights to the words appearing in the title, headers and the alt fields of the IMG tags along with words emphasized with the different fonts like bold, italic, etc.

It is obvious that making effective use of textual keywords can improve image retrieval, and make it possible to apply existing text retrieval techniques to image retrieval. Given a web page with an image and its surrounding text, there is a challenge though: how can relevant words be distinguished from non-relevant? We know that textual content may contain information that are not relevant to the image, or so called "noisy" information, that leads to poor retrieval performance. MARIE-3 [16] is the system that tries to solve this problem by developing tools that analyses surrounding image text. Initially the system has to decide if an image is a photograph or not. The assumption here is that photographed images are more likely to be annotated then others. The second step involves identifying keywords by examining text near each image reference for possible captions. The system designers observed that image captions often are marked to appear differently from the ordinary text. Firstly, near text have to be within a fixed number of lines in relation to image reference on the page (that is HTML document), and secondly, its marked according to the font family (e.g, Times Roman), font style (e.g., Italic), font size (e.g., 12 pt), text alignment (e.g., center), text colour (e.g., red), text state (e.g., blinking), and text significance (e.g., a page title). The system also considers alternative text, names of web pages that image refers to (anchor text) and name of the image file itself.

One of the main advantages of TBIR systems is that people can use natural language to express their queries. According to the paper by Thijs Westerveld [9], most of the users are interested in semantic entities rather than visual appearance. Another main advantage of the text-based retrieval is that image annotation contains a semantically meaningful description or information that is difficult to express by visual features.

But text-based image retrieval has some limitations. First of all, textual description or annotation of an image is generally a manual task and has its limitations. Annotation is very time consuming and does not scale for the large image repositories such as WWW. Also, manual annotations are very subjective and depend greatly on the user and user's perception of the image content. Documents can discuss the same subject using different words (synonyms) or use the same words describing different concepts. That is - for the same image content different users may perceive it differently. Keywords can sometimes say more about the person who assigned the keywords then they do about image [6], for example the background knowledge, the work environment, if user is a parent or a child. Tools for automatic annotation and standards for metadata can help to solve this problem. Secondly, text assumed to be related to an image not always does. In this case it is considered as "noise" and leads to poor retrieval performance. Thirdly, text-based approaches may be especially insufficient when users are interested in the visual components of the image that consist of several objects. In this case annotations can end up with a long list of attributes. In addition there are image properties such as texture, compositions and other objects that are difficult to express by words. In this case, use of visual image features can do the job.

2.3.2 Content-based image retrieval (CBIR)

Instead of manually annotating images by the use of keywords, CBIR systems automatically extract and index visual or low-level features of an image such as colours, shapes, textures, faces, etc. For each visual feature, there exist multiple representations that are application dependent [1,2,3,4,5,6]. For example in a fingerprint recognition application texture and features could be sufficient; for a face retrieval application, shape descriptors may do the job.

With CBIR images are retrieved based on example images or drawings, also called query-by-example. QBIC (Query by Image and Video Content) [7] is the first commercial content-based image retrieval system. It allows for queries on large image and video databases based on example images, sketches and drawings, selected colour and texture patterns. VisualSEEK [8], for example, lets user to submit a query based on image regions and their colours, size and spatial locations. For example: red-orange colour on the top and blue-green region at the bottom of the sketch. Google's Picasa make use of visual features as face recognition, finding and removing red eyes.

The motivation behind the CBIR system is obviously weaknesses in manual image annotations. Firstly manual annotation is difficult to apply to large-scale image collections because its time consuming and costly. Secondly, manual annotation depends on subjectivity of human perception. That is – for the same content image different users may perceive it differently [1, 4]. And in some systems text based annotation is not relevant or meaningful. CBIR systems are well suitable within medicine and criminal investigation, for example medical diagnoses based on the comparison of X-ray pictures, or finding the faces of criminals from video shots of a crowd, finding similar images in a copyrighted image databases.

But CBIR systems have also some limitations. Requesting the user for image examples or sketches is not very convenient. Today, people are familiar with searching for desired information including images, video and music by keywords, or "*Googling*" information. CBIR systems are not able to translate image content to the textual description of an image, and they are also not especially useful when it comes to searching images for a specific event or a subject, or where visual content of an object can vary a lot.

2.4 Semantic gap

In the previous section two image retrieval techniques were described – CBIR that is based on extracting low-level features such as colour, texture, shape, and TBIR that is based on extracting high-level features, such as available text within an image, to categorize and identify images. The aim of both techniques is to make image retrieval as efficient as possible. However, it is still difficult to extract objective high-level concepts either from images or from their surrounding text. Human beings are much better than computers to extract and make use of semantic information from images. The lack of coincidence between information that can be extracted from visual data and the interpretation that a user assigns to the same data in a given situation is known as *the semantic gap* [2].

The key issue in the image retrieval is how to adopt/derive high-level concepts automatically from the image content and its surrounding text. As usually, text has a clear semantic meaning, but this is not the case with the image content. Analysing images require a reflective thinking that computers are not capable of doing. However, the use of text involves some problems too. First, if an image is embedded in a textual document or a web page, there is generally a lot of text, and the system has to decide which words are related to the image and which are not. Second, even if words in a given textual document are related to the image content, subjectivity of using words in a given document for a given image can be a problem. This problem is also referred to as *synonymy* and *polysemy* [19]. *Synonymy* is used to describe the fact that there are many ways to refer to the same object, e.g., subject can also be issue, matter, case, business, course, etc. The prevalence of synonyms tends to decrease the recall performance. *Polysemy* refers to the fact that that most words have more than one distinct meaning. For example subject can mean theme and matter, but it can also be used in a court to force upon someone. *Polysemy* is a factor underlying poor precision performance.

To narrow down the semantic gap many approaches have been developed. One approach is to combine both low-level features and textual features. Google Image Search and Yahoo! Image search is a good example of this approach. All systems that were mentioned in the subsection 2.2.1 are also based on the same approach. In order to mitigate the problem with the subjectivity, automatic annotation of images could solve the problem. Latent Semantic Indexing (LSI) [9, 15, 19] automatically indexes images with linguistic terms based on statistical model comparison. For example images annotated by words "*reservation*", "*double room*", "s*hower*" and "*breakfast*" are related to other images about hotels. LSI approach as a rule is used together with the CBIR approach to narrow the semantic gap.

2.5 Context

We have to understand what context is, and how it can be used within image retrieval. Different understandings and definitions have been used in an attempt to define context, referring to it as situations, locations, user's emotion state, environment, temperature, date and time, etc. Some consider context to be a user's environment, other – application's environment. A definition of context suitable for this thesis/our approach is given by Dey [20]:

> *Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object considered relevant to the interaction between a user and application, including the user and application themselves.*

In other words, context can be everything about situation that is relevant to the application and its users. In this thesis entity is an image, and context is all relevant textual information that describes an image in an HTML document.

Image context can be divided into two parts: capture context and user context. Capture context is typical metadata that is created with the image. Images taken by digital cameras store a lot of information in the EXIF header, such as location (if the camera is equipped with GPS), date and time of creation, camera information and settings such as shutter speed, white balance, brightness, flash etc. Images taken by sensor cameras might store information such as light, temperature, movement, pressure, etc. User context as a rule is textual information that is related to an image in a document an image is used within.

If we look at information that could be related to an image embedded in HTML documents, there are obvious parts of the textual content that might be well related to the embedded image. These are:

- Image title *<img src=""title="">*. Image filename or/and image title that could be obtained from the SRC attribute or/and from the TITLE attribute in the IMG tag. Image filename and image title does not necessarily contain the same keywords. For example: an image with the filename *P0250.JPG* can have an image title such as "*the sketch of Agora*". In the real world, many users don't care about writing anything in the title attribute in the IMG tag.

- Image ALT (alternative text) *<img … alt="">*. This attribute is meant to describe the abstract of the image content. For example an image with the filename *P0250.JPG* illustrates a sketch of Agora and could have an alternative text such as "*En skisse av det opprinnelige Agora, en åpen forsamlingsplass, slik det forelå på sofistenes tid*". In reality, many users don't care about writing alternative text in the IMG tag.
- Page title *<title></title>*. Since images are mainly used for enhancing a Web page's content, page titles should most probably be related to the image content. It is usually a short sentence that could summarise the Web page's textual content.
- Image caption – that provide the most semantics about an image. Ideally this is the text that is bellow or above the image, and usually differs from other text by its position, font and style. It can vary from few words, or a sentence to a paragraph.
- Headings *<h1>*, *<h2>* and *<h3>* that precede or follow an image. Usually a short sentence that summarises the content of a paragraph.
- Words extracted from the URL of an image.

There are also other parts of the HTML tags that can provide some information about an image, such as HTML metadata and textual content of the whole HTML document. But this information is often excluded from indexing images because it contain too much unrelated information, and indexing the whole HTML document for each image can end up with a very large database and is not expected to be proven as an efficient solution.

2.6 Information Retrieval

For many years, text and images were manually indexed at the end of a book. In 60's and 70´s initial exploration of text retrieval systems were introduced for small scientific corporations, law and business documents, and libraries. In the early 90´s we saw the introduction of the World Wide Web, which changed the way people shared, published and searched available information. The Web became a universal repository of human knowledge and culture, and how to find useful/relevant information on the Web became a main research area within Information Retrieval. Many automatic indexing and retrieving techniques, within research and also commercial, were developed for the Web – like Altavista, Google and Yahoo search engines.

Generally, Information Retrieval (IR) is about the retrieval of unstructured data or getting what user want when he/she wants it from an archive of unstructured data. In general it is retrieval of text documents, audio, video and images. Baeza-Yates and Riberiro-Neto [21] define IR as a system that deals with the representation, storage, organization of, and access to information items. The representation and organization of the information items should provide users with easy access to the information in which the user is interested. User information needs has to be translated to the query, which can be processed by the search engine (or IR system). Given the user query, the primary goal of an IR system is to retrieve all the documents, which are relevant to the users query while retrieving as few non-relevant documents as possible.

2.6.1 Components of the IR systems

Figure 2.2 illustrates what a typical IR system looks like. The diagram shows three components; input, processor and output.

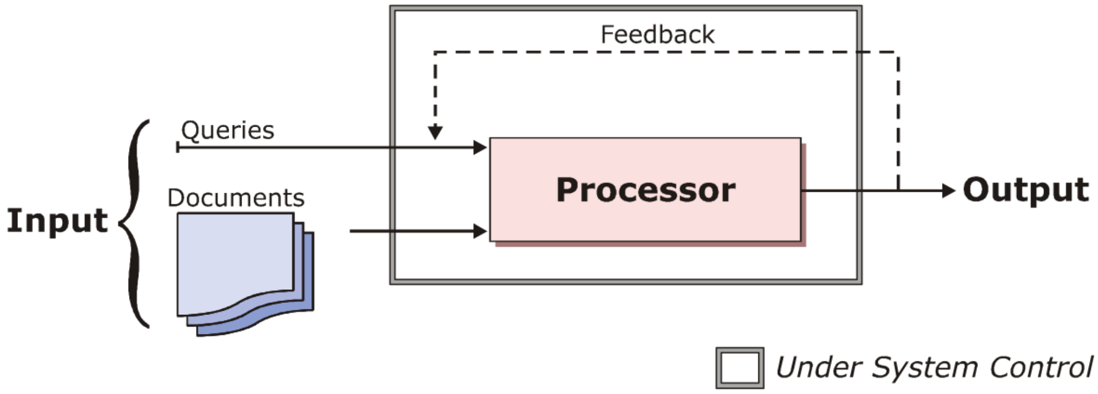

Figure 2.2: Components of IR system [21]

Input

Inputs are documents and queries. Documents represents a data collection crawled from the Web for example, while queries are user-defined keywords that describe his or her information needs. Documents in a collection are translated into internal representation - represented by a set of keywords. Such keywords can be extracted directly from the text from a specific part of a document or might be specified by a human subject, or be a full set of words for the entire document. These keywords represent a *logical view of the document* [21]. Usually, different normalization techniques are applied to the extracted set of words, such as elimination of stop-words, or high frequently words, and stemming. Stop-words are articles and connectives, for example "a","at", "are", "of", etc., that appear in a document very often but do not give any meaningful information about the textual content of the document. Use of stemming reduces distinct words to their common grammatical root, for example words such as "walk", "walking" and "walker" will be represented as "walk" for each word. Applying normalization techniques reduce complexity of the document representation and transform logical view of the document to the set of index terms. The same normalization techniques are applied to the queries.

Processor

Processor is concerned with the retrieval process that involves structuring and classifying of internal representation of documents or index terms. Once the logical view of the documents is defined, the database manager assigns weights to the index terms and builds an index of the text. Different models building index structures may be used, but the most popular one is the *inverted index file* structure [21]. Index file structure should provide fast, efficient and effective search and retrieval. This part of the system then generates a set of documents that best match the user information need.

Output

Output is a set of ranked or unranked documents that best matches the query.

2.6.2 Term weighting and measuring models

An index term is a document word whose semantic helps in remembering the document's main themes or subject. Thus, index terms are used to index and summarize the document content [21]. Indexing documents is not a new phenomenon; we can still find indexes at the end of the many books.

Given a set of index terms for a document, not all terms are equally useful in describing a documents' content. For example, given a collection of documents containing 100 000 documents. A word that appears in every document of the collection is completely useless as an index because it does not tell anything about which document might be relevant to the user. On the other hand, a word that appears only in five documents of the collection is quite useful because it considerably narrows down the space of the documents that a user might be interested in. This effect is captured through assignment of numerical weights to each index term of the document. There are many approaches for assigning weighs to index terms and measure similarity between documents and queries. Boolean, vector and probabilistic are the classic models within IR.

Boolean model

Boolean model considers index terms to be present or absent in a document [21]. As a result, the index term weights are assumed to be all-binary, i.e., 0 or 1. A query allows the user to specify their information need using a complex combination of three connectives: NOTs, ANDs and ORs. The disadvantages of this model are obvious, e.g., the Boolean model predicts that each document is either relevant or non-relevant and thus there is no notion of a partial match to the query conditions and no ranking of documents. It is also very difficult for user to form a good search request. The main advantage of the Boolean model is the "*clean formalism*" [21] behind the model and its simplicity – either document is relevant to the query or not.

Vector model

Vector model assigns non-binary weights to the index terms in the queries and documents and use vector space to measure degrees of similarity between each document in the collection and the user query. Retrieved documents are ranked in decreasing order of the degree of similarity, taking into consideration documents that are partially matching the users query.

Document d_j and query q are represented as a t-dimensional vector. The vector model evaluates the degree of similarity of the document and query as the correlation between vectors $\vec{d_j}$ and $\vec{q}$. This correlation can be quantified by the cosine of the angle between two vectors [21]:

$$SIM(d_j, q) = \frac{\vec{d_j} \bullet \vec{q}}{|\vec{d_j}| \times |\vec{q}|} = \frac{\sum_{i=1}^{t} w_{i,j} \times w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^2} \times \sqrt{\sum_{i=1}^{t} w_{i,q}^2}}$$

Where $w_{i,j}$ is the weight value of the index term *i* of the document d_j and $w_{i,q}$ is the weight value of the index term *i* of the query *q*, and where *t* is the total number of index terms in the system.

As an alternative, the *inner-product* or *dot-product* between two vectors are often used as a similarity measure. If all the vectors are forced to be unit length, then the cosine angle between two vectors is the same as their *dot-product*. If $\overrightarrow{d_j}$ is the document vector and $\vec{q}$ is the query vector, then the similarity between vectors can be represented as [22]:

$$SIM(d_j, q) = \sum_{i=1}^{t} (w_{i,j} \times w_{i,q})$$

To compute rankings there is a need to specify how index term weights are obtained. There are many different ways to do it, but the most popular one is *tf –idf weighting*. This approach is based on the frequency of occurrence in a collection of documents (*idf* - inverse document frequency) and individual documents (*tf* - term frequency) and document length (*N*). Term frequency measures how important a term to a document and value of it is a number of a term occurrences in a document. Inverse term frequency allocates term importance which is inversely proportional to the total number of documents containing that term and covers two core principles: 1) the higher document frequency (*df*) is, the less discriminating that term is and 2) lower document frequency a term occurs, the more discriminating that term is [21]. This means that *idf* gives high value to the terms that occur infrequently:

$$idf_j = \log\left(\frac{N}{df_j}\right)$$

Where *N* is a number of documents in the collection (document length) and df_j is the number of documents (document frequency) that contain the term *j*.

Tf-idf weighting schemes is given by:

$$w_{dj} = tf_{dj} \times idf_j$$

Document length is also used for normalization of term frequency values. For example, when documents in the collection are varying in the lengths, longer documents tend to score higher since they contain more words and words repetitions. Usually document length is normalized. There are two simple techniques to it: 1) taking logarithm of term frequency or divide term frequency by the maximum value of the term frequency in the whole collection. *Tf-idf* weighting scheme will be given by:

$$w_{dj} = \log(tf_{dj}) \times idf_j \text{ or}$$

$$w_{dj} = \left(\frac{tf_{dj}}{\max(tf_{dj})}\right) \times idf_j$$

The advantage of the vector model is that it allows the retrieval of documents that approximately match the query and it ranks documents according to their degree of similarity. It improves retrieval performance. But the disadvantage of the model is that it does not assume dependencies between terms, e.g. "*Be or not to be*".

Probabilistic model or binary independence retrieval (BIR)

The fundamental idea of this model is the ideal answer set to a query that contains exactly the relevant documents and no others, and document in a collection are ranked by decreasing probability of their relevance to a query [21]. Since the true probabilities are not available at the starting point, BIR estimates that the probability of relevance of documents to a query by the ratio of probability when documents is relevant to the query and probability when it is not: P(d_j relevant to q) / P(d_j non-relevant to q). The advantage of this model is that a set of documents ranked in decreasing order of their probability of being relevant. The main disadvantages are that relevance is being guessed; it does not take into account how important the term is to the document (term frequency), and this model assumes term independences as the vector model.

2.7 Measurements of Image Retrieval

Image retrieval is a subset of information retrieval and therefore inherits many of the aspects within IR. The main goal of all retrieval systems is to be as effective and precise as possible in retrieving desired information to the end-user, in other words to retrieve all relevant images (precision) and not to retrieve non-relevant images (recall). Let |R| be the set of relevant images for the query I. Let |A| be the answer set for the query I. Let |Ra| be the set of images of the intersection of sets A and R (shown on figure 2.3).

Then, precision is a fraction of retrieved images that are relevant [21]:

$$precision = \frac{|Ra|}{|A|}$$

And recall is a fraction of relevant images that have been retrieved [21]:

$$recall = \frac{|Ra|}{|R|}$$

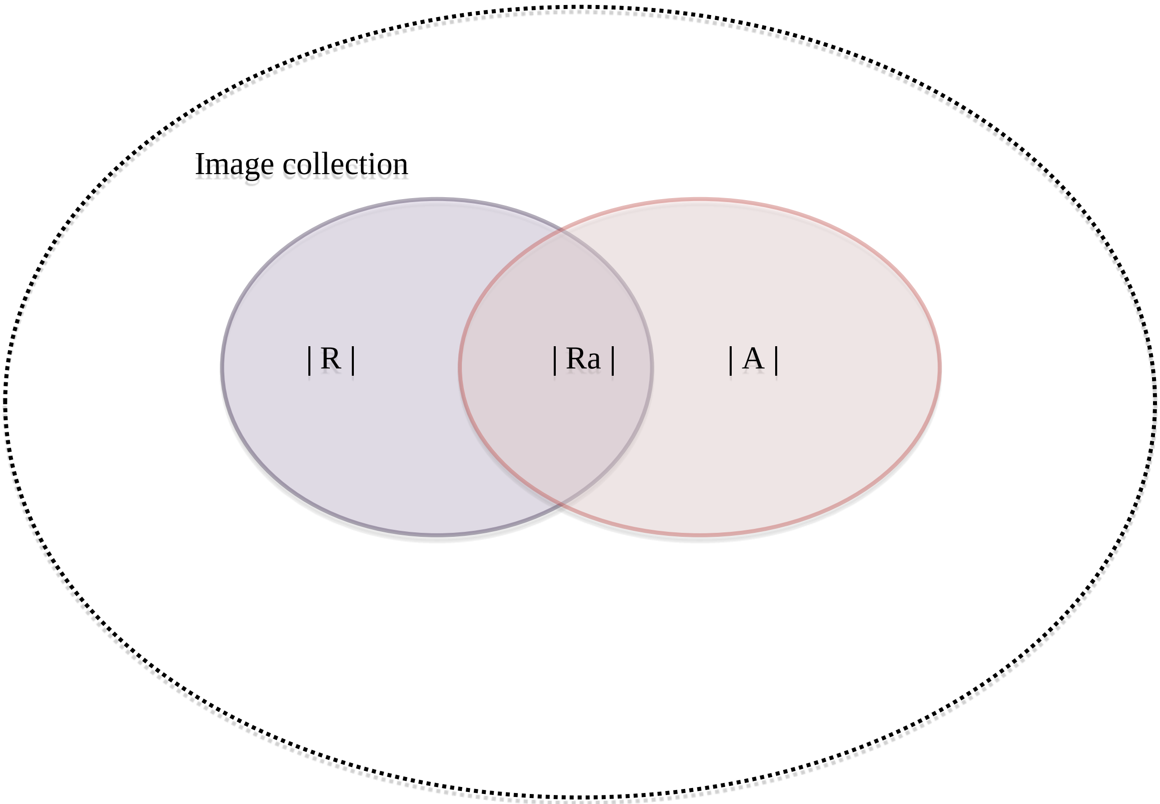

Figure 2.3: |R| is relevant images, |A| is answer set, |Ra| is relevant images in the answer set

Relevance is extremely subjective when testing the effectiveness of an image retrieval system. Similarity criteria may vary from user to user. Whereas one person may consider the same image is relevant, another person may consider the same image as not relevant. IR research has shown that the precision and recall follow an inverse relationship. In many situations, improvement of one leads to the deterioration of the other. Both recall and precision are set oriented measures and have no notion of ranked retrieval. But precision and recall works very well on a fixed and relatively small set of documents/images that has predefines queries and predefined set of all relevant documents to the query set such as reference collections, for example TREC[10]. Traditionally result is summarized as precision-recall curves, or precision-scope curves.

But recall and precision are not always the most appropriate performance measures for evaluating retrieval performance. The recall and precision scores are often combined into a single measure known as harmonic mean *F-score*. *F-score* gives an average score of the system efficiency without specifying the value of recall and precision and is computed as:

$$F - score = \frac{2 \times precision \times recall}{precision + recall}$$

[10] http://trec.nist.gov

Chapter 3

Introduction to DeStore and Fronter

This chapter will give an overview of two systems. The first one is the VLE system Fronter that is used at the University of Tromsø for publishing and developing educational content. And the second one is DeStore that is a data storage system that offers a decentralized storage solution under central control.

3.1 Introduction

As was mentioned in chapter 1, the Internet has become a natural medium for finding information and resources and has probably become the most important tool in conception of flexible education and e-learning. Teachers at the University of Tromsø use the VLE system Fronter to achieve their educational needs such as uploading files and creating educational content to students/pupils. Extended use of multimedia files such as video, audio and image as a part of the content is a growing trend. The storage capacity needs in Fronter have grown dramatically causing great challenges both to the system and its users. Expanding storage capacity is a minor cost to Fronter, but centralized data storage is not always a satisfying solution – since it, as a rule, has severe scaling problems that affect the user experience of the system. This poses new challenges for the communication infrastructure and the central server systems. Adding to that – user data must also be stored for an extended period of time, creating a continually expanding challenge for the service provider.

DeStore is a data storage system that will catch the problems described above by providing a decentralized storage solution under central control.

3.2 Fronter overview

Fronter is a Virtual Learning Environment (VLE). The main goal of this system is to provide "easy-to-use" tools for learning and online collaboration. These tools cover the fundamental areas of educational needs such as personal work, learning, collaboration, publishing and administration:

- By personal work means managing personal content such as storage of files, managing contacts, virtual meetings, and portfolio. (Figure 3.1).

- Learning tools provide learning activities such as creation of tests, hand-in assignments and possibility of importing external learning materials. (Figure 3.2).
- Collaboration tools focus on collaboration and communication within a group such as creating of documents that allow for multiple authors to work together and comment each other's work. Discussions and forums allow users to share their opinions and ideas. (Figure 3.2).
- Publishing tools allow users to create, upload, edit and publish their work in a variety of ways. (Figure 3.2).
- And, finally, administration tools provide institutions with the simple structure for access, rights and roles. (Figure 3.3)

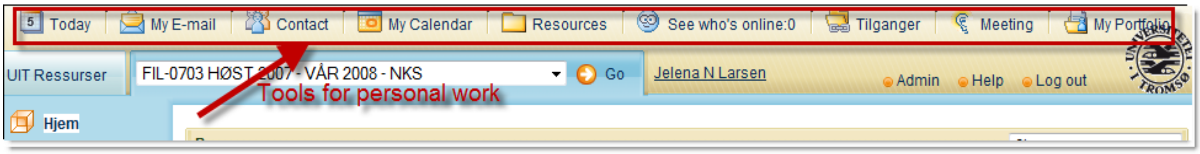

Figure 3.1: Illustrates some of the personal tools available in Fronter

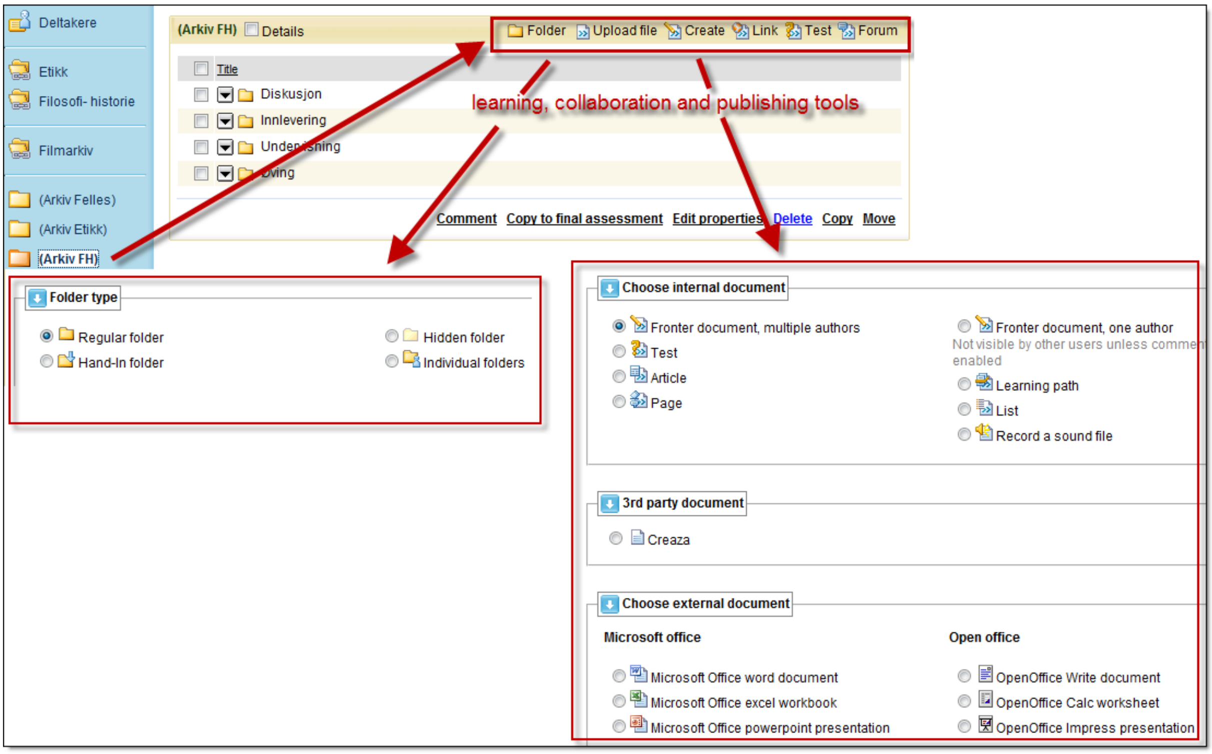

Figure 3.2: Illustrates some of the learning, collaboration and publishing tools available in Fronter

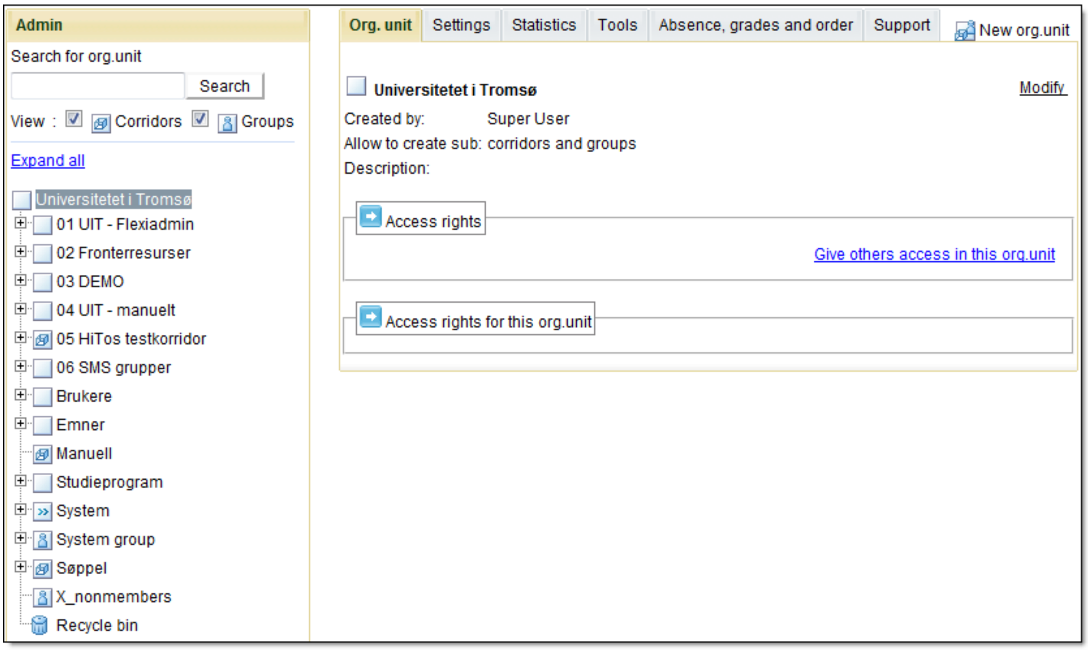

Figure 3.3: Administration tool in Fronter. Illustrates organization of information at University of Tromsø

Fronter offers centralized storage that is divided into two parts:

1) Internal storage stores data created in Fronter and thus are system specific, such as forums, tests, links, internal Fronter documents (pages, learning path, articles, etc. See figure 3.2).
2) External storage stores files uploaded to the system.

As mentioned in section 3.1, uploading multimedia files in Fronter is a growing trend, and the volume of produced content for external storage in Fronter grows considerably, challenging both users and system. Many Fronter users already experience the system to be slow when uploading and requesting files. Files bigger than 50 MB are impossible to upload to the system. And lastly, institutional costs for extending storage in Fronter grow in accordance with the user storage needs.

Another challenge to the system and its users is searching Fronter for the desired content. Fronter integrated the third part search engine Solr that offers indexing and searching internal and external storages. Solr is an Open Source full-text search engine based on Lucene Java search library[11], which provides APIs similar to web-service for other application systems, like Fronter. (Documents are indexed via XML over HTTP and queried by standard HTTP GET request.) The problem with the Solr is that it indexes and searches images and other multimedia files or non-textual files only by the words available with the file properties in Fronter, e.g., title (normally corresponds to the filename), file description, author, and path to the file in the Fronter Archive (see figure 3.4).

[11] http://lucene.apache.org/java/docs/index.html

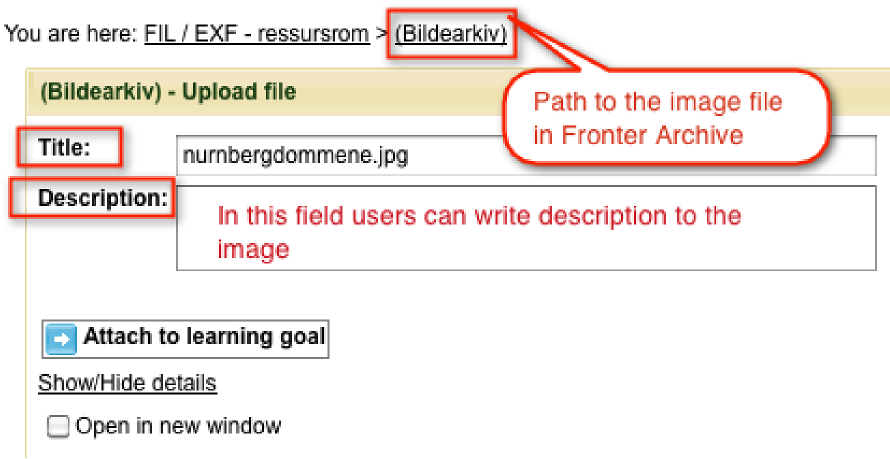

Figure 3.4: Illustrates properties for the image file "*nurnbergdommene.jpg*" in Fronter

If an image has no available description, and title and keywords in the path give no clue about image content, Solr has no chance to see if an image can be related to users query request. As uploads and use of multimedia files are growing, it is important that Fronter develops or integrates a powerful search engine that allows indexing and search not only for pure text documents, but also for the multimedia content.

3.3 DeStore overview

DeStore is a decentralized data storage system that tries to solve the problems attached to the centralized storage as described in the section 3.1. DeStore is built by a set of independent domains, where each domain consist of a set of nodes that are organized in a self-administrating peer-to-peer network with a centralized controller, or commander, as shown in the figure 3.5. A domain represents a WebDAV (Web Distinguished Authoring and Versioning protocol)[12] compatible interface that allows users directly access files stored within domain. Domain includes two types of nodes –slave and master [23, 24].

Slave node is responsible for the storage and replication of data. It can also provide HTTP or WebDAV access to data. All meta-data is stored locally. The slaves log all resource modifications and then use logs to synchronize changes with the other slaves.

Master node is responsible for data management in DeStore including data replication, load balancing, resource locking, meta-data, and access control. Master gathers all information from each slave node regarding their replicated data.

Commander is the central controller that can be assigned to more than one DeStore domain. Commander is responsible for both user and node authentication. Commander also keeps track over master nodes in all attached to its DeStore domains and performs a master selection.

[12] http://www.webdav.org/

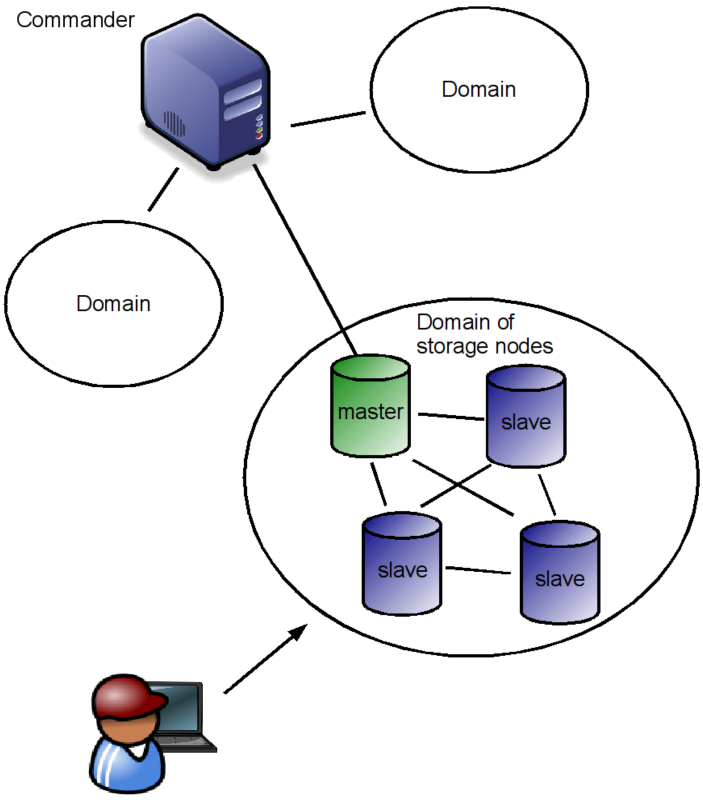

Figure 3.5: DeStore architecture [23]

All communication in DeStore use TSL (Transport Layer Security). The commander holds authorized (SSL) keys for all participating nodes. If a node key is stolen or compromised, other nodes stop communications with the compromised node until it gets a new key from the commander. Communication between commander and DeStore domains go through the master. Master node verifies keys periodically with the commander, caches keys and provides them to the slaves. In case of revocation, slaves periodically check keys to their neighbours that they get from the master [23, 24].

User authentication can be performed by commander itself or can pass to another system as Fronter. In case of authentication by another system, user must provide a valid ticket, and commander must be able to validate it.

3.4 DeStore role in Fronter

To resolve the problems attached to the centralized storage there is an ongoing project in Fronter that will implement integration to DeStore. DeStore will replace Fronters external storage that is used for storage of uploaded files. In theory Fronter will install DeStore boxes in all educational institutions that use the VLE system. Storage boxes at each institution will represent a DeStore domain with the centralized commander installed at Fronter, which will be responsible for both user and domain administration. DeStore boxes are meant to be relatively cheap PC’s with possibility to easily extend disk space. The philosophy of DeStore is that it is a self-administrating and self-repairing system with low need of administration and low storage costs.

Integration between DeStore and Fronter will resolve problems and limitations attached to the centralized data storage that is used by Fronter today. In theory, users will experience the system to be more efficient, and it will be possible to upload big multimedia files such as high-resolution images, videos and audiofiles while institutional costs of Fronter data storage will be dramatically reduced.

But integration of DeStore in Fronter will not resolve challenges regarding efficient retrieval of information, and the specific concerns of this thesis is image retrieval. DeStore does not offer retrieval of information and its assumed that Fronter will extend Solr search engine integration to DeStore, and thus challenges of image retrieval will remain in the system.

Chapter 4

Approach

This chapter will give an introduction to the challenges users experience today when it comes to the image search. It will give a definition and description of the image relevant context for this thesis and present an approach of how it will be used to annotate images automatically.

4.1 Introduction (Problem definition)

In general, a well knowing problem concerning image retrieval it that users very often have an abstract notion of the type of image they are searching for, and therefore need image retrieval based on semantic concepts. However, current image retrieval techniques have limitations that make it difficult to search for images based on semantic understanding of what the image is about. This problem is referred to as the semantic gap – mentioned in section 2.4.

This thesis it will implemented a prototype of the image search engine (ImSE) that automatically annotates images and support image searches based on these annotations. But first, the users and user environment has to be considered and described.

1) Users are academics that use Fronter for developing and publishing educational content. For example developing and publishing presentations, images, links to the relevant websites for the subjects they are teaching, etc. As the amount of published files grows there is a need to search files for relevant content, in this case images. Users can be divided into two categories: some of them are very clear about what kind images they want to search for, for example: "Socrates", but some of them are not, but they are clear about a category or a subject images should illustrate, for example: "Images about war and ethics". In both cases the goal of the search is to find images that might be relevant to the subject defined by the users.

2) The data collection contains about 50 HTML documents and roughly 200 image files uploaded to DeStore.

3) Users want to express their queries by keywords and expect the system to retrieve relevant images to their queries. There is no interest to formulate queries as image examples or visual features of images.

Contribution of this work is to implement image search on DeStore and analyse how contexts images are used within influence relevancy and user experience of the ImSE. Relevance and recall of ImSE will be compared to Fronter's existing search system Solr.

4.2 Scenario

Teachers from the Department for Philosophy at the University of Tromsø use Fronter today as a tool for courses in philosophy. In some courses Fronter is mainly used to publish resources, and for dealing with student hand-ins. And in some courses all interaction and communication between students and teachers are strictly organized in Fronter. In both cases the main tool in Fronter for organizing and publishing information and recourses is Archive where all teachers in the department have access rights to re-use others resources and publish their own resources. Archive for images is created as a tool on its own and currently contains about 200 images. The department of Philosophy uses a lot of images in their course materials in many different types of contexts. For example when discussing ethical issues in war and politics, or visualizing arts as a part of history, images that are used as a symbol to the textual content, etc.

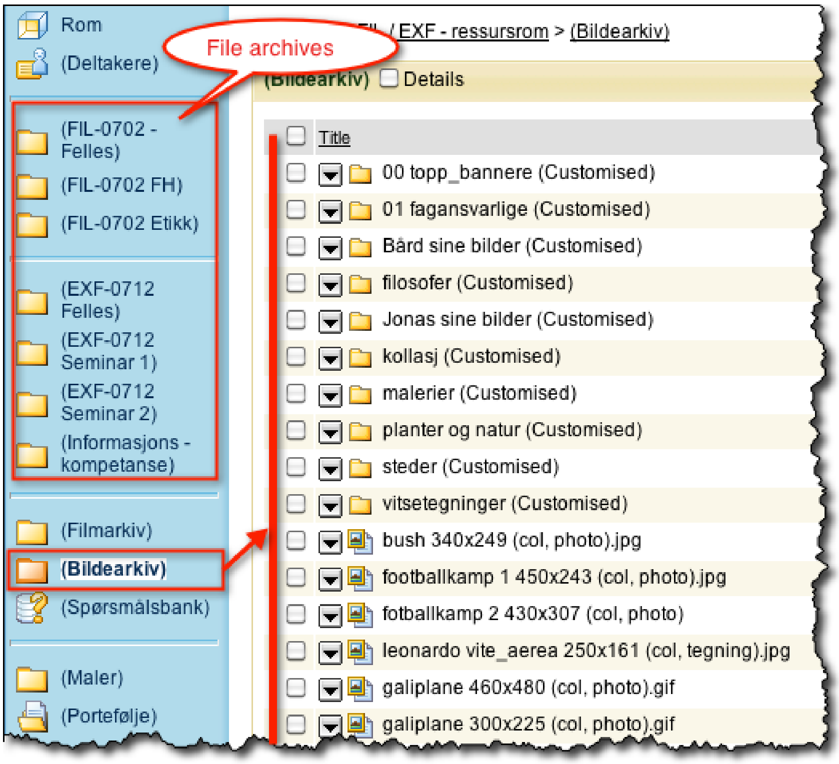

Figure 4.1: Archives in Fronter

Bildet til venstre: President Bush taler til nasjonen på 5 årsdagen for 11. septemberangrepet. Til høyre: Osama bin Laden, medansvarlig for angrepet.

I dialogen Laches diskuterer Sokrates hva mot er med to generaler, Laches og Nikias. Denne diskusjonen har relevans også for moderne politisk debatt og retorikk, hvor begrepene mot og feighet fortsatt er sentrale. Og det pågår også i dag en debatt om hva den korrekte bruken av disse ordene er. Kan man f eks karakterisere terrorister som modige? Eller er de snarere feige?

I sin første offisielle kommentar til terrorangrepene 11. September sa president Bush:

"Make no mistake: The United States will hunt down and punish those responsible for these cowardly acts."

Bush brukte med dette en karakteristikk som også f eks presidentene Reagan og Clinton brukte om terrorangrep som rammet USA. Feighet og mot er begreper som ofte tas i bruk i kritiske og dramatiske situasjoner.

Figure 4.2: Example of the HTML document with embedded images in Fronter

The main challenge of the archives is organizing it consistently so that it easily could be re-used by others. Some teachers prefer to create their own folders, some name images by numbers such as *05200.JPG*, some give images names that do not describe the content at all such as *untitiled.GIF*, and so on. It is very time consuming to click through every folder in the archive to see if there are any images that might be of interest or with any relevance. Fronter offers text-based search of the stored content with Solr – that has several limitations when it comes to image retrieval. For instance *05200.JPG* could be an image of Socrates but Solr would not retrieve it because the metadata does not contain any textual information about it.

Two categories of queries will be tested: query with keywords such as "*Socrates*", "*Platon*" or "*Agora*". These queries are very concrete and there is no doubt about what kind of image content is relevant to these queries. For example if 05200.JPG, an image of Socrates without textual information, is used on a page that describes the philosophies of Socrates and ethical issues, the image search engine will use this context to decide/learn that this image is about Socrates and retrieve it in a search query for "*Socrates*".

Another category of queries is the subject search. Subject search means that keywords in the query does not describe any concrete object as for example "*Socrates*", "*philosophers*", "*Agora*", but rather describes an event or subject. For example, subject search for "*Terror and courage*" will return images of President Bush and Osama bin Laden that might be highly relevant to the subject, if one knows about the 9-11 terror attack in USA, and the speech President Bush made to the nation, where he talked about courage. Another good example of subject search is “*Moral in war*”. Images that are relevant to this subject are highly dependent on users’ perception and understanding of this subject matter and thus very subjective.

With introduction of ImSE, users/teachers can easily search for the images by typing keywords in the search field. The search result will be represented to the users as a collection of thumbnails to the images with the filename, size of the pixels and image file.

4.3 Context

Based on general observations, an image in an HTML document is typically semantically related to its surrounding text. This surrounding text in some cases is directly used to illustrate some particular semantics of the image content, i.e. what objects are in the image, what is happening and what place. In some cases surrounding text has relation to the image subject, and in some cases images are used to illustrate the semantics of the textual content, as it was illustrated in the examples of the previous subsection. In particular in a HTML document, certain components are expected to provide more information than others. In chapter 2 different text-based approaches were described where surrounding text or text that is available with the image is used to annotate images automatically. These include image captions, its title, document title, image file URL etc. Also it was discussed the subjectivity and the “noise” of the different components of surrounding text that can lead to bad retrieval performance.

For this thesis, it is considered that the context an image is used within – is its surrounding text that is abstracted from the HTML document. It includes:

- Filename of the image. It is assumed that in some cases a filename can provide meaningful information about an image, e.g., “*Socrates_death.JPG*”.
- Image URL. Words extracted from the image URL might help to classify or categorize an image, e.g., URL *http://destore00.uit.no:8235/.../filosofer/gadamer-sm 200x262 (col, photo).jpg* can help to classify or categorized an image by keywords such as “*philosopher*”, “*colour*” and ”*photograph*”.
- Text in the title (TITLE) and alternative text (ALT) attributes of the image tag (IMG). Texts in these fields are manual annotations and are probably the most accurate annotations to the image. As was discussed in the previous chapter, manual annotation has weaknesses such as subjectivity and it is time consuming. Subjectivity and time is not a problem for a given environment - academics almost never annotate images because lack of time and/or skills.
- Image caption. More detailed description was given in the previous subsection. Some image-caption relationships are not explicit (figure 4.9).
- Near text is the ordinary text that is placed visually within the same table raw as an image. In the most cases this text is relevant either to the image content or to the subject an image is about. In cases where near text is irrelevant to an image it is not a problem because the probability of the search by keywords that is a part of the irrelevant text is almost zero.

Other components such as headings, filename-, URL- and title of the HTML document are not included in the surrounding text in this approach. This decision is based on the study of the HTML documents for this particular data-scope. It turned out that these components has little or no relevance to the images and are defined as “noise” in this approach.

4.3.1 Studies of the users and their publishing skills

To decide what textual information and how it could be related to the images used within HTML documents user studies have been performed. As was mentioned, users are academics that publish their teaching content on the web. Content is HTML files with embedded images. Based on the observation, users have different skills and understanding when it comes to design and publishing content on the web. An example of fixed structure is shown in figure 3.1, where images, image captions and near text have the fixed structure as layout, font, style and colour.

Before further discussion some definition will be given for the terms that will be used in this thesis:

- Heading is usually a short sentence that summarizes content of a paragraph that precedes an image, and it differs from the ordinary text by font size and its colour. Headings are shown at the figure 4.3.
- Image caption is a text placed above or underneath an image, but sometimes when images are big or with a combination of images, image caption could be placed to the right, or to the left, or in between of images. This text, as a rule differs by its style (“Italic”), font (font size=”1”) and sometimes colour. Examples are shown in figure 4.3
- Near text is ordinary text that is placed near an image. It can be to the right or to the left of the image. In some cases text could be placed under images, as illustrated in figure 4.3
- Surrounding text, also referred in thesis as a user context or textual information, is all text that is available within the image and assumed to be the most relevant to it. It includes image ALT and TEXT fields of the IMG tag, image file name, text available with the image URL, image caption and near text.
- Subject search is searched keywords that define specific matter, event or theme, not a specific object. For example “*Ethics in war*”, “*Cultures and conflicts*”, “*animal’s rights*” is the subject search, but “*Thinking ape*” is a search about a concrete object – ape in a thinking position.

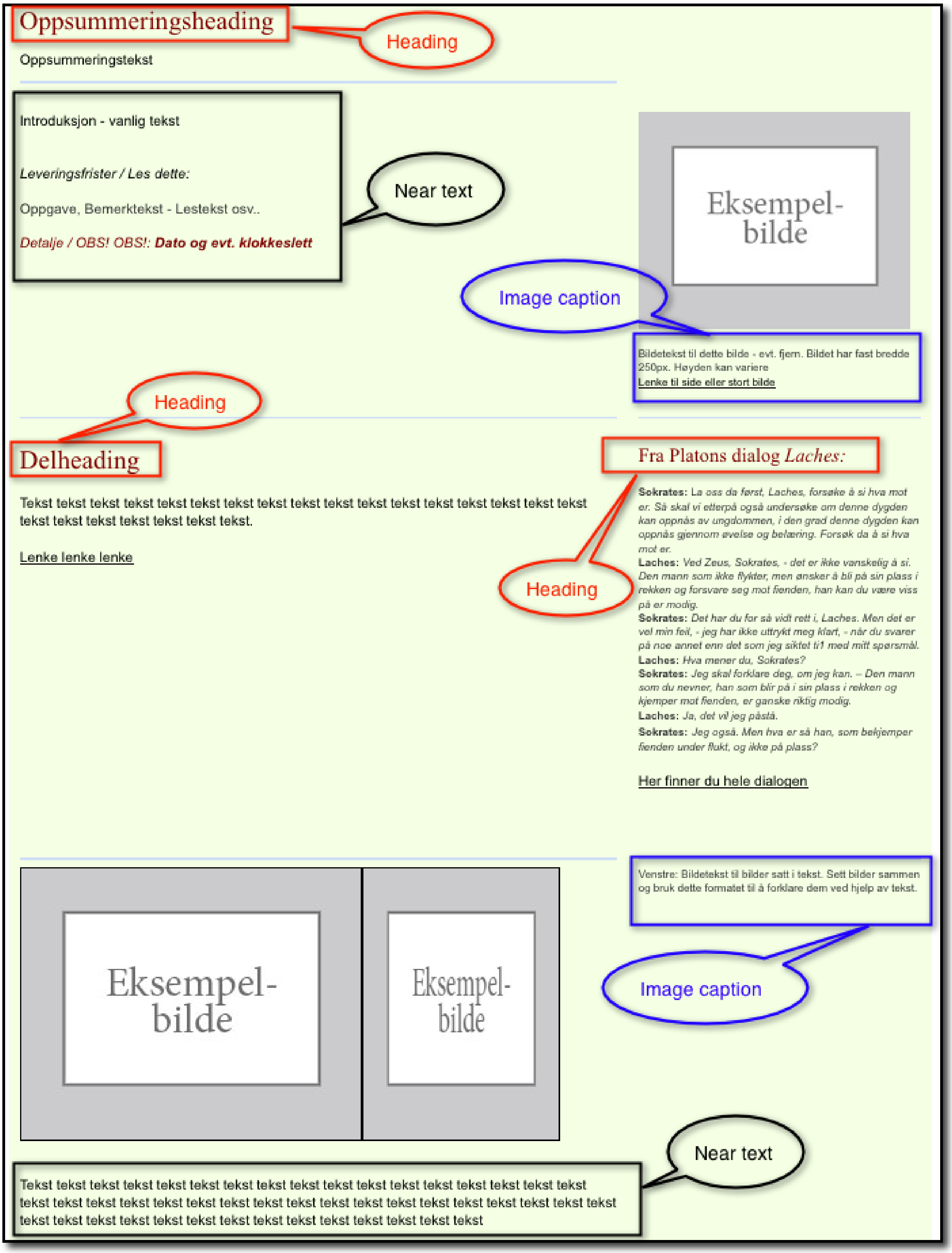

Figure 4.3: design template

Figures 4.4-4.9 illustrate how academics use images in publishing their content.

The image in figure 4.4 is used to illustrate some point at further discussion of juridical, social and moral norms. Near text has no relation to the image content, but the system will use this text to relate it to the image. On one hand it is “noise” that can lead to poor system performance, but on the other hand, the probability that user will type search keywords that match keywords from the near text is almost zero. It will result in the image will not be retrieved by the system as relevant to the search.

In figure 4.5 the image is used to illustrate the subject that is rhetoricians, and only the last paragraph of near text is related to the image. But as in the previous example, the probability that user will use search keywords from the not related text is regarded as zero.

Vi fortsetter med Johansen og Vetlesen, og skal i de to neste rundene ta for oss kapittel 8, som omhandler etisk argumentasjon. Denne uken skal vi konsentrere oss om hva et resonnement er, og begrepene premiss og konklusjon.

Les

- Johansen og Vetlesen kap 8

Levér besvarelse av oppgavene

- torsdag 1.10 kl. 22.00

Spilleregler utgjør en av mange normtyper, og fotball er verdens mest utbredte spill. Bildet er fra en Premier League-kamp høsten 2007, og viser Chelseas Didier Drogba, som forsøker å overliste Manchester Citys keeper, Joe Hart. En nødvendig betingelse for scoring er at ballen passerer målstreken. Går ballen i mål?

Figure 4.4: Text under the image, or image caption, is directly related to its visual content. Text to the left has none semantic relation to it.

Vi fortsetter med Johansen og Vetlesen, og skal i de to neste rundene ta for oss kapittel 8, som omhandler etisk argumentasjon. Denne uken skal vi konsentrere oss om hva et resonnement er, og begrepene premiss og konklusjon.

Les

- Johansen og Vetlesen kap 8

Levér besvarelse av oppgavene

- torsdag 24.9 kl. 22.00

Argumentasjonsteori har vært en sentral bestanddel i filosofien fra antikken av, og hadde en viktig plass i Aristoteles' verk *Retorikken*. Her ble det lagt vekt på at en god taler først og fremst skal basere seg på gode argumenter og konsistent argumentasjon. Men det stilles ulike krav til ulike typer argumentasjon. Innenfor etikk og politikk kan vi ikke stille de samme kravene til logisk konsistens som innenfor vitenskapene. Du kan lese mer om argumentasjon i kap. 11.

Bildet er kalt Retorikere i et vindu, *og er malt av den hollandske maleren Jan Steen på 1660-tallet.*

Figure 4.5: Text under the image, or image caption, is directly related to its visual content. Text to the left is related to the subject the image illustrates, rhetoricians, thus indirectly related to the image.

Figure 4.6 is the example of when images are used to illustrate the textual content. This means that readers can assume what subject the information is about by looking at the pictures. In this case all near text is related to the subject the images are about – different cultures and conflict. And the system will relate near text equally to both images, even if the first image is about harmony between two musicians, Shankar from India and Menuhin from Europe. And the second image is about flag burning in Pakistan in a protest against publishing drawings of Mohammed in a Swedish newspaper. If user searches for "*cultures and conflicts*" – both images will be retrieved and be relevant to the search. But if user searches for "*Flag burning in Pakistan*" or "*Shankar and Menuhin*", both images also will be retrieved, but one of them will be irrelevant to the search.

I nyheter og dagsaktuell debatt fokuseres det i stor grad på motsetninger og konflikter mellom ulike kulturer eller mellom stater med ulik kulturell tradisjon. Men vi vet selvsagt at det også foregår mange former for samarbeid mellom kulturelt sett ulike stater. Og vi vet at ulike religiøse grupper kan leve fredelig side om side innenfor samme stat eller område, slik muslimer, katolikker og ortodokse kristne gjorde i Bosnia. Men det er – kanskje forståelig nok – konfliktene som stjeler overskriftene.

De siste årenes konflikt mellom på den ene siden vestlige land og på den andre enkelte muslimske land og grupper kan i stor grad betegnes som en verdikonflikt (selv om konflikten også har politiske, sosiale og økonomiske sider. I tillegg har konflikten mellom Palestina og Israel hatt ringvirkninger). Denne verdikonflikten kommer til uttrykk på mange ulike måter, bl.a. som en konflikt mellom den politisk liberalistiske verdien ytringsfrihet og religiøse verdier, noe vi bl. a. ser gjennom saken om karikaturtegninger av profeten Muhammed, en konflikt som selvsagt ble viet stor plass også i norske aviser.

Vi bør i denne konflikten for øvrig ikke glemme at Life of Brian i 1980 ble forbudt i Norge av filmkontrollen med henvisning til blasfemiparagrafen i straffeloven. Vi bør heller ikke glemme det positive samarbeidet og den konstruktive kommuniskasjonen som foregår mellom ulike kulturer. Her er defor et eksempel på en harmonsisk dialog mellom representanter fra ulike kulturelle tradisjoner, mellom Shankar m fl, skolert i klassisk indisk musikktradisjon, og Menuhin, skolert i klassisk europeisk musikktradisjon: Shankar/Menuhin (fra CD'en West meets East): **Raga Piloo.mp3**

Foto: AFP PHOTO/Arif ALI

I august 2007 ble det bl.a. i Pakistan demonstrert mot tegninger av Muhammed. I byen Lahore ble det brent flagg, og en svensk diplomat ble innkalt til pakistansk UD for å forklare seg om saken. Kilde: Aftenposten 31.8.2007.

Spørsmålet om hvor grensene for ytringsfrihet bør gå tas opp i en tekstsamling (Frihet og toleranse), som er laget i tilknytning ett av de foreliggende forslagene til hjemmeoppgave. Du finner mer informasjon om disse forslagene i orienteringsheftet.

Figure 4.6: In this example images are used to illustrate the subject of the text to the left.

Figure 4.7 exemplifies when an image is used as a symbol to the subject that is about writing an assay in the ethic issue in the philosophy discipline. It is difficult to say what subject aspects this ape illustrates, this is up to the reader. As a human we see an ape on the image and it seems to pretend to think. But the system will relate all near text to the image that in this case is irrelevant to the image content. The question here is how users/academics will search for this image. If the image is annotated manually that is TITLE and ALT fields of the IMG tag, for example by keywords such as "*Thinking ape*", there is not a problem to retrieve the image as relevant to its content. But in the worst case, when the image is not manually annotated, image filename has no relation to its content, for example *P02500.JPG*, URL of the image does not give a clue of the image category or classification, and there is no caption - search for this image can end up with no results. On the other hand it will depend how users will search for this image. If the user knows that he or she published this image as a part of the context about writhing an assay, probable search phrase will contain keywords such as “*assignment, assay*”. If not – there is no possibility to find the image with keywords that should be relevant to the visual content.

Den neste uka skal dere arbeide med pensumoppgaven i etikk. Oppgaven er andre innlevering til eksamensmappa.

Generelt om eksamensopplegget (fra orienterinsheftet): Eksamen består av tre skriftlige innleveringer: en oppgave i filosofihistorie og en i etikk, samt en semesteroppgave, som er et litt lengre arbeid. Begge de to første oppgavene skrives ut fra oppgitte problemstillinger, der du velger mellom to alternativer. Hjemmeoppgaven velger du selv problemstilling til, enten ut fra et av de fem semesteroppgaveemnene eller ut fra pensum. Hjemmeoppgaven og de to mindre oppgavene blir vurdert under ett av en eksamenskommisjon med to sensorer. Det gis bokstavkarakterene A – E for bestått og F for ikke bestått. Det vises ellers til orienteringsheftet.

Pensumsoppgave i etikk

Pensumoppgaven i etikk inngår i eksamensmappen. Oppgaven kan leveres som en individuell oppgave eller som en gruppeoppgave (inntil 3 personer per gruppe). Den skal baseres på en av de diskusjonene dere alt har gjennomført. Dere kan bruke både egne momenter og det dere måtte ha lært fra innlegg som andre har hatt i diskusjonsforum. Dere kan revidere og reformulere oppfatninger, og føye til nye poenger. Dere skal generelt forsøke å videreutvikle det som alt er gjort. Oppgaven kan leveres enten i dialogform eller som en drøftende sakprosatekst. Enten dere lager en dialog eller en sakprosatekst skal dere forsøke å få til en drøfting av ulike synspunkter. Hvis dere velger dialogform må dere konstruere rollefigurer som forsvarer en posisjon (en proponent) og som argumenterer mot posisjonen (en opponent). Det er nok lettest å holde oversikten hvis dere ikke innfører for mange stemmer, og to er altså tilstrekkelig. Det viktigste - uansett hvilken litterær form dere velger - er at dere viser at dere kan forholde dere til ulike argumenter, at dere kan vurdere argumentene, sette frem motargumenter etc.

Oppgavens lengde: Ca tre sider er passe lengde for en som skriver alene, legg på en side for hver av de eventuelt andre som inngår i gruppa. Man bør bruke marg på ca 2.5 cm, og linjeavstand 1 ½, skriftstørrelse 12. Sidene bør ha sidetall. Husk å skrive navn på besvarelsen.

Levér besvarelsen innen torsdag d. 29. oktober kl. 22.00

Det gis to oppgavealternativer. Én av oppgavene skal besvares. Alternativ 1 baserer seg på den diskusjonsoppgaven som ble gitt i steg 2 (om kulturrelativisme). Alternativ 2 baseres på den diskusjonsoppgaven som ble gitt i steg 4 (om tortur kan forsvares moralsk).

Figure 4.7: This image does not have any captions describing its content, and near text is about writing an assay in ethics in a philosophy discipline.

Figure 4.8 is a typical example of inconsistency between textual and visual content. Near text of the top image has few keywords that might be relevant, and the rest is the “noise”. But yet it is not a problem because of search probability for irrelevant keywords. And because of the image caption for the top image will be annotated by the relevant keywords anyway. The situation with the near text for the bottom image is much worse. The first two paragraphs of that text are about a subject that is covered by the top image – a drawing of anthropologists – cultural relativism and multiculturalism. The last paragraph is related to the bottom image – the great Greek historian Hesiod. As a human we can see what text is related to what image. But the system will consider all near text of the bottom image as relevant to it and not to the top image. If user searches for “*Cultural relativism*” both images will be returned as relevant to the search, but only the image with anthropologists is relevant. If user will search for “Hesiod”, only image of Hesiod will be returned by the system that is correct.

Last image, figure 4.9, illustrates inconsistency of caption. Text marked by red should be image caption and placed above or under the image. The system will capture the text anyway, but relate it to the image as a near text, not as an image caption. This could be a problem for systems like MARIE-3 [16] that is based on finding captions, or other systems that consider certain parts of textual information differently, for example captions are more weighted because it is assumed to be more relevant to an image then near text.

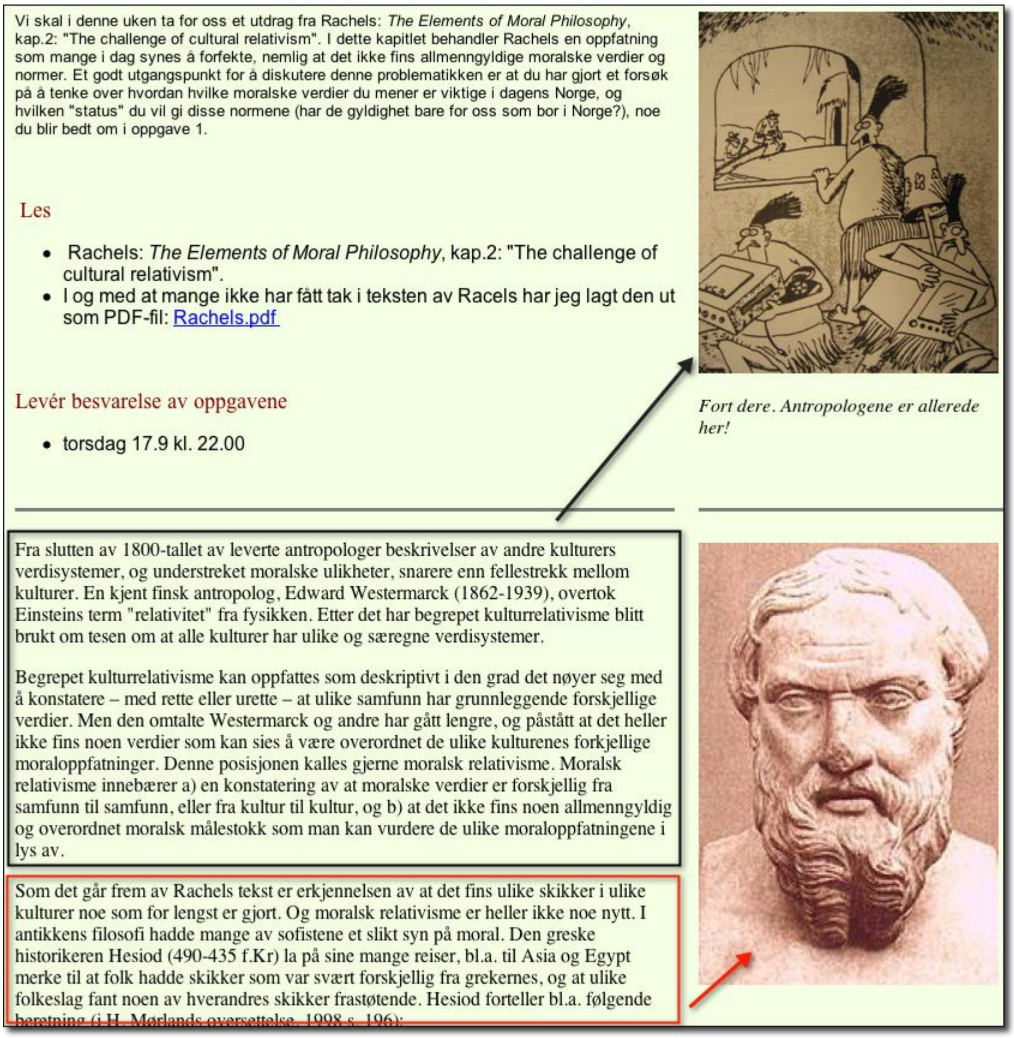

Figure 4.8: This image illustrates inconsistencies of near text. Text marked by black is related to the subject/event of the top image, while text to the left has no relation to it at all. Text marked by red is related to the image at the bottom. It is difficult or near impossible for the system/computer to see the inconsistency.

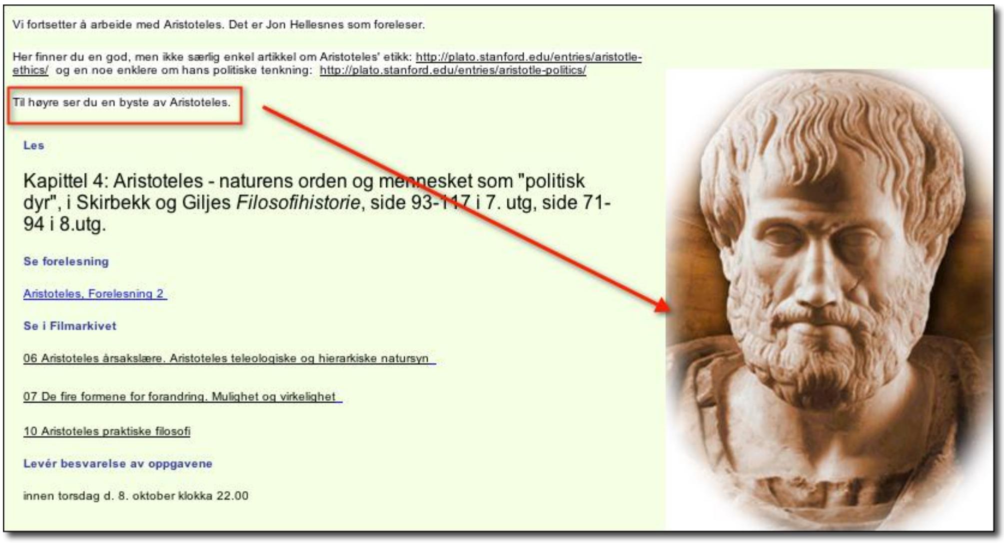

Figure 4.9 illustrates the case where image caption is not obvious. The region in the red should be placed over or under the image and differs from other text by its style and font.

All these examples show that user skills of understanding design and publishing on the web is very important. Inconsistency between text and usage of images can lead to poor image retrieval. The approach in this thesis is to use textual information to learn about the images visual content. The question is how to deal with inconsistency of the near text? On one hand, if an image has no manual annotations, filename and URL do not give a clue of the images visual content or images subject then near text could be an important provider of information about the image content and its semantics. On the other hand, near text could be also “noise”. One technique is to consider near text as less relevant to an image as for example ALT and TITLE fields of the IMG tag. This approach will give poor results in a given data collection, because in average 10 of 200 images that are uploaded to the archive are manually annotated. Rejection of the near text as a source of image description, if image caption is present, could give better results in a given data collection. Advantage of this technique is also that a lot of irrelevant text will be excluded from annotating the image and it will also result in a smaller index database. On the other side, the chance that a user will search for irrelevant keywords is almost zero (figure 4.4 and 4.5). In cases where the image has no other textual information then near text (figures 4.6 and 4.7) this technique will not help.

4.4 Term weight model

To begin with, each HTML document that contains image or images is parsed for the context. The scope of the context is defined in the previous section. It is obvious that words appearing with the specific HTML tags are more important to the image content then others. In given context scope the list of the HTML tags could be ranked in a following order, from the most to the less relevant:

- ALT field of the IMG tag
- TEXT field of the IMG tag
- Image caption is the text inside the ** tag
- Image filename extracted from the file URL
- Words extracted from the image URL
- Near text that is text inside the parent TR tag.

One of the approaches here could be assigning different weights to the words appearing in the fields according to the likelihood of useful information that may be contained by the text. This term weight model is used by the systems such as WebSEER[13], Diogenes[14] and Google. But this is not the approach for this thesis. The decision is based on the observation that, unfortunately, very few IMG tags contain text in the ALT and TITLE fields. Image captions are in some cases absent (figure 4.7) or are not obvious (figure 4.9). After studying what information was extracted for the images in the data collection I saw that in almost all documents image captions end outside of the ** tag. When it comes to the image filenames and URLs, in some cases image filenames such as “*untitiled.JPG*” and words extracted from the URL such as “*http://destore…./Undervisning/untitled.JPG*” does not give any clue about image content. The approach of rejecting words from near text when present words inside ** tag is not an issue of this thesis either, but this method is easy to implement and integrate in a new version of the system for further testing.

Next, images are represented by a set of keywords extracted from the representative tags. Normalization techniques such as stop-words removal and stemming are applied. These techniques were described in section 2.6.

After normalization of words, term frequency and inverse document frequency are computed. Term frequency is occurrence of the term *j* for the image *m* and it captures the premise that if a rare word appears frequently in a context related to an image then it is very significant to it. On the other hand, if a common word appears frequently, it may not be as significant. (See section 2.6 for more information about inverse term frequency). Removing high-frequency words will help to avoid such situations. In any case, the higher the term frequency is – the higher the significance of that term is to an image. The principle of the inverse term frequency is to allocate term importance, which is inversely proportional to the total number of images that this term is related to.

Idf (inverse document frequency) value for each term in the collection is stored to its own table, *VAL*, and is used to compute term weight for the given image or query.

Next a term and image matrix *TID* is constructed where each row corresponds to the term, and each column corresponds to the image. Value of each cell tid_{mj} represents a frequency of term *j* related to an image *m*.

Term frequency and inverse term frequency is combined to generate a weight for each unique term *j* that is related to each image *m*. This weight model, also known as *tf-idf weighting,* and described in section 2.6, takes into account the importance of the term that is related to an image and also to the whole collection.

4.5 Measure model

The query process is to compute the similarity between query terms and terms related to an image, $SIM(Image_m, Query_q)$. These are measure models or techniques that have been successfully applied to text-based image retrieval.

In this thesis simple dot-product similarity measurement is used to find images that might be relevant to a query. An overview of the model was given in the section 2.6. In this model query terms follow the same normalizations procedure as image terms, and query term weights are computed. Next, term weights for an image are multiplied by the query term weights and the result is used to produce the image relevance score.

The advantage of this similarity measurements model is that it is easy to implement, it allows retrieval of images that approximately match the query and it is easy to rank images according to their degree of similarity. But this model does not capture and resolve dependencies between terms, e.g., “*be or not to be*”, or “*animals rights*”. Nor does this model use list of synonyms to resolve semantic similarity between query terms and terms related to an image, as it done by the LSI model or by adoption of thesauri or dictionary lists.

One of alternative measuring models could be usage of machine-readable thesauri or dictionaries such as WordNet that has been proposed in [11]. It connects query terms with annotation terms semantically rather than lexically. This measurement technique might be efficient if a number of terms related to an image and queries are limited, or with poorly annotated images. In a given data collection there is a gap between poorly annotated images, i.e. images that are only annotated by filename and words extracted from the image URL, and in worst case scenario where the whole text of the HTML is related to an image. Adoption of thesauri or dictionary in this case could be applied to the images that are poorly annotated, for example fewer than 5 words, but this is not implemented and tested in this version of the system.

The use of LSI (Latent Semantic Indexing) is another technique to overcome problems such as *polysemy* and *synonymy* that was described in section 2.4. LSI is the method that uses statistical methods to find similar words to the document's terms. This term-document probabilistic relation can be formulated by using a vector space model, where each image with relation to it words/terms is represented as a vector, and where each vector component will represent a particular term associated with the given image. The similarity measure between query terms and image terms is done by calculating the cosine of the angle between the query and the image in the vector space [9, 15, 19]. LSI technique has been successfully applied to many text-based approaches of image retrieval, but is not considered in this thesis.

Chapter 5

Design

5.1 Introduction

This chapter will give an overview of the design and architecture of the prototype called ImSE, an image search engine, which automatically annotates images and supports image search in DeStore based on these annotations.

The implementation of ImSE is based on the DeStore as the storage for uploaded and created files in Fronter, i.e. all image and HTML files stored in Fronter Archive for philosophy disciplines are copied to DeStore. The goal of ImSE is to find relevant images on DeStore for textual queries.

The system automatically annotates images with the keywords extracted from the available surrounding text, also referred to as context. It includes image filenames and URLs, text in the TITLE and ALT fields of the IMG tags, image captions and text that is near to the image (see chapter 4.3 for more detailed description). If an image is used within different contexts, it will be annotated by the keywords gathered from the different contexts. The keywords are assigned weights according to their importance for the image and indexed for the search. Indexed keywords are referred to as terms.

During image retrieval relevant images are founded by comparing query terms and indexed terms for an image. Images that contain the most similar terms to the query will be returned to the user as relevant.

5.2 Fronter HTML editor

This section describes Fronter HTML editor that is specific in its way to create an HTML content both on user- and application levels. For parsing and extracting desired text one has to understand how HTML structure in Fronter is built, and created by the user content.

The Fronter HTML editor is the third part application, CKEditor[13], which is tightly integrated into the VLE system. This integration is very specific, and an HTML document in Fronter is

[13] http://ckeditor.com/

called a page. On the user level the HTML editor is a container that can be filled out with different types of content, as it illustrated in figure 5.1.

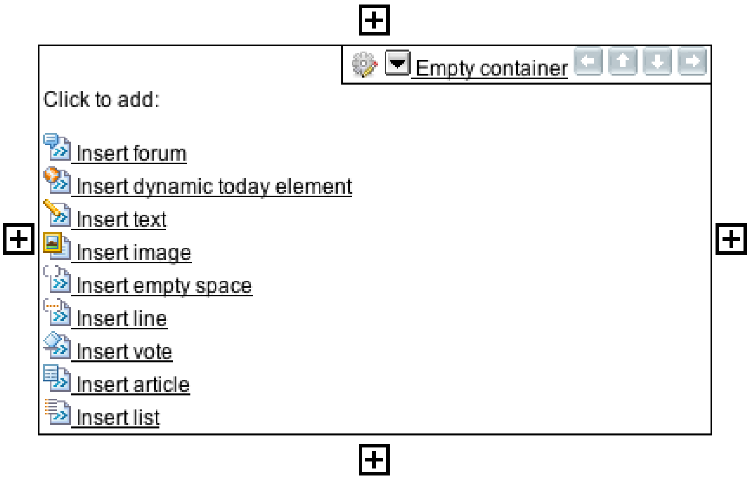

Figure 5.1: User interface of the Page-tool in Fronter

At the application level, each container corresponds to a HTML table in the table structure. The content of each table in table corresponds to the content inserted by the user. Most often – used content is text and images. The HTML structure generated at the application level for an empty container (figure 5.1) is shown in figure 5.2.

Content to be inserted

```
<TABLE width='100%' summary='Display table'>
   <TBODY>
      <tr>
         <td id="some predefined attributes and values in Fronter editor">
            <table width=100% style='height:100%;'  summary='Layout table'>
               <tbody>
                  <tr>
                     <td id="some predefined attributes and values in Fronter editor">
                        <p> </p>
                     </td>
                  </tr>
               </tbody>
            </table>
         </td>
      </tr>
   </TBODY>
</TABLE>
```

Figure 5.2: HTML structure generated for an empty container

If a user clicks on the "*insert text*" in the empty container, writes some text and saves it, the system will automatically place the text inside of the inner table, for example within the *<p></p>* tags as it marked with the red arrow in figure 5.2.

The user can add new content containers (cells) to the page by clicking on the "*plus*" signs (see figure 5.3) which then creates a new table in table structure that has either a column or row to the already existing cell, depending on "*plus*" position the user clicked on. Figure 5.4 illustrates HTML code generated at the application level for the page illustrated in figure 5.3.

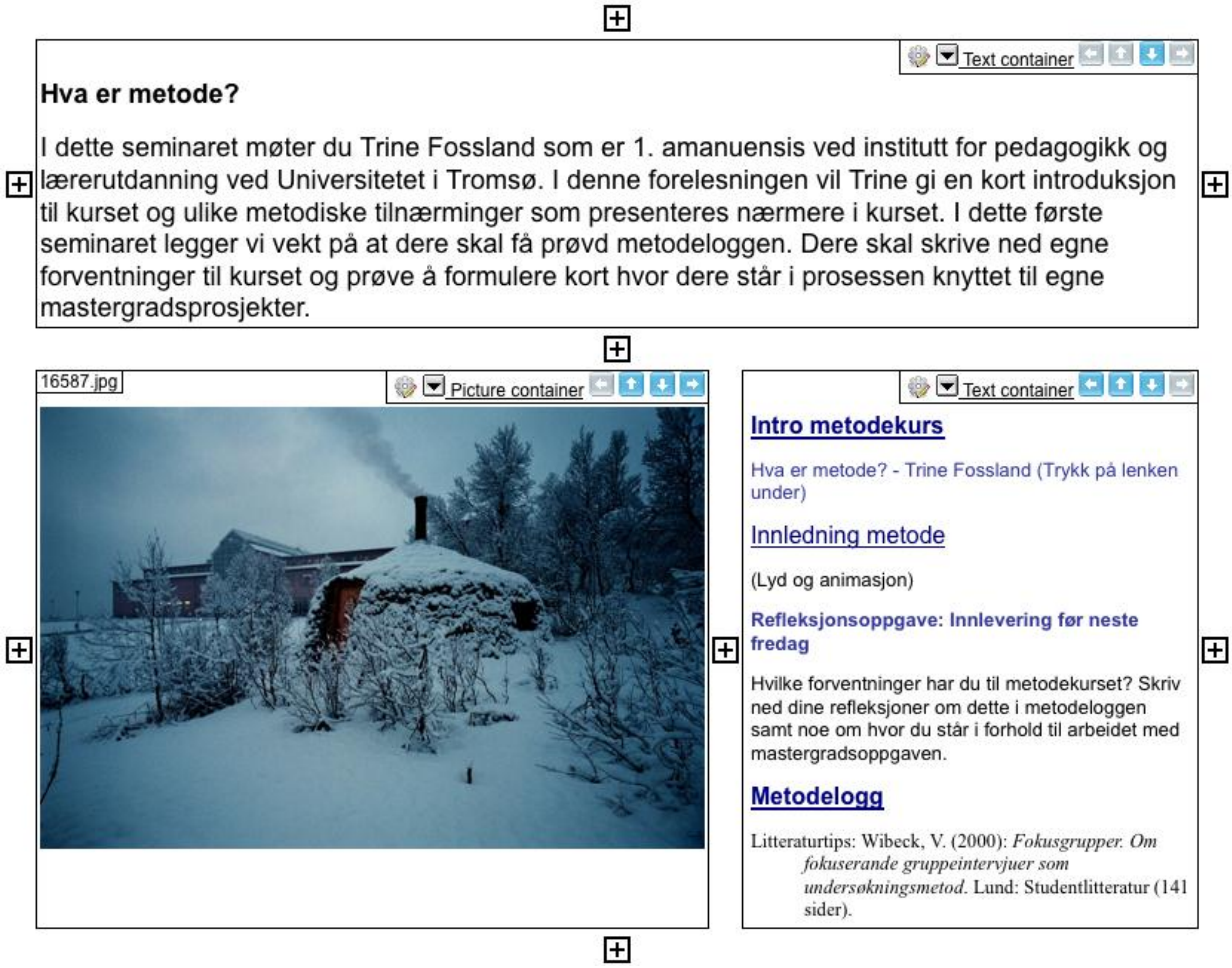

Figure 5.3: Example: Page generated by a user in Fronter (VPMA course at the University of Tromsø).

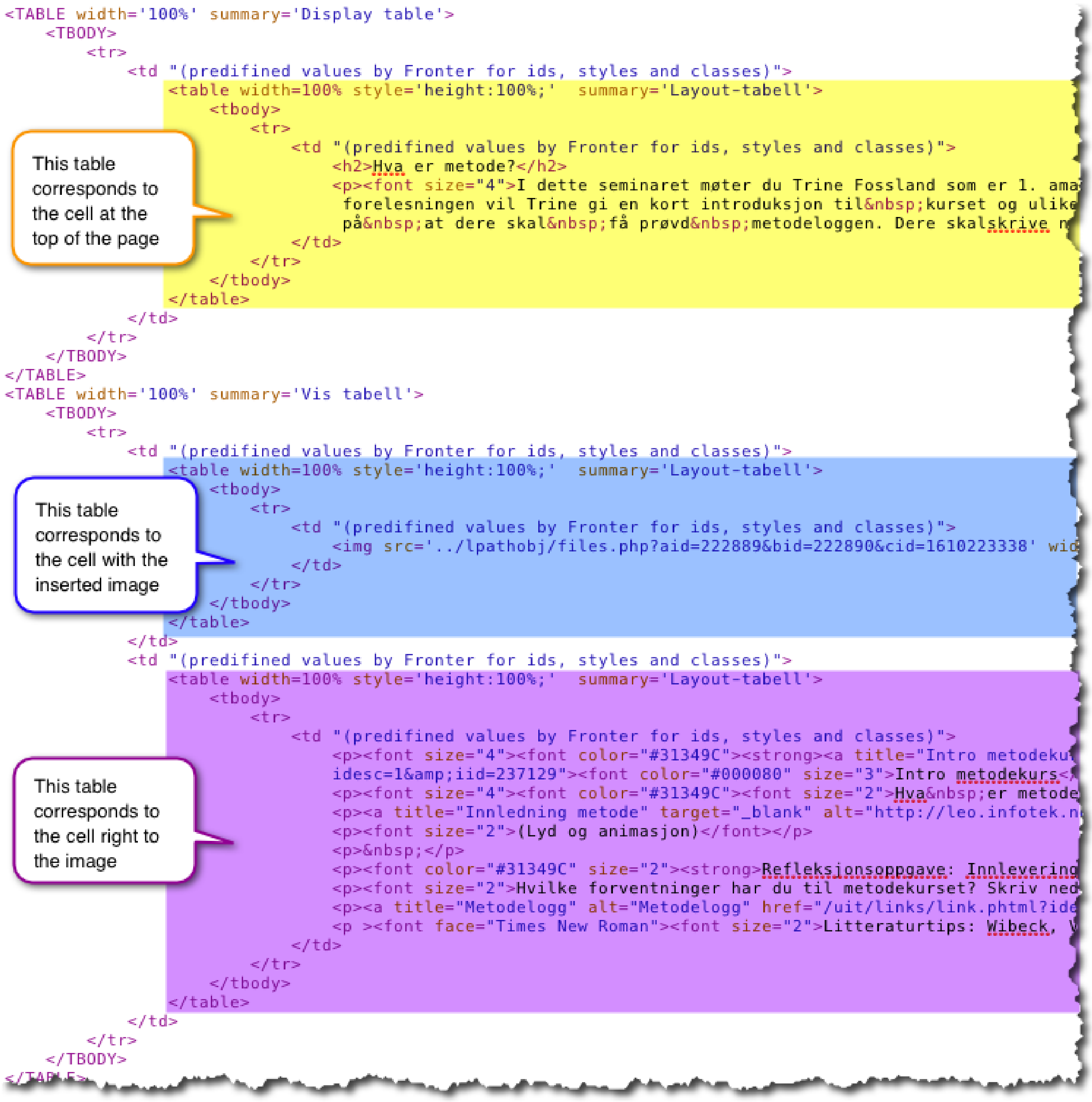

Figure 5.4: HTML code generated at the application level for the page illustrated in figure 5.3

If this document was created with other editors and published to other web sites or on-line systems, the HTML structure could be something other than tables.

So, if a page in Fronter is created by using many "*pluses*", the HTML structure in the document will end up with a long and complicated table in table in table structure. To avoid such publishing complexity for the philosophy disciplines, and make publishing of the subject content easier and better structured, content designers made a *CSS* (Cascading Style Sheets[14]) template with predefined tables and cells, and with page specific classes for describing content and layout. On the user level the *CSS* template is inserted into the first text container – therefore all content is published in only one text container as it illustrated in figure 5.5. Typical HTML

[14] http://www.w3.org/Style/CSS/

structure generated at the application level for the Fronter page with the inserted template is illustrated in figure 5.6.

Figure 5.5: Example of the page with the inserted template for the ethical issues in the philosophy discipline

```
<TABLE width='100%' summary='Display table'>
   <TBODY>
      <tr>
         <td id="some predefined attributes and values in Fronter editor">
            <table width=100% style='height:100%;'  summary='Layout table'>
               <tbody>
                  <tr>
                     <td id="some predefined attributes and values in Fronter editor">

                        <table> Template </table>

                     </td>
                  </tr>
               </tbody>
            </table>
         </td>
      </tr>
   </TBODY>
</TABLE>
```

Figure 5.6: HTML structure in Fronter with inserted template

Template contains specified classes with predefined styles, colours, fonts and sizes for publishing text according to if it is a heading, an image caption, ordinary text, quotation, etc. Chapter 4.3 identifies and describes different context elements that might be relevant to an image and thus are of our interest. It includes image filenames, URLs, captions, text in the ALT and TITLE fields of the IMG tag and near text. Near text, headings and captions might be different in their styles, colour and size (illustrated in figure 4.3 chapter 4.3) and belongs to the different *CSS* classes respectively to *"vanligtxt"*, *"subHeader"* or *"colsubHeader"*, and *"bildetekst"*. Figure 5.7 illustrates HTML structure of the designed template shown in figure 5.5. The predefined *CSS* classes are marked with red:

```
<table width="850" cellspacing="0" cellpadding="0" border="0">
   <tbody>
      <tr bgcolor="#d5edb3"></tr>
      <tr>
         <td width="850" bgcolor="#811622" colspan="2">
            <div><font size="2"> <br />
            </font></div>
         </td>
      </tr>
      <tr>
         <td width="850" bgcolor="#f4ffe4" colspan="2">
            <table width="100%" cellspacing="0" cellpadding="10" border="0">
               <tbody>
                  <tr>
                      <td width="550" style="color: rgb(128, 0, 0); font-family: Times New Roma
                        <font size="5">Innledning<br /></font>
                        <p><font size="2" style="color: rgb(0, 0, 0); font-family: Arial;">
                           <span class="vanligtxt">I denne delen av kurset skal vi ta for oss e
                           del grunnleggende problemstillinger innenfor etikk og politisk tenki
                           </span></font>
                        </p>
                        <hr size="1" noshade="noshade" style="height: 2px;" />
                     </td>
                     <td valign="top">
                        <div><font size="2"> </font></div>
                     </td>
                  </tr>
                  <tr>
                     <td width="550" valign="top" style="font-family: Arial;" class="tdmain">
                     Hvordan kan vi begrunne våre moralske standpunkter? Hva vil det si å argu
                     etisk? Fins det en objektiv etisk målestokk som kan si oss om visse verdi
                     bedre eller "sannere" enn andre? Fins det universelle rettigheter, dvs re
                     gyldighet for alle, uavhegig nasjonalitet, klasse, kjønn etc? Fins det uni
                     oppskrift for hva en rettferdig stat er, dvs. en målestokk som har gyldig
                     nasjoner? Dette er noen av de problemstillingene vi skal ta for oss i løpe
                        <p>Kurset skal både gi en innføring i sentrale etiske begreper og teori
                        begrepene og teoriene kan anvendes i praksis.</p>
                     </td>
                     <td valign="top" colspan="3" class="tdcol">
                        <div><a target="_blank" href="/uit/links/link.phtml?idesc=1&amp;iid=454
                        <img width="250" border="0" alt="" src="/uit/links/link.phtml?idesc=1&a
                        </a></div>
                        <div><font size="2"><span class="bildetekst">
                        <br style="color: rgb(51, 51, 51);" />
                        <em><font size="1">Smakssansen</font></em><font size="1">, 1618 er malt
                        av Jan Brueghel den eldre (1568-1625).</font> </span></font></div>
                     </td>
                  </tr>
               </tbody>
            </table>
         </td>
      </tr>
   </tbody>
</table>
```

Figure 5.7: HTML code for the *Template* with designed classes for the philosophy disciplines

5.3 Extracting information from HTML

In chapters 4.3-4.4 we were discussing the context an image is used within, what parts of the context that might be related to the image content, and what HTML tags correspond to the relevant parts of the context.

Filename of the image and keywords form the image URL are obtained by the system from the file URL while parsing DeStore for the images.

Information extracted from ALT and TITLE fields of the IMG tag corresponds to the text extracted from the fields of the *<img alt="alternative text" title="title of the image">* tag. Value of the *alt* contains alternative text. For example if the image name is P025.JPG, value of the ALT attribute might be "*This images illustrates Agora in Socrates lifetime*". Value of the *title* contains title, and it doesn't have to be the same as the filename of the image. In our example it might be "*Old Agora*".

Image caption is extracted from the text inside the ** tag. This tag belongs to the *CSS* template and was designed to be helpful for publishing contextual texts with the image with predefined location, font and size. This tag should be assigned to the image text/comments (image caption). There is no corresponding HTML standard tag that corresponds to it in the HTML version 4.4.

Near text is the textual content placed visually within the same table row as IMG tag. It corresponds to the all text inside parent *<tr>…</tr>* tags. Discussion on how relevant and useful near text might be to an image is given in the chapter 4.3. Figure 5.5 exemplifiers when near text has no relevance to the image.

What context that is considered as relevant to an image and corresponding to its HTML tags is summarized in table 5.1:

Context considered as relevant to an image	**HTML tags considered with the context**
Image filename	Extracted from the URL of the *scr* value of the *<img scr="URL">* tag
Keywords from the image URL	Extracted from the file URL
Alternative text	Text extracted from the *alt* attribute of the *<img alt="Alternative text">* tag
Image title	Value extracted from the *title* attribute of the *<img title="Image title">* tag
Image caption	Extracted text inside the *…* tag
Near text	Extracted text inside the parent *<tr>…</tr>* tag

Table 5.1: List of the context considered as relevant to an image with corresponding HTML tags.

Figure 5.8 illustrates what text is extracted from the respective HTML tags for the image in figure 5.5.

Figure 5.8: Shows what text is extracted to annotate image illustrated in figure 5.5

5.4 Architecture

This section gives an overview of the ImSE architecture; workflow and assumptions will be represented and discussed.

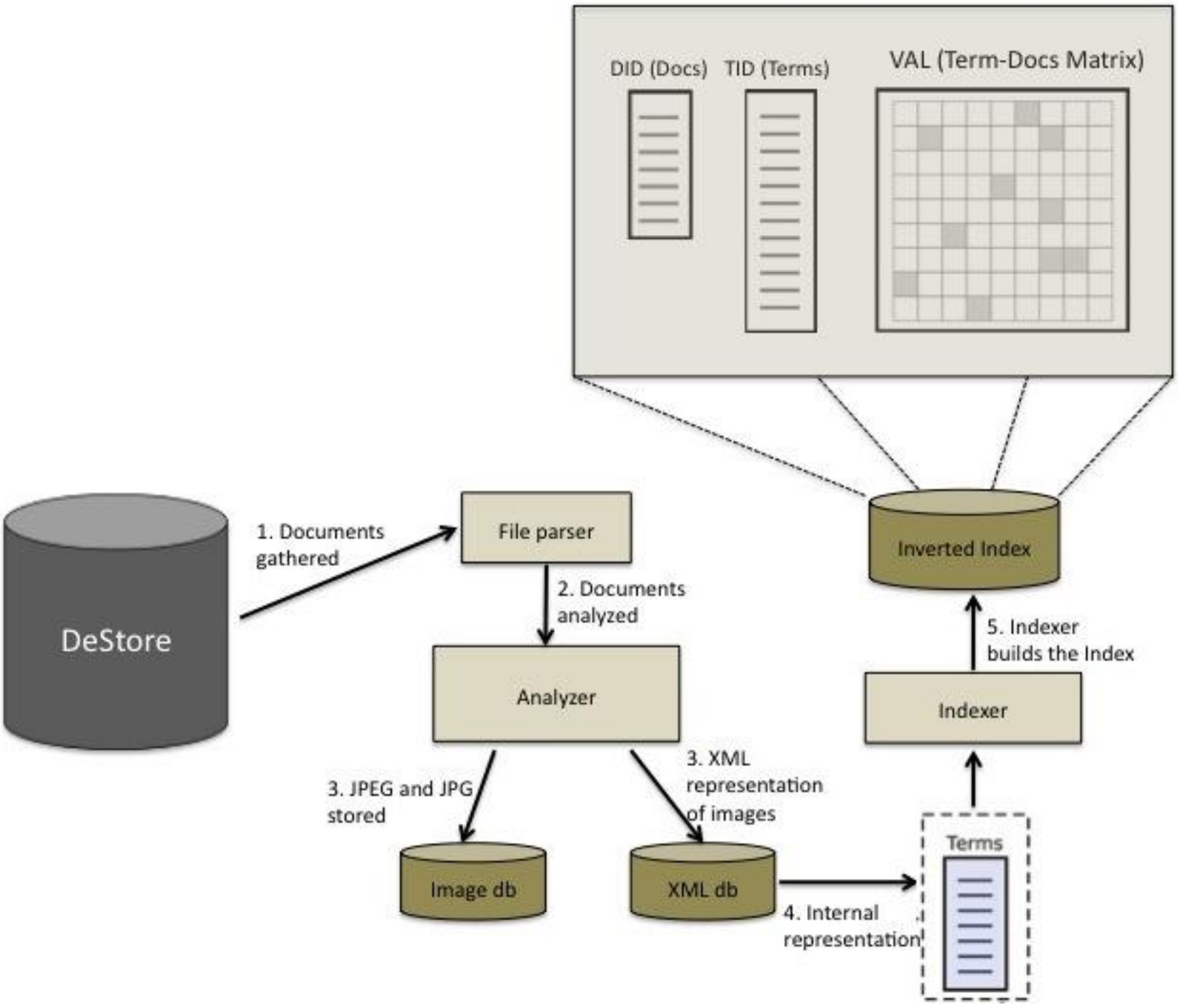

Figure 5.9: ImSE architecture

5.4.1 DeStore

As was mentioned in chapter 3.3 - DeStore is a decentralized storage. All image files and HTML files created in Fronter Archive for philosophy disciplines are copied to DeStore.

5.4.2 File parser

Firstly, file parser searches DeStore for all JPEG and JPG images and sends them to the analyzer. Secondly, it searches DeStore for all HTML files and sends them to *Analyzer* for further processing. The flow illustrated as point 1 and 2 in figure 5.9.

5.4.3 Analyzer

The analyzer component is divided into two parts: the image and HTML handlers.

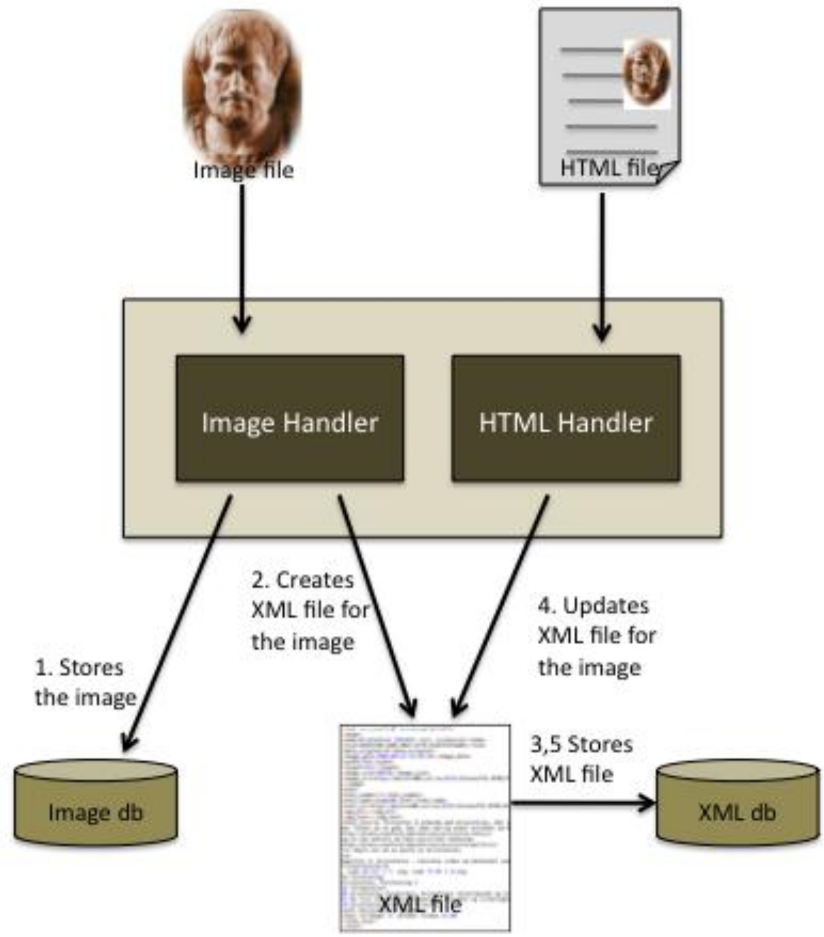

Figure 5.10: Analyzer components

Image handler copies all JPG and JPEG images into its own image database (illustrated as point 1 in figure 5.10). At the same time it creates an XML file for every image and stores the image meta-data information in it (illustrated as point 2 in figure 5.10).

Table 5.2 shows the XML tags that are created for the image meta-data.

Image Handler does not check if an image might be a copy or a crop of another image. It treats each image uniquely as long as the filenames are unique. Copies and crops of images are handled manually in the system, by adding extra XML tags to the image XML file: <*copy_of_image*> and <*crop_of_image*> that contains URL to a copy or crop of the image. Discussion of image copies and crops is given in chapter 5.5.

Meta-data XML tag	Value
<name><name>	Filename of the image
<width></width>	Width of the image
<length></length>	Length of the image
<image_size></image_size>	Size of the image
<image_src></image_src>	Image URL
<copy_of_image></copy_of_image>	URL to the copy of the image
<crop_of_image><crop_of_image>	URL to the crop of the image

Table 5.2: The XML tags created for the image meta-data

When all images parsed from DeStore are copied into the image database, *HTML handler* looks at the HTML document itself. If the HTML file contains reference to an image it will parse through the HTML tags that are considered to be relevant to the image (see table 5.1 for the considered tags), and update the XML file created for the image with the extracted information from HTML (illustrated as point 4 in figure 5.10).

Table 5.3 shows the XML tags that are created for the text extracted from HTML files where the image is referring.

XML tag considered with the context	Information in HTML file
<img_alt></img_alt>	Text extracted from the ALT field of the IMG tag
<img_title></img_title>	Text extracted from the TITLE field of the IMG tag
<img_text></img_text>	Extracted text inside the **
<html_text></html_text>	Near text extracted inside parent TR tag to the IMG tag

Table 5.3: The XML tags created for the context

Other XML tags that are created by *HTML handler* are:

- *<html_number></html_number>* : HTML file counter
- *<html_name></html_name>*: contains HTML name obtained from the file URL
- *<html_url></html_url>*: contains URL to the HTML file

If an image is referred to in several HTML files, the image XML file will be updated with XML tags created for every HTML where it is referred to and increase the value of the *<html_number>* by the following number of HTMLs. Figure 5.11 illustrates the XML structure for an image that is referred to in more than one HTML file. The <*html_name*>and <*html_url*> tags are considered as meta-data for HTML file and not used for image annotation in this version of the prototype.

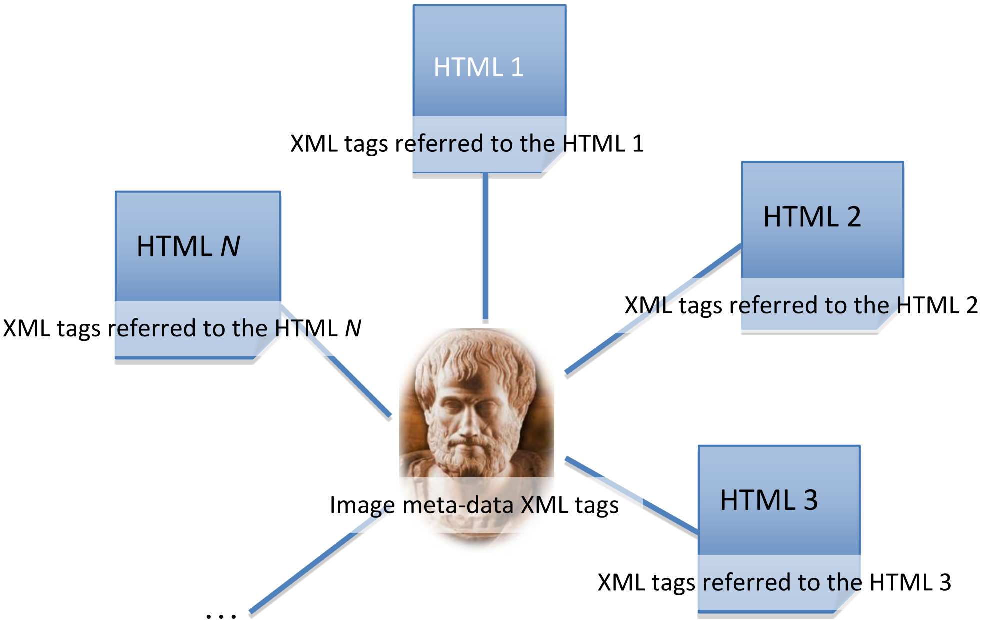

Figure 5.11: The figure illustrates the structure of an XML file for an image that is referred to in *N* HTML documents.

5.4.4 Image db

The image database stores JPEG and JPF image files. When processing queries, thumbnails of the images that the system finds relevant to a query are retrieved from the database and represented to the user with a reference to the image location on DeStore location. If an image is not relevant to the query, it will spare DeStore from having to retrieve the image.

5.4.5 XML db

The XML database stores XML files created for every image in *Image db*. Information stored in the XML files is used by *Indexer* to annotate images.

5.4.6 Indexer

Indexer parses XML files stored in the XML database and extracts information inside of the meta-data and context XML tags (see tables 5.2 and 5.3 for the XML tags and what information they correspond to). Extracted values for the image meta-data are stored in the own file called DID in *Inverted Index* (referred to as point 5 in figure 5.9.). Extracted keywords inside the <*name*>, <*image_src*> and context XML tags are translated to the internal presentation, i.e. normalized by removal of stop-words and stemmed (referred to as point 4 in figure 5.9). Then *Indexer* assigns weights to the terms, and stores them in the own files called TID and VAL in *Inverted Index* (referred to as point 5 in figure 5.9.). Next section describes DID, TID and VAL files in more detail.

5.4.7 Inverted index

Inverted index database contains three files: DID (**D**ocument **i**n**d**ex file), TID (**T**erm **i**n**d**ex file) and VAL (**Val**ue index file) for image indexing as illustrated in figure 5.12.

The DID file creates one entry per indexed image. Each entry contains the following information extracted from the corresponding XML tags for the image:

- A unique ID is assigned to the image and referred to as document ID,
- Image width,
- Image length,
- Image size,
- Image URL,
- URL to the image copies and
- URL to the image crops

The TID file is an array of terms and document IDs that stores term frequency value for an image. Terms obtained from the extracted keywords inside of the following XML tags for the image:

- *<name></name>*
- *<image_src></image_src>*
- *<img_alt></img_alt>*
- *<img_title></img_title>*
- *<img_text></img_text>*
- *<html_text></html_text>*

Extracted keywords are normalized by removing stop-words and stemmed by the *Porter* algorithm. Term frequency value (*tf*) is obtained by counting the occurrence of that term for the image.

The VAL file stores value of the computed *idf* weight for the term (chapter 2.6 gives more detailed description of the *idf*). This value is used to calculate the weight for the term in the document. The file structure of the inverted index is illustrated in figure 5.12.

Thus *Inverted Index* indexes and stores terms the images are annotated with.

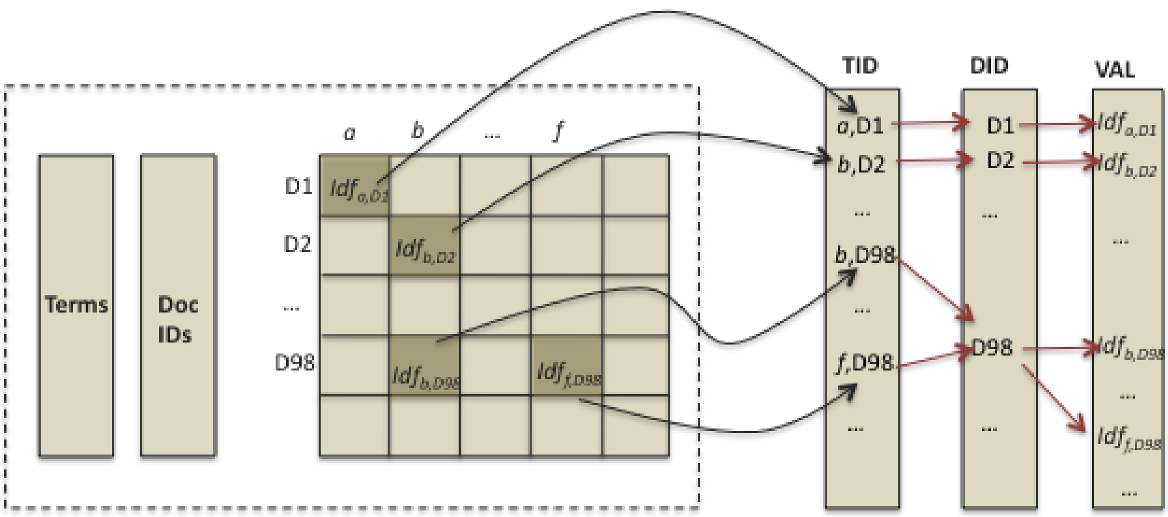

Figure 5.12: Inverted index file structure

5.5 Image copies and crops

The current implementation of the ImSE has no mechanism that is able to see if an image might be a copy or a crop of another. Thus copies and crops of images are handled manually by updating image XML file with extra XML tags *<copy_of_image></copy_of_image>* and *<crop_of_image></crop_of_image>*. The value of tags is URL to the copy or a crop of the image. This conclusion is based on the observation that users tend to make several copies of images but they might differ in size, location or name or a combination of the above. Users might also be interested in a specific area of an image and thus creates a crop.

As mentioned earlier, the main focus of the ImSE is to retrieve all relevant images to a query including copies and crops. Consider the situation where an image has 4 copies that are unique in name, size and location, but only one of the copies is used in the document and thus annotated by terms that hits the terms of the query. As the system treats all images as unique objects, only the annotated copy will be retrieved as relevant to the query, then 4 others will remain in the not retrieved set. Set from the user perspective, it is not a problem as long as the retrieved image has a usable size. Seen from the systems perspective – it will reduce the retrieval performance. The same argumentation is applied to the image crops.

In this version of the system copies and crops are recognized and handled manually and used only for the retrieval performance measurement. But it is also possible to take advantage of copies and crops for searchers as well, for example retrieving only originals. It is often images with the best resolution and, as a rule, contains much more pixel information.

5.6 Processing a query

This section will give an overview of how the system handles queries and generate a retrieval set with images to the user.

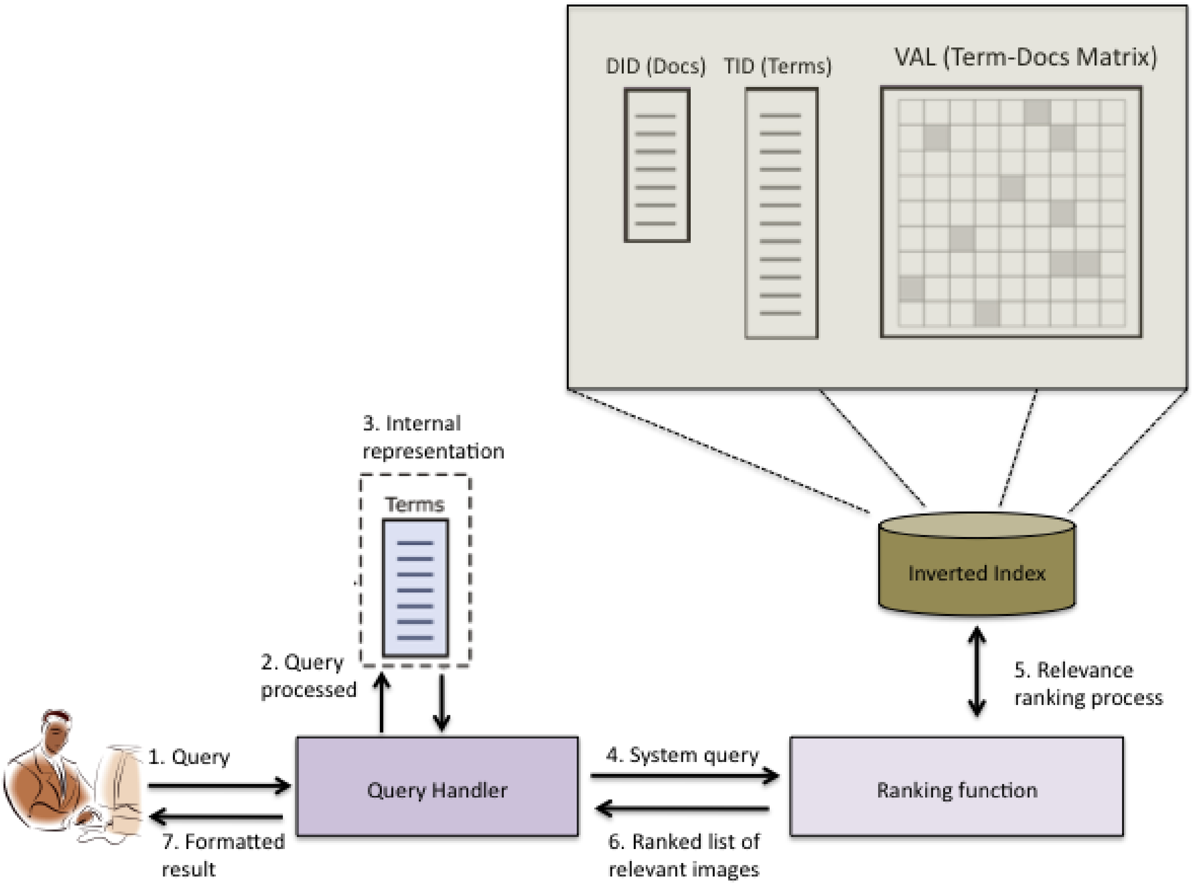

Figure 5.13: Schema for processing a query

The user formulates a query by typing keywords that express his or her information needs. It could be a keywords that describes an image subject, for example "*Animals rights*", "*War and ethics*", or keywords that describe concrete objects or places in the image, for example "*Socrates*", "*Agora*", etc. (referred to as point 1 in figure 5.13).

Query handler processes query by translating query words to the internal representation (illustrated as points 2 and 3 in figure 5.13). It is achieved by applying the same normalization techniques to the query words as to the words for the image collection, i.e. removing of stop-words and stemming by the *Porter* algorithm. By using *tf-idf* formula (see section 2.6 for more details) query handler generates the score for each query term.

The system then searches the inverted index for the query terms, and the ranking function generates a small matrix containing only the query terms and the documents containing these terms (points 4 and 5 in figure 5.13). Value of the *query term x document* matrix is the term weight multiplied by the query weight (see figure 5.14).

Finally, the ranking function generates a score for each image in the matrix by using dot-product similarity measurement, i.e. adding all term weight results (see figure 5.14). The images are ranked in the decreasing order and list of the ranked relevant images is sent to the query handler. Creation of results in formatted HTML and presenting the result to the user are not implemented in this version of the system.

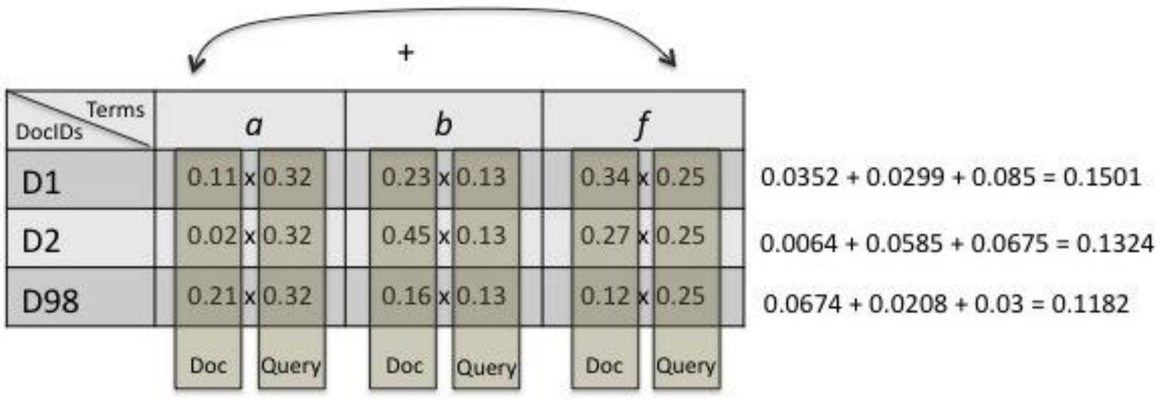

Figure 5.14: An example of generating a score for the retrieved images, by using dot-product similarity measurements.

Chapter 6

Implementation

6.1 Hardware

ImSE is implemented and tested on 2.53 GHz Intel Core 2 Duo CPU with 4 GB RAM running Mac OS X version 10.6.6.

The server for hosting web interface is http://www.uvett.uit.no and placed at the IT department of University of Tromsø. The server is built by LAMP open source technology, i.e. Linux, Apache, MySQL and PHP.

DeStore system is also hosted by the IT department of the University and has URL http://destore.uit.no:8235. Image and HTML files were copied to DeStore using WebDAV client built in MAC OS X version 10.6.6.

6.2 ImSE

ImSE is written in Python version 2.6.1. The general performance of the system has not been prioritized in this thesis, as the main focus of the system is to automatically annotate images and retrieve all relevant images to the search and do not to retrieve non-relevant results.

Beautiful Soup[15] Python library is used to parse and extract information from the HTML and XML files.

Image db and *XML db* are the folders *image db* and *xml db* containing respectively JPG images and XML files created for these images. To make search of images fast and effective, the internal representation of *Inverted Index* is built by tree files; DID, TID and VAL (chapter 5.5 gives more detailed description of the inverted index file structure). In reality *image db*, *xml db* and *Inverted Index* could be created anywhere on external storage, for instance at own DeStore domains for providing secure, fast and efficient storage.

The list of the queries is written by the teachers in the philosophy disciplines at the University of Tromsø and is attaches in appendix A. There is no limitation to the number of keywords in the queries – queries can contain as many keywords as the user likes for formulating his or her

[15] http://www.crummy.com/software/BeautifulSoup/documentation.html

informational needs. In average there are 2 keywords per query. Formulation of queries based on Boolean operators AND, OR, NOT and BUT is not supported by the system.

The system reads queries from the list as *input* and produces the ranked list of retrieved images to the each query as *output. Query Handler* does not assume dependences between keywords in the query. On one hand, it allows retrieval of images that are annotated with terms that approximately matches the query terms. On the other hand, it also allows retrieval of images that are not relevant to the query.

The generated set of retrieved images is translated into the XML format that includes image filename, URL, width, high and size of the images. To make retrieval performance fast and efficient, values for the XML tags are retrieved from the DID file of *Inverted Index*.

Because the interaction between user interface and ImSE (input and output to points 1 and 7 in figure 15.13) is not implemented in this version of the prototype, the XML file with the retrieved set of images is uploaded to the user interface server manually. An example of formatted result in HTML is illustrated in figure 6.2 and was tested in Firefox and Google Chrome web browsers.

6.3 About meta-data

To make the search engine more complete, some important meta-data for the file objects has to be extracted, analysed and indexed. This meta-data is also very useful in the updating mechanism. Initial plan was to extract meta-data for the HTML and image files from DeStore. File id, creation date, modification date and author are data that would be very useful in updating images and index databases. Since the updating mechanism is not implemented in this version of ImSE, meta-data for the file objects are not extracted.

In regards to images and meta-data in the EXIF header, it turns out that most images that are used within HTML do not have any information from EXIF headings available. This is due to the fact that most of these images were downloaded from the Internet, and most image compressions techniques for reducing image size is to remove EXIF information.

Meta-data that was gathered and stored by the system:

- File name – extracted from the file URL
- Width – calculated by using Python *Image* library
- Length – calculated by using Python *Image* library
- Image size – calculated by using standard Python library (*os.stat()*)
- Image URL – calculated by using *geturl* of the Python *urllib2* library

Filenames are used as unique identifiers in this system and replace file object IDs that are available as a DeStore property. Since all image files in a given data collection have unique names, it is not a problem to use filenames as a unique identifier for the images. Width, length, size and URL properties are used to present the image to the user.

6.4 Web interface

The simple web interface is written in PHP and is intended as a test environment for the implementation. As mentioned in chapter 6.2, the interaction between web interface hosting server and ImSE is not finished for this version of the prototype.

Figure 6.1 illustrates the web interface for the user where he or she can type keywords for the search. Figure 6.2 illustrates user interface for the formatted output to the query “War and terror”.

Figure 6.1: Web interface for the image search on DeStore

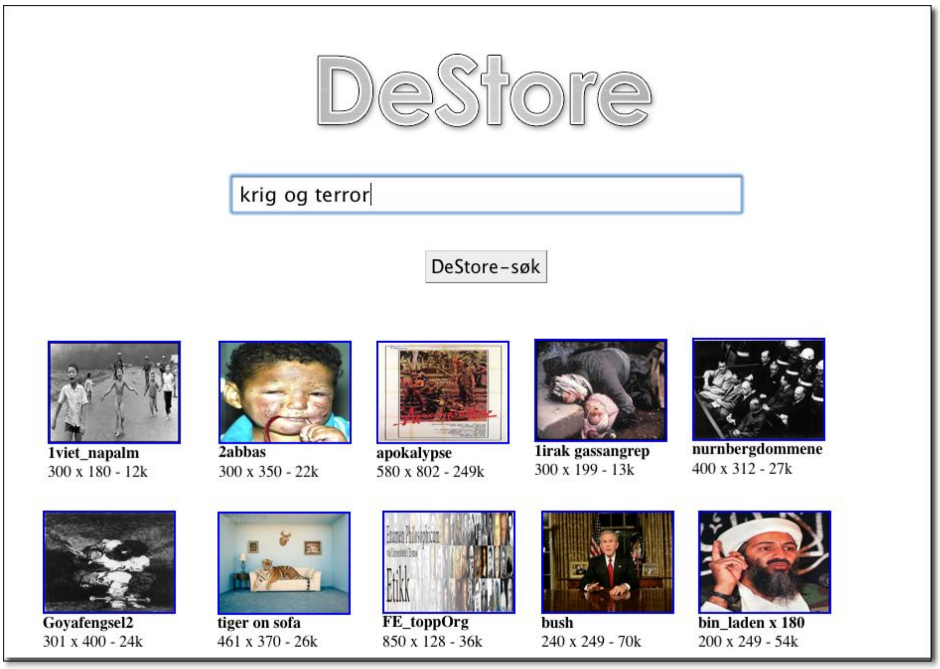

Figure 6.2: example of the web interface to the query “*war and terror*”

Chapter 7

Results and evaluation

In this chapter the results of retrieval sets to the list of queries attached in the appendix A will be presented. The retrieval performance of ImSE will be measured and evaluated. The system will be also compared to the already existing search engine in Fronter – Solr.

7.1 Fronter image archive and Solr

This section will describe the Fronter image archive that is used for testing of Solr retrieval performance in this thesis, and give more a detailed description of Solr search technique.

7.1.1 Fronter image archive

The Fronter image archive (*"(Bildearkiv)"* in figure 7.1) was created for the philosophy disciplines at University of Tromsø. This archive is used to test Solr retrieval performance in this thesis and contains approximately 200 images. These are organized in the user-created hierarchy of folders. Most of the images have a good and describing title. Naming and categorizing images into folders was done for the philosophy courses before this thesis was initiated. The intention was to make image search easier for the users because of the image search limitations by Solr in Fronter.

But when users began to upload their own images to the archive, it turned out that they did not named images as intended. Thus images with the bad filenames such as "*untitled.bmp*", "*p05002.jpg*", "*exp_nett_topporg.jpg*", "*picture 025.jpg*" etc. appeared in the archive (attention to the mages with the bad names is given in the evaluation, section 7.5).

Image files in Fronter has a number of properties as it illustrated in figure 7.1 and descriptions of the file properties are given in table 7.1. Fronter treats these properties as a local "meta-data" and stores it in the Fronter database, and not as a part of the image file header. This means that properties are not stored as a part of the EXIF header for the JPG and JPEG files. Thus, if an image is copied outside Fronter, e.g. DeStore or in the computers local file system, properties will not follow the image.

Image file properties in Fronter	Description
Title	Name of the image as given by users. This does not have to be the same as the image filename. Image name is obtained from the "*Title*" field of the file property in Fronter. (Figure 7.1)
Author	Name of file publisher in Fronter.
Description	Corresponds to the manual annotation of the image. Obtained from the "*Description*" field of the file property in Fronter. (Figure 7.1)
Path to the image location in Fronter Archive	Corresponds to the folder structure in Windows explorer or other file system structures, e.g., *Finder* in Mac OS. It is a user-created hierarchy of files and folders. For example "(*Bildearkiv)/filosofer*". (Figure 7.1)

Table 7.1: Description of image file properties in Fronter

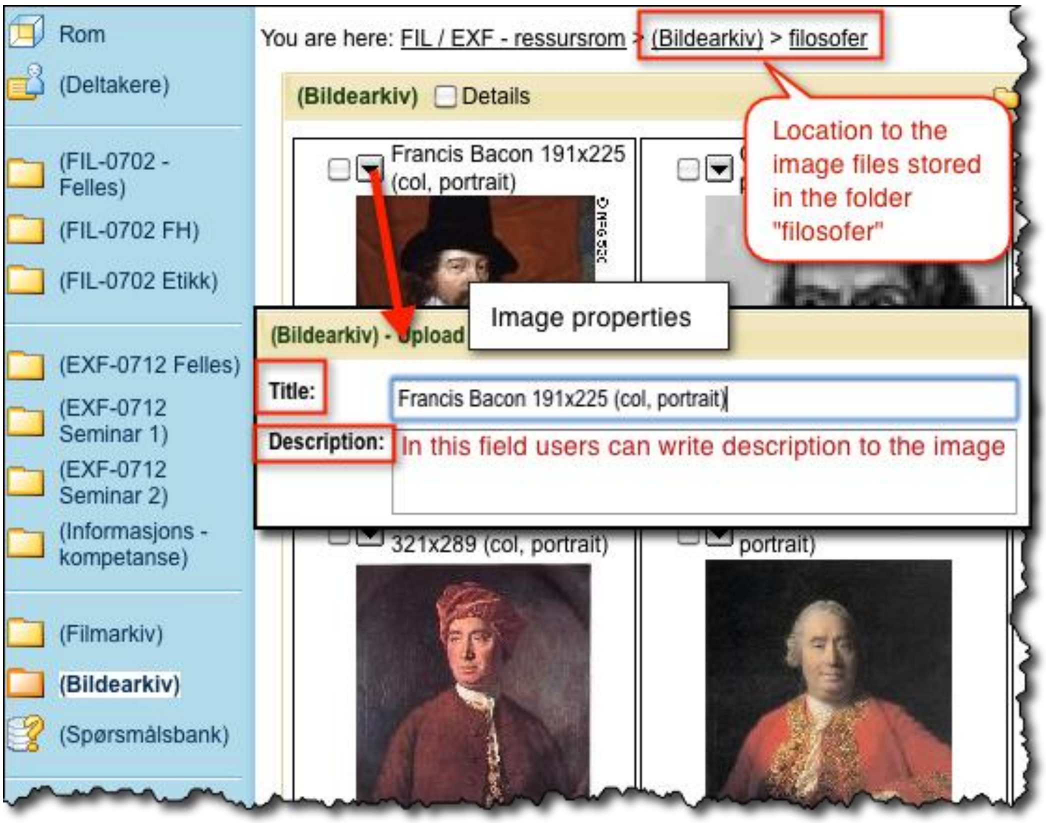

Figure 7.1: Illustrates image properties where title corresponds to the image file name, description field corresponds to the image manual annotation, and path to the user-created folder where images stored in: "*(Bildearkiv)/filosofer*"

Text in the title and description fields and hierarchy of files and folders are user-created. Author property follows the user who published or created the file or folder in Fronter automatically. But this property is not relevant for this thesis.

7.2.2 Solr

As mentioned in chapter 3.2 – Solr indexes/annotates files using text-based techniques. Users search desired information by typing keywords into search filed.

Solr uses the textual information available within image file properties described in table 7.1 to index and search for the images. It seems that Solr treats the query keywords as a string and matches it exactly to the sequence of the text available with image properties. It does not make a partial search for the query keywords as ImSE does. For example, if the query is "*Maleri, painting*" Solr will search images that are annotated with exactly the same string, while ImSE will make a search for "*maleri*" and "*painting*" separately.

Assume that an image is manual annotated in Fronter, i.e. there is some text in the description field, and it is referred to in an HTML document. Than the image description will be represented identically as the text in the ALT and TITLE fields of the IMG tag in the document. This means that manual annotation in Fronter will become "public" in the sense of that ImSE and other search engines will see this annotation as the values of the ALT and TITLE fields of the image tag: *<img ... title=”image file description” alt=”image file description”*>.

There is a total of 5,5 % (11 out of 200) annotated images in Fronter Archive for the philosophy disciplines, and none of these images are referred to in HTMLs. Among these 11 images there are 7 annotated copies.

If the user does not give a proper title to the image in Fronter, does not add any description and the keywords in location path is not logical – Solr has no chance to find this image to the query containing meaningful keywords such as "*filosofer*". Solr does not rank retrieved image set, i.e. it consider the image either relevant or not to the query.

However, Solr is chosen as a comparison system because it is interesting to see the results of the measurements of these two systems, where both use text-based approaches but in different ways. ImSE annotates images by the keywords extracted from the textual context images are used within, while Solr uses textual information available as image properties. ImSE makes a partial match for the query keywords, while Solr does exactly match for the sequence of the query keywords. Solr is also the only search engine that is available in Fronter.

7.2 Queries

This section gives a description and categorization of the queries used to test ImSE and Solr retrieval performance. It also gives an overview of how the queries will be evaluated. List of queries is attached in appendix A.

Queries and sets with the relevant images to each query are the user-defined for the test in this thesis. The users in this test are two teachers in the philosophy disciplines at the University of Tromsø. Firstly, they formulated queries based on the question "*What images do I want to search for if creating a new course in philosophy disciplines*?" In total 29 queries were formulated. Secondly, the users looked through the image collection in Fronter image archive and chose all relevant images to the each query in the list. After the retrieval sets were defined, the users were asked to assign the degree of relevance to each image for each query measured by numbers 1-5. 1 meaning the image is very relevant to the query, 5 meaning the image is not relevant:

- 1: 100 – 90% of relevance to the query
- 2: 90 – 70% of relevance to the query
- 3: 70-40% of relevance to the query
- 4: less than 40% relevance to the query (unsure value)
- 5: 0% relevance to the query

By ranking images with the degree of relevance we are trying to capture different interpretations/understandings of what images are about, and if they somehow can be relevant to the query. If an image is given the value 4 it is difficult to decide if the image is relevant to the query or not. Some will regards it as relevant while others will not. That is why the value is marked as unsure.

7.2.1 Query categories

User-formulated queries for this test are categorized into object and subject:

- Object queries for the philosophy disciplines are normally about specific places or persons, e.g., "*Socrates*", "*Agora*", and include query numbers 6,7,9,10,12,13,15-17,19,20,21,24-27 and 29 (see queries in appendix A).
- Subject queries for the philosophy disciplines normally are about events or subjects within the history of philosophy and ethics, e.g., "*war and ethics*", "*animals rights*", "*Sophists*", or about image categories, e.g. "*Antiquity*", "*Paintings*", "*Illustrations about philosophy history*". The subject queries include numbers 1-5,8,11,14,18,22,23 and 28 attached in appendix A.

It is important to distinguish between object and subject queries for this test, because object queries are very precise in formulation and there is no doubt if an image is relevant to it or not. Subject queries are often ambiguous in formulation, and decision of what images are relevant or not, as mentioned above, depends on the users interpretations/understandings of the subject.

I believe that retrieval performance will be also reflected by the query category, and retrieval set for the subject search will include more non-relevant images in user interpretation of relevance than for the object queries.

To summarize, queries are distinguished in to two categories: object and subject. And retrieval sets of images will be evaluated by the degree of relevance as it illustrated in table 7.2.

Query category	Evaluation of the system performance
Object category	• Relevance degree 1-3 • Relevance degree 1-4
Subject category	• Relevance degree 1-3 • Relevance degree 1-4

Table 7.2: Shows query categories and how they will be evaluated

7.2.2 Ambiguously formulated queries

As previously mention in chapter 2.3, a big challenge in image retrieval lies in how users search for images. For example if one searches for "*Venezia*", what kind of images would a user expect to be retrieved: photos or paintings of geographical places in Venezia? Coffee machines? Furniture? Perfumes? Clothes? Etc. All these images retrieved by a search engine will satisfy the query keyword "*Venezia*". This illustrates that there is a gap between what user means by the keyword at the moment of submitting a query and what kind of images the image retrieval systems might associate with the query keyword.

In the query list, attached in appendix A, there are two queries that are distinguished from the others by their formulation: "*Heading ex.phil, filosofihistorie*", "*Illustrasjoner i filosofohietorie*" (respectively query numbers 2 and 4 in appendix A).

If we look at the query "*Heading ex.phil filosofihistorie*" we understand that the query has something to do with the history of philosophy. But what kind of images does the user expect to be retrieved? Does "*Heading*" means cover to the books or top banners or posters containing the word "*filosofihistorie*"?

If we look at the query "*Illustrasjoner i filosofihistore*", what does the user mean by the illustration? Illustration is a very wide concept that might include everything that illustrates – charts, diagrams, photographs, paintings, drawing, sketches, cartoons, etc.

The ambiguously formulated queries will obviously decrease the systems performance, but anyways are important to include in the test because users often want search results to just such queries.

7.3 Evaluation method

This section will compare Solr and ImSE test collections of the images. The systems will also be compared according to the context considered as relevant in image annotations for this thesis. A described of what criteria and measures will be used to evaluate the retrieval performance for both systems will also be included.

7.3.1 Comparison of Solr and ImSE

Image collection from the image archive in Fronter and all HTML files (pages) created for the philosophy disciplines in Fronter are copied to DeStore and further referred to as a test collection. As previously mentioned in chapter 7.1, image file properties in Fronter will not follow the image copy, unless it is referred to in an HTML document (*in this case none of annotated images in Fronter are referred to in any document*), but title property for the images in Fronter will be identical to the image filenames in DeStore.

List of the context considered for this thesis	Corresponding context, i.e. properties used by Solr	Corresponding context used by ImSE
Image filename	Title. For this test collection it corresponds to the image filename.	Image filename
Image URL	Path to the image location in Fronter Archive	Image URL in DeStore
Manual annotation of the image	Description	• If image is referred to in HTML document in Fronter – corresponds to the text in the ALT and TITLE attributes in IMG tag: *<img title="description" alt="description">* • If image is NOT referred to in HTML document in Fronter – no corresponding text fields.
Image caption	No corresponding property	Text inside the ** tag
Near text	No corresponding property	Text inside the parent TR tag of the IMG tag: *<tr>"near text"<img …>"near text"</tr>*

Table 7.3: List of the context considered for this thesis with according to it context used by Solr and ImSE.

Table 7.3 compares lists of the context relevant to the image content and its semantics considered for this thesis (according to the definition given in chapter 4.3) corresponding to its context used by Solr and ImSE.

From the entries in table 7.3 we can conclude that images in both test collections, Fronter image archive and DeStore, are equally annotated when it comes to the title/image file names and path to the image location in Fronter/image URL in DeStore.

Few image annotations in Fronter, and searching images by exactly matching query keywords indicate that Solr will have a poor retrieval performance.

7.3.2 Adjustments made for Solr search

To adjust for the differences between the search methods in Solr and ImSE, test queries containing more than one word were divided into sub-queries for Solr (totally 14 queries). Each sub-query corresponded to each keyword in the query. The retrieved image set in Solr is a combination of retrieval sets to the sub-queries. For example: search for the query "*fotball, football, fotballkamper*" was divided into tree sub-searches in Solr: "*fotball*", "*football*" and "*fotballkamper*". Retrieval sets to the queries were combined with retrieval results for these tree sub-searches. Such adjustment will improve the search results for Solr thus having a better comparison basis of the retrieval performance between the systems. Without this adjustment, Solr retrieval performance is very low and there is no basis for comparison.

7.3.3 Evaluation measures

As already mentioned in chapter 7.2, evaluation of the relevance is a user-oriented task that relies on the thier background, knowledge, education, age, etc. which makes evaluation of the image retrieval systems relative/subjective. The degree of relevance for all images are measured with numbers 1 to 5, where 1 is 100-90% of image relevance to the query and 5 is 0% relevance to the query. The degree of relevance is used only for the systems evaluation.

Recall, *precision* and *F-score* are used as evaluation measures of the systems. *Recall* is the relation between the retrieved relevant images and all relevant images in the whole data collection, where *precision* is the relation between the retrieved relevant images and all retrieved images. *F-score* is a combination of recall and precision values and gives an average score of the system efficiency (see chapter 2.7 for more detailed description). Precision score 1.00 means that all images found to the query are relevant, and recall score 1.00 means that all relevant images from the whole test collection are retrieved to the query. Two sets of the measures will be computed:

1. Recall rec_1, precision pre_1 and $F\text{-}score_1$ for the retrieved image sets with a relevance degree of 1-3

2. Recall rec_2, precision $pre_2$and $F\text{-}score_2$ for the retrieved image sets with a relevance degree of 1-4

When it comes to ImSE the ranked set of images generated by the systems will also be compared to the user ranking of the images to the queries.

7.4 Retrieving results

This section reports the retrieval results of the systems Solr and ImSE for the list of the queries attached in appendix A. The next section will evaluate and discuss the retrieving results. Tables that represent the detailed retrieval results with the precision and recall values for each query are attached in appendix B.

Tables 7.4 and 7.5 report results for the object and subject queries separately. It is done to give a better overview of the hit results and to show the difference in retrieval performance for those two query categories. *R* in the tables is the number of relevant images in the test collection to the query, *A* is the number of total hits by the system and *Ra* is the number of relevant hits by the system.

Object query number	***R*, relevance degree 1-3**	***A***		***Ra***	
		Solr	**ImSE**	**Solr**	**ImSE**
6	11	2	8	2	8
7	6	4	6	4	6
9	10	3	12	3	7
10	6	4	14	4	6
12	11	1	6	1	6
13	4	1	4	1	3
15	2	2	2	2	2
16	2	2	2	2	2
17	3	3	3	3	3
19	4	4	5	4	4
20	5	5	5	5	5
21	4	4	12	4	4
24	1	0	1	0	1
25	2	0	2	0	2
26	2	0	7	0	2
27	9	0	10	0	6
29	3	0	3	0	3
Object query number	***R*, relevance degree 1-4**	***A***		***Ra***	
		Solr	**ImSE**	**Solr**	**ImSE**
10	8	4	14	4	8
26	5	0	7	0	5

Table 7.4: Number of total and relevant hits for Solr and ImSE for the object queries.

Table 7.4 summarizes the number of total and relevant hits for Solr and ImSE for the object queries. Number of hits that include images with the relevance degree of 4 is only relevant for the queries 10 and 26 for this category.

Table 7.5 summarizes the number of total and relevant hits for Solr and ImSE for the subject queries. Numbers of hits that include images with a relevance degree of 4 are only relevant for queries 2,4,8,11,18 and 22 for this category. Ambiguous formulated queries are marked bold (query numbers 2 and 4).

Subject query number	***R*, relevance degree 1-3**	***A***		***Ra***	
		Solr	**ImSE**	**Solr**	**ImSE**
1	14	0	12	0	10
2	**3**	**0**	**25**	**0**	**1**
3	25	26	26	25	25
4	**32**	**0**	**18**	**0**	**12**
5	21	0	8	0	3
8	6	0	5	0	4
11	3	0	3	0	1
14	15	0	8	0	1
18	15	0	5	0	2
22	37	23	29	23	28
23	43	23	26	22	25
28	3	2	3	2	3
Subject query number	***R*, relevance degree 1-4**	***A***		***Ra***	
		Solr	**ImSE**	**Solr**	**ImSE**
2	**8**	**0**	**25**	**0**	**6**
4	**33**	**0**	**18**	**0**	**13**
8	7	0	5	0	5
11	5	0	3	0	3
18	17	0	5	0	4
22	38	23	29	23	29

Table 7.5: Number of total and relevant hits for Solr and ImSE for the subject queries.

Next, the retrieval results for the Solr and ImSE will be represented graphically with the separate graphs for the object and subject queries. This is done because it is useful in a comparison of the systems – it is easier to see what query category the systems are good at and not. As we see from the results in tables 7.4 and 7.5, the number of hits for Solr is not affected by including images with the relevance degree of 4 into retrieval sets. Therefore, comparison of retrieval sets for the images with relevance degrees of 1-3 and 1-4 is relevant and shown only for ImSE in this test.

Figures 7.2 and 7.3 show precision and recall values for all 29 queries attached in appendix A where retrieval sets include images with a relevance degree of 1-3. Average retrieval performance for the systems is summarized in table 7.6.

Precision, *pre*$_1$		**Recall, *rec*$_1$**		**Average score of the system efficiency, *F-score*$_1$**	
Solr	**ImSE**	**Solr**	**ImSE**	**Solr**	**ImSE**
0.51	0.72	0.36	0.71	0.42	0.71

Table 7.6: Summarization of the average retrieval performance for Solr and ImSE for all queries in the test (29 queries). Retrieval sets includes images with a relevance degree of 1-3.

In average ImSE has 29% better precision and 49% better recall. F-score value shows that ImSE has in general 41% better retrieval performance than Solr.

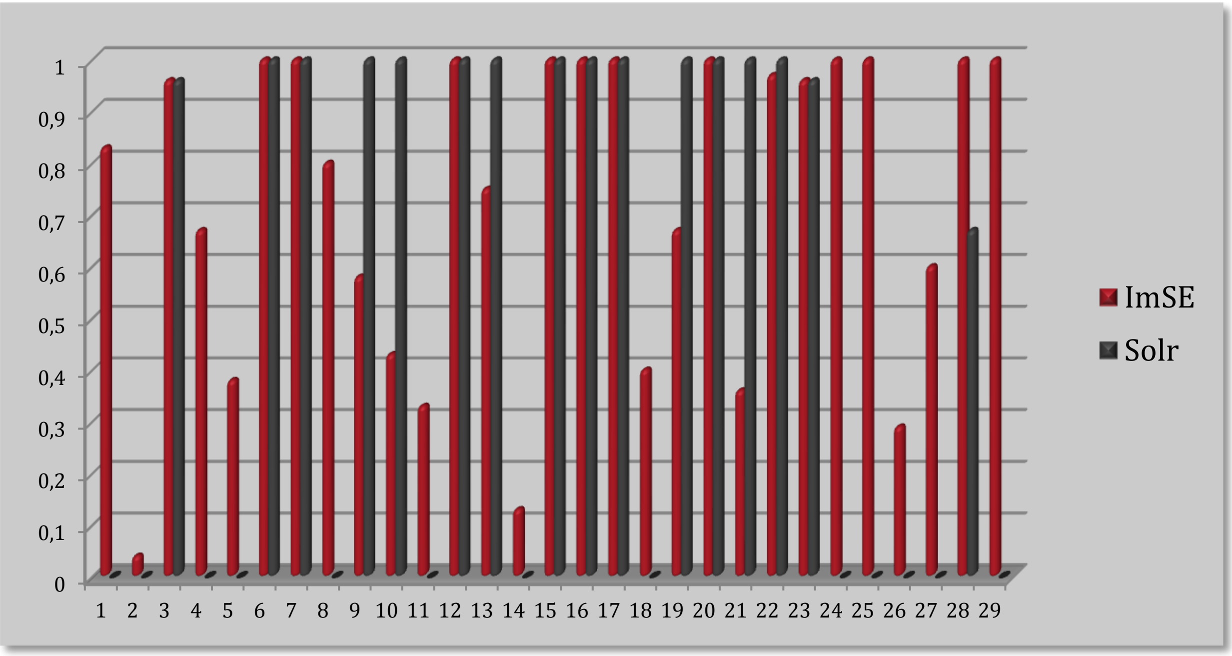

Figure 7.2: ImSE and Solr precision measurements for all queries attached in appendix A where retrieval sets include images with a relevance degree of 1-3 (queries 2 and 4 are defined as ambiguously formulated), with average *pre*$_1$=0.72 for ImSE and *pre*$_1$=0.51 for Solr.

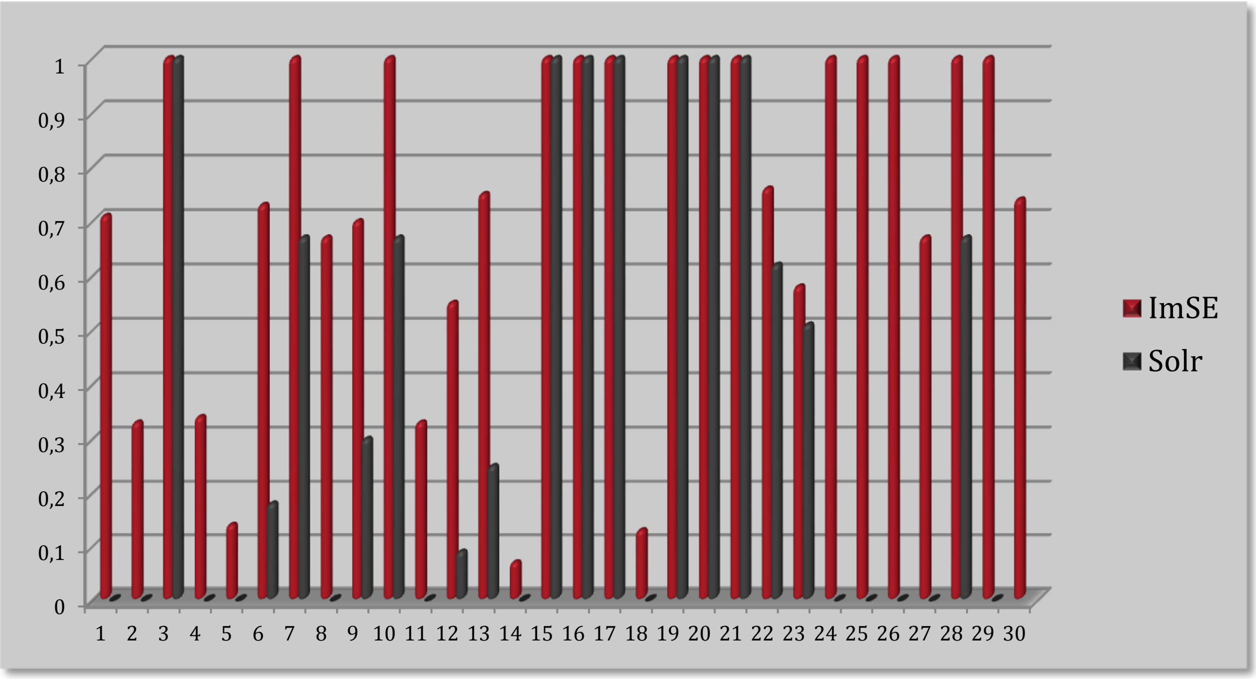

Figure 7.3: ImSE and Solr recall measurements for all queries attached in appendix A where retrieval sets include images with a relevance degree of 1-3 (queries 2 and 4 are defined as ambiguous formulated), with average rec_1=0.71 for ImSE and rec_1=0.36 for Solr

Figures 7.4 and 7.5 show Solr and ImSE precision and recall measurements for the object queries, which include images with a relevance degree of 1-3 in the retrieval sets. Average retrieval performance for the systems for this category is summarized in table 7.7.

Precision, pre_1		**Recall, rec_1**		**Average score of the system efficiency, $F\text{-}score_1$**	
Solr	**ImSE**	**Solr**	**ImSE**	**Solr**	**ImSE**
0.71	0.81	0.48	0.91	0.57	0.85

Table 7.7: Summarization of the average retrieval performance for Solr and ImSE for the object queries where retrieval sets include images with a relevance degree of 1-3.

For this category of queries ImSE has 12% better precision and 47% better recall than Solr. F-score measurement shows that ImSE has 33% better retrieval performance than Solr for the object queries.

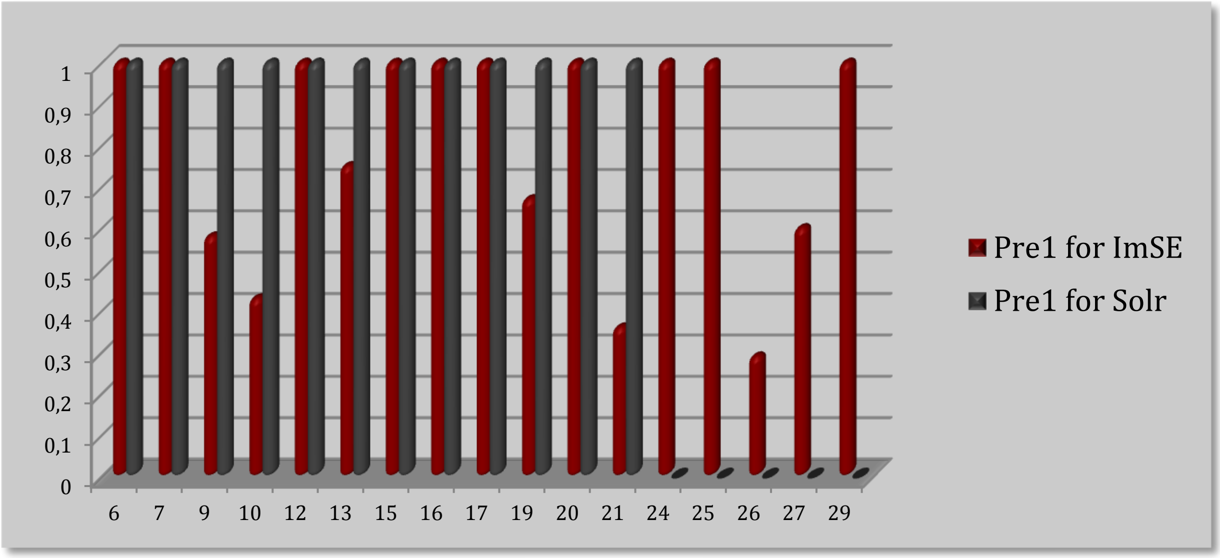

Figure 7.4: ImSE and Solr precision measurements for the object queries that include images with a relevance degree of 1-3 in the retrieval sets. Average pre_1 = 0.81 for ImSE and pre_1 = 0.71 for Solr.

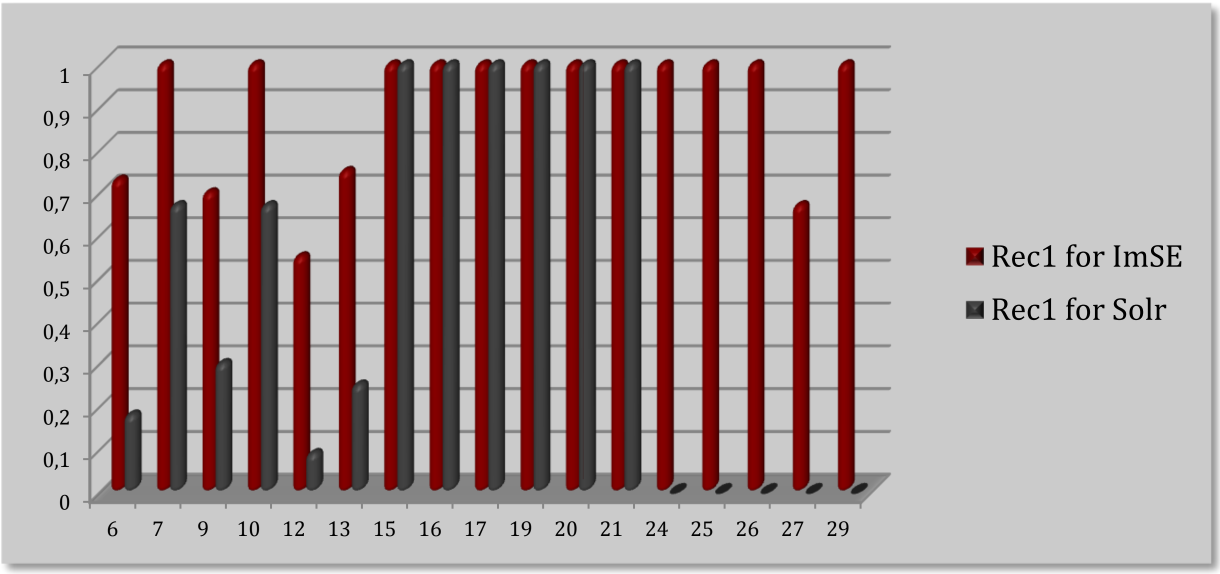

Figure 7.5: ImSE and Solr recall measurements for the object queries that include images with a relevance degree of 1-3 in the retrieval sets. Average rec_1 = 0.91 for ImSE and rec_1 = 0.48 for Solr.

Figures 7.6 and 7.7 show ImSE and Solr precision and recall measurements for the subject queries where retrieval sets include images with a relevance degree of 1-3. Average retrieval performance for the systems for this category is summarized in table 7.8.

Precision, *pre*$_1$		**Recall, *rec*$_1$**		**Average score of the system efficiency, *F-score*$_1$**	
Solr	**ImSE**	**Solr**	**ImSE**	**Solr**	**ImSE**
0.30	0.62	0.23	0.51	0.26	0.56

Table 7.8: Summarization of the average retrieval performance for Solr and ImSE for the subject queries where retrieval sets include images with a relevance degree of 1-3.

For this category of queries ImSE has 52 % better precision and 55% better recall than Solr. F-score values show that ImSE has 54% better retrieval performance for the subject queries.

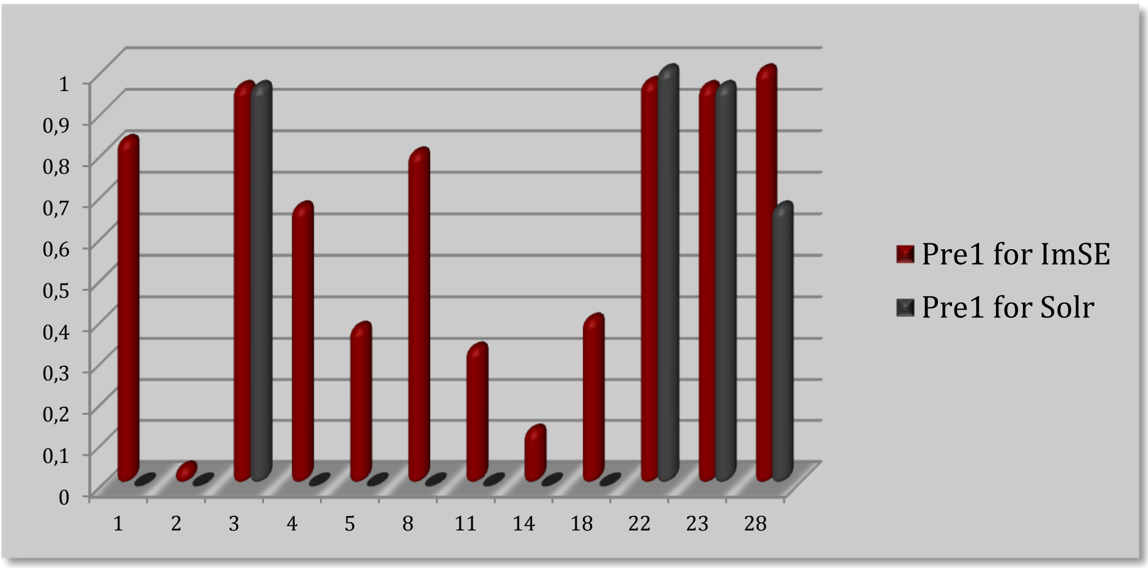

Figure 7.6: ImSE and Solr precision measurements for the subject queries that include images with a relevance degree of 1-3 (queries 2 and 4 are ambiguously formulated). Average *pre*$_1$ = 0.62 for ImSE and *pre*$_1$ = 0.30 for Solr.

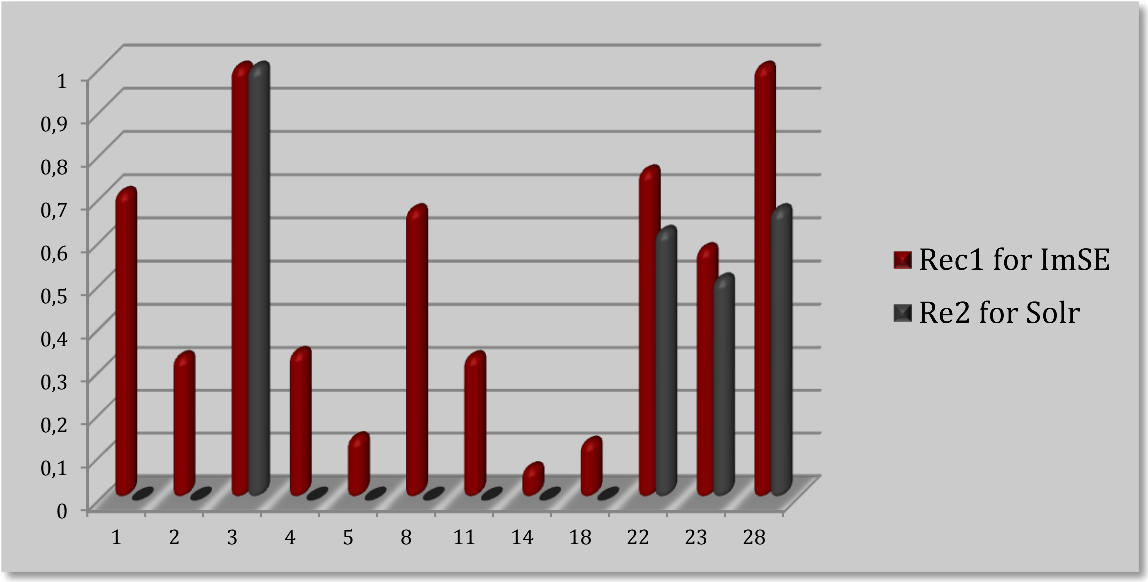

Figure 7.7: ImSE and Solr recall measurements for the subject queries that include images with a relevance degree of 1-3 (queries 2 and 4 are ambiguous formulated). Average rec_1 = 0.51 for ImSE and rec_1 = 0.23 for Solr

Comparison of the retrieval results for the queries where retrieval sets includes images with relevance degree of 1-3 and 1-4 in this thesis is relevant only for ImSE. As mentioned in section 7.2, the relevance degree 4 (40% or less) is an unsure value, i.e. the decision if an image is relevant to the query or not relies on the users interpretations/understandings of the search. Good examples of such queries are 2, 4 and 26 (see table 7.4 for the hit results to these queries).

Next, figures 7.8 and 7.9 are graphs for ImSE precision and recall measurements for the object queries where retrieval sets include images with relevance degree of 1-3 and 1-4. Average retrieval performance for ImSE for this category is summarized in table 7.9. Pre_1, rec_1 and $F\text{-}score_1$ values are performance measurements for the retrieval sets including the images with a relevance degree of 1-3. Pre_2, rec_2 and $F\text{-}score_2$ – images with a relevance degree of 1-4.

Precision		**Recall**		**Average score of the system efficiency**	
Pre_1	**Pre_2**	**Rec_1**	**Rec_2**	**$F\text{-}score_1$**	**$F\text{-}score_2$**
0.81	0.85	0.91	0.91	0.85	0.88

Table 7.9: Summarization of ImSE's average retrieval performance for the object queries where retrieval sets include images with relevance degree of 1-3 and 1-4.

As we see there is no significant difference between precision, recall and F-score values for the object category of queries. According to the results, only precision for the queries 10 and 26 is different if images with relevance degree of 4 are included in the retrieval set, while recall stays unchanged.

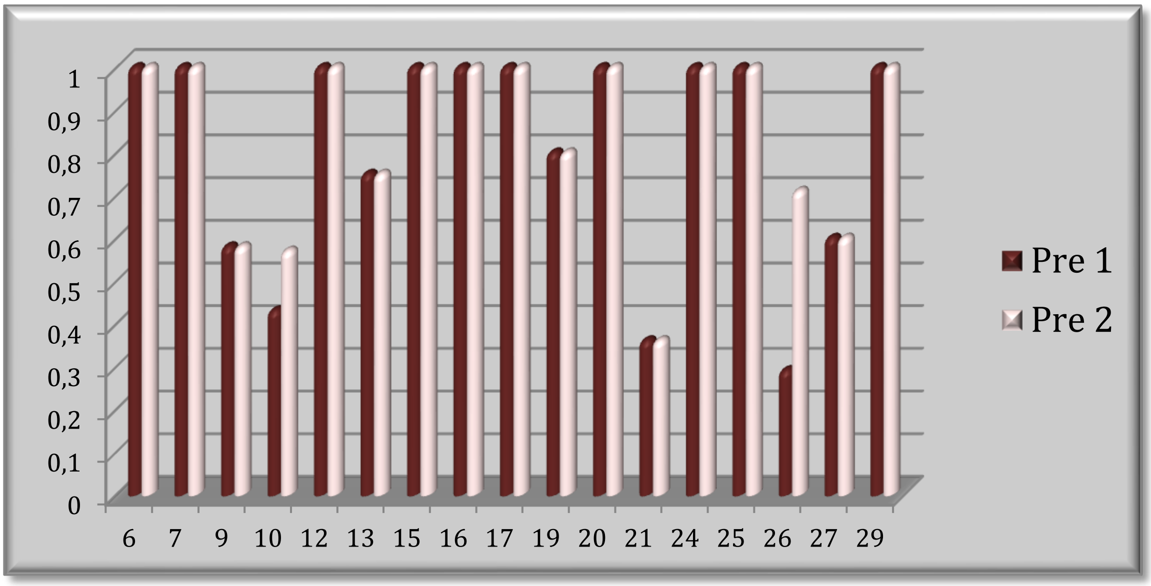

Figure 7.8: ImSE precision measurements for the object queries that include images with relevance degree of 1-3 and 1-4 in the retrieval sets. Average *pre_1* = 0.81 and *pre_2* = 0.85. The diagram shows difference only for the queries 10 and 26.

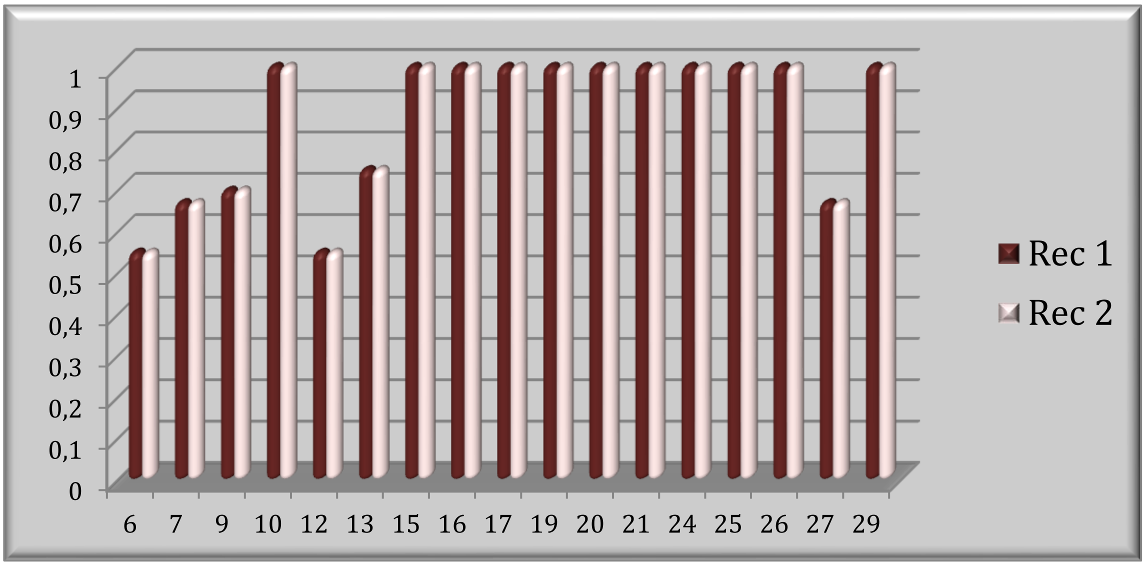

Figure 7.9: ImSE recall measurements for the object queries that include images with relevance degree of 1-3 and 1-4 in the retrieval sets. Average *rec_1* = 0.91 and *rec_2* = 0.91. The diagram shows no difference in recall for this category.

Next, figures 7.10 and 7.11 show ImSE's precision and recall measurements for the subject queries where retrieval sets include images with relevance degree of 1-3 and 1-4. Average retrieval performance for this category is summarized in table 7.10. *Pre_1*, *rec_1* and *$F\text{-}score_1$* values are measurements for the retrieval sets including the images with a relevance degree of 1-3. *Pre_2*, *rec_2* and *$F\text{-}score_2$* – images with a relevance degree of 1-4.

Precision		Recall		Average score of the system efficiency	
Pre_1	**Pre_2**	**Rec_1**	**Rec_2**	**F-$score_1$**	**F-$score_2$**
0.62	0.75	0.51	0.58	0.56	0.65

Table 7.10: Summarization of ImSE average retrieval performance for the subject queries where retrieval sets include images with the relevance degree of 1-3 and 1-4.

If images with the relevance degree of 1-4 are included in the measurement it gives 22% improvement of the precision and 12% improvement of the recall. In this case precision and recall do not follow an inverse relationship, i.e. improvements of one does not leads to the deterioration of the other. There is 14% improvement of the system performance according to the F-score values.

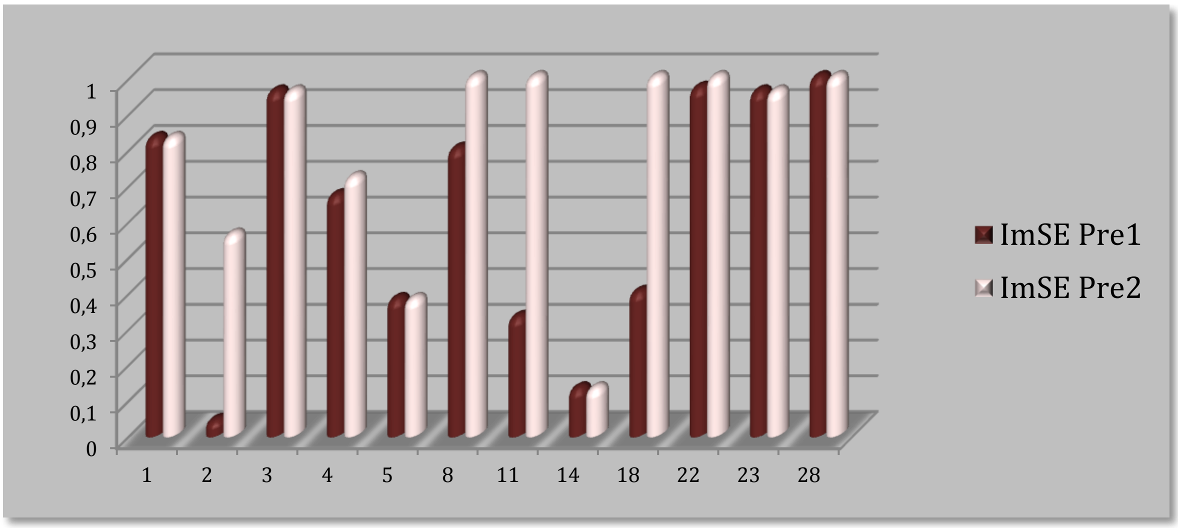

Figure 7.10: ImSE precision measurements for subject queries where retrieval sets include images with relevance degree of 1-3 and 1-4. Average *pre_1* = 0.62 and *pre_2* = 0.75

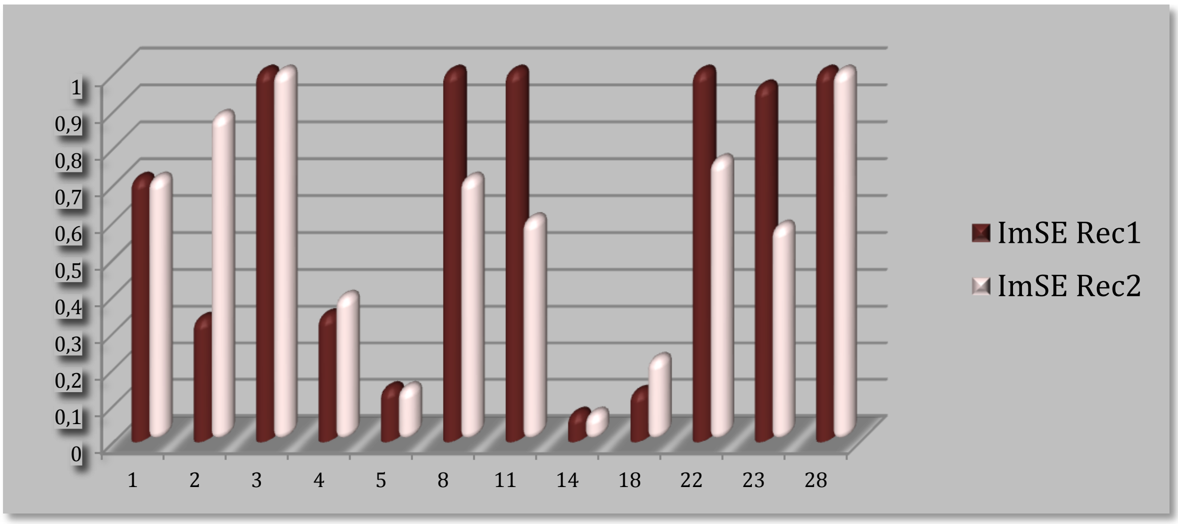

Figure 7.11: ImSE recall measurements for the subject queries where retrieval sets include images with relevance degree of 1-3 and 1-4. Average *rec_1* = 0.51 and *rec_2* = 0.58

Summarization of the average precision, recall and F-score results for the different query categories and different sets of the relevance degrees are in table 7.11. Figure 7.12 shows the graphical view of the retrieval performance for Solr and ImSE.

Query category	Relevance degrees \ System	ImSE	Solr
Object	1-3	Precision = 0.81 Recall = 0.91 F-score = 0.85	Precision = 0.71 Recall = 0.48 F-score = 0.57
	1-4	Precision = 0.85 Recall = 0.91 F-score = 0.88	Precision = 0.71 Recall = 0.47 F-score = 0.57
Subject	1-3	Precision = 0.62 Recall = 0.51 F-score = 0.56	Precision = 0.3 Recall = 0.23 F-score = 0.26
	1-4	Precision = 0.75 Recall = 0.58 F-score = 0.65	Precision = 0.3 Recall = 0.23 F-score = 0.26

Table 7.11: Summarization of the average performance measurements for ImSE and Solr.

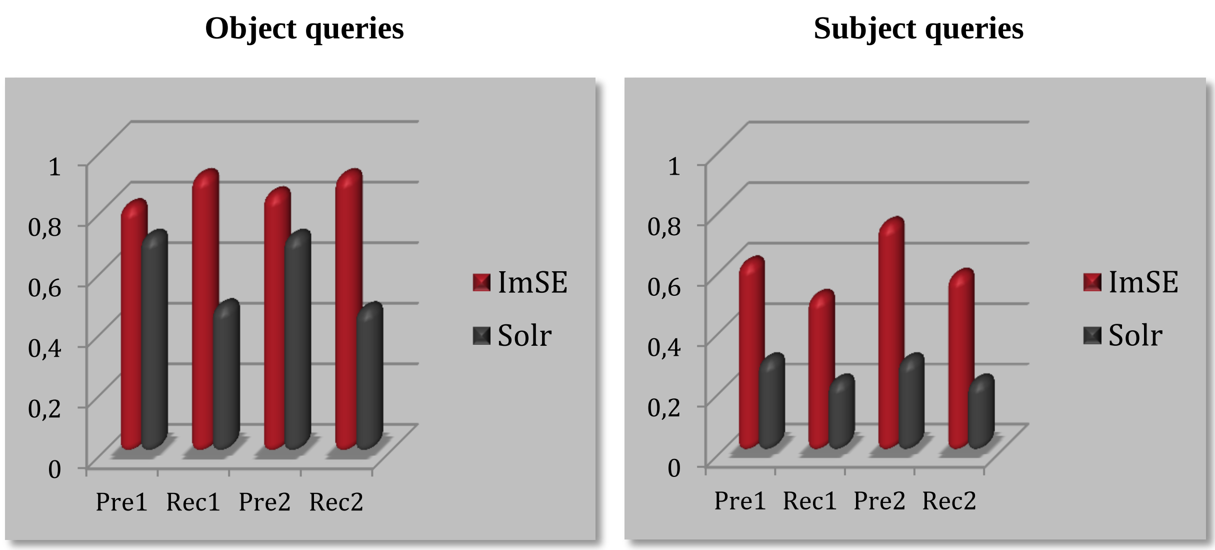

Figure 7.12: Summarization of the average measurements of retrieval performance for Solr and ImSE.

7.5 Evaluation

As mentioned previously, Solr treats query words as strings and do exactly query string matches to the text available with image properties, while ImSE does the partial matches of the query keywords. The adjustment that was made for the Solr search was to divide queries containing more than one keyword into sub-queries. Even with this adjustment in favour of Solr retrieval performance, summarized results in table 7.11 and figure 7.12 show that ImSE has much better performance in average for all categories of queries, and separately for the object and subject. From the same results we also see that retrieval performance for Solr is not affected by including images with a relevance degree of 4 in the retrieval sets. This is because Solr has general bad retrieval performance with F-score 0.42 for all test queries compared to the ImSE F-score value 0.77, including images with a relevance degree of 4. This is in average a 45% improvement in image retrieval.

The evaluation of the systems will be divided in the 6 parts:

1. Evaluating of the retrieval performance for the object queries
2. Evaluating of retrieval performance for the subject queries
3. Evaluation of retrieval performance ambiguous formulated queries
4. Evaluating of retrieval performance for the images with bad filenames
5. Evaluation of images used within different contexts
6. Evaluation of ImSE ranking

Evaluation of parts 1 and 2 will be done by comparing and analysing the context listed in table 7.3 (see sub-section 7.3.1). As mentioned previously, images in both test collections, Fronter image archive and DeStore, are equally annotated when it comes to the titles/image filenames and path to the image location in Fronter/Image URL in DeStore.

7.5.1 Evaluation of the retrieval for the object queries

Analysing words in all queries for the object category we see that they do not contain words that match any words in the image path in Fronter Archive or image URL in DeStore. All images relevant for this set of queries in Fronter have no description. Thus, retrieval performance of Solr in this case depends only on good and meaningful image titles, while retrieval performance of ImSE will rely on meaningful image filenames, and if image was referred to in HTML – image captions and near text.

Queries 12, 24, 25, 26, 27 and 29 from this category (see appendix A for the list of queries) contain two keywords per query. For Solr it means that retrieval sets consist of the two sub-sets for each keyword in the query. Hit results in table 7.4 shows that Solr has no hits for the queries 24-27 and 29.

If we look at the performance retrieval results for the object queries (illustrated in figures 7.4 and 7.5 and results in table 7.6), precision measurements for Solr are either 1.0 or 0. This means the relevant images are either found or not found by the system. It can be explained with the fact that Solr looks for exact matches for query keywords to the keywords found in an image title, thus, query either has a match or it does not. Recall measurements have some variations that depend on how many relevant images Solr is able to find in the test collection. It also depends on how many relevant images have matching titles. An average Solr has a good precision (0.71) for the object queries and not so good recall (0.48) for the retrieval to the object queries. Good precision can be explained with good and meaningful titles for the relevant images. Recall shows that not all images in the collection have titles corresponding to the image content.

When it comes to ImSE, image file names are the same for the test collection in DeStore as it is for images in Fronter Archive, since they are copies. Relevance performance for ImSE is expected to be at least the same or even better than for Solr, because ImSE uses image captions and near text in addition to the context that Solr uses (see table 7.2). Figure 7.4 shows that ImSE precision values are 1.0 or lower than Solr's to the queries 6-21. Solr is more precise in retrieving relevant images, but recall (figure 7.5) values for ImSE are much better (45%) than Solr's for the same queries (6-21), because ImSE is able to find more relevant images that Solr.

The advantages of image annotations relevant to its context are captured by the recall and disadvantages, i.e. not all context is assumed to be relevant to the image always does, are captured by the precision.

The average precision, recall and F-score measurements for the object queries, including relevance degree of 4, are summarized in table 7.11 in the previous section. We see that ImSE on average has a 35% better retrieval performance for the object queries.

7.5.2 Evaluation of retrieval for the subject queries

By analysing words in all queries for the subject category we can see that some of them contain words that matche words for some folders in Fronter Archive and image URL in DeStore. There is only one relevant image in Fronter with the description text. (As mentioned previously, none of the images in Fronter with the description are referred to in any HTML which means that the description of the images is not "public".) But there are many relevant images in DeStore that are annotated with the image captions and near text.

Figures 7.6 and 7.7 in the previous section show that Solr, in general, has a very poor retrieval performance to the subject search – as expected. It retrieves images only to the following queries:

- Query "*fagansvarlige*" with precision = 0.96 and recall =1.0 compared to ImSE with precision = 0.96 and recall 1.0.

  The query word matches exactly the word in the location path in Fronter Archive and image URL in DeStore. That is why retrieval performance for Solr and ImSE are equal for the query.
- Query "*Maleri, painting*" is a combination of two retrieval sub-sets to the "*maleri*" and "*painting*" sub-queries for Solr. Precision and recall measurements for this query for Solr are respectively 1.0 and 0.61 and for ImSE – respectively 1.0 and 0.76.

  The query word "*painting*" is an exact match for the keyword "*painting*" in the image title/ filename in both systems. But ImSE retrieves 6 more images to the query because these images are annotated by the relevant context the images are used within. That is why the recall value for ImSE is greater than for Solr.

- Query "*Photo, fotobilder, photobilder*" is combination of three retrieval sub-sets to the "*Photo*", "*fotobilder*" and "*photobilder*" for Solr. Solr's precision = 0.96 and recall = 0.51 compared to ImSE with precision = 0.96 and recall = 0.58.

  The query keyword "*photo*" is an exact match for the keyword "*photo*" in the image title/filename in both systems. But ImSE retrieves 3 more relevant images to the query because these images are annotated by the relevant context the images are used within. That is why the recall value for ImSE is greater than for Solr.

- Query "*fotball, football, fotballkamper*" is a combination of tree retrieval sub-sets to the "*fotball*", "*football*" and "*fotballkamper*" for Solr. Solr's precision and recall for the query are 0.67 and 0.67 compared to ImSE's 1.0 and 1.0

  The query keyword "*fotballkamp*" is an exact match for the keyword "*fotballkamp*" in the image title/filename in both systems. But ImSE retrieves 1 more relevant image to the query because this image is annotated by the relevant context the image is used within. That is why retrieval performance (precision and recall) for ImSE is better than for Solr.

The average precision, recall and F-score measurements, including relevance degree 4, are summarized in table 7.11 in the previous section. We see that ImSE has 60% better retrieval performance than Solr for the subject queries, even with an adjustment to Solr search to improve its retrieval performance.

7.5.3 Ambiguously formulated queries

This sub-section will discuss retrieval performance for the queries that have ambiguous formulation, i.e. it is hard to understand what the user means by the formulated query words. There are queries "*Heading ex.phil filosofihistorie*" and "*Illustrasjoner filosofihistorie*" in the attached query list in appendix A.

Solr has no hit to these queries, because title and location path for the relevant images in Fronter do not match any keywords in the queries.

If we look at the ImSE's retrieval set to the"*Heading ex.phil filosofihistorie*", it contains 25 different types of images: top banners, nature, philosophers, places, paintings, etc. All of these images satisfy the search keyword "*filosofihistorie*". But only 3 are relevant images (including relevance degree of 4) in the test collection. Therefore precision to this query is very low - 0,24, while recall is 0,75.

If we look at the ImSE's retrieval set to the query *"Illustrasjoner filosofihistorie*" there are 33 relevant images (including relevance degree of 4) in the test collection that illustrates paintings, nature, sculptures, cartoons, etc. It seems that users meant the term "*illustrasjon*" literally. The retrieval set for ImSE includes 18 images to the query satisfying the search with 13 relevant images. Precision and recall values for the query are respectively 0,72 and 0,39.

7.5.4 Retrieving images with bad filenames

This sub-section will evaluate retrieval of images with the bad names, i.e. names that do not give any meaningful information about image content.

Solr does not retrieve images with bad names, because they are not located in folders with the names that somehow categorize images, e.g., folders with the names "*philosophers*", "*paintings*", etc., and are not annotated with text in the description field in Fronter.

Images in the test collections that do not have proper name and which are relevant to the numbers of queries are:

- *che_858_1193500351.jpg*, is relevant to query 28
- *EXP_Nett_toppOrg800px.jpg*, is relevant to queries 2 and 4
- *FH_toppOrg.jpg*, is relevant to queries 2and 4
- *FH_toppOrg800px.jpg* (is the copy of the *EXP_Nett_toppOrg800px.jpg*), is relevant to queries 2 and 4
- *p05002 400x276.jpg* relevant to queries 4,5,6,7,8,9,12 and 23
- *p05002 508x350.jpg* (is the copy of the *p05002 400x276.jpg*) and is relevant to queries 4,5,6,7,8,9,12 and 23

ImSE retrieves *che_858_1193500351.jpg* in query 28, because this image is referred to in the HTML document and thus is annotated by the keywords extracted from the context describing the image, i.e. football match.

ImSE does not retrieve the images *EXP_Nett_toppOrg800px.jpg* and *FH_toppOrg800px.jpg* in queries 2 and 4. These images are not used in any context and therefore are not annotated with keywords related to the image content. If one of those images were annotated, both would be retrieved by the system, because one is a copy of the other.

Images *p05002 508x350.jpg* and *p05002 400x276.jpg* retrieved to many queries, because image *p05002 400x276.jpg* is used in different contexts and thus is annotated with the keywords extracted from different contexts and is relevant to many queries. Image *p05002 508x350.jpg* is a copy and therefore also retrieved by ImSE.

7.5.5 Images used within different contexts

This subsection will discuss some of the images used within different contexts and how this influences retrieval performance for ImSE.

By different context we mean different HTMLs (as at illustrated in figure 5.11 in chapter 5.4). Unfortunately, the only three images are used within two different HTMLs:

- *p05002 400x276.jpg* relevant to queries 4,5,6,7,8,9,12 and 23
- *p05002 508x350.jpg* is copy of the *p05002 400x276.jpg* and is also relevant to queries 4,5,6,7,8,9,12 and 23
- *leonardo vite_aerea 250x161.jpg* is relevant to queries 4, 14 and 18.

Image *p05002 400x276.jpg* is referred to in two HTML documents, one is about sophists and Socrates other is about sophists, Agora and Athen. This image and its copy *p05002*

508x350.jpg are retrieved to the 6 of 8 queries: queries 4,6,7,8,9 and 12. If the image were referred to in only one HTML document, e.g., sophist and Socrates, it would be only retrieved to queries 2,4,9.

Image *leonardo vite_aerea 250x161.jpg* is also referred to in two HTML documents. In the first document the image is used as an illustration to the context that has nothing to do with the image content. In the second document the image is used to illustrate the subject of science. The image *leonardo vite_aerea 250x161.jpg* is retrieved to all queries it is relevant to. Actually, because this image was used as an illustration, it is relevant and retrieved by query 4, which is about illustrations.

Even if there are only few images used within different context in this test collection, the results show that gathering context for automatic annotation of images has merit for the system retrieval performance.

7.5.6 Evaluation of the system ranking

This sub-section will evaluate ranking of images by ImSE according to their degree of relevance as defined by the users.

In appendix B the sequence of the retrieved images is the same as ImSE's ranking of the images. The system ranks images according to their score value for the query (dot-product similarity measurement, see chapter 5.5 for more details). In general the system has a precise ranking order compared to the user level of the relevance degree.

Table 7.12 reports to how many relevant images ImSE retrieves within top 7 hits and its precision for each query in the test. The table shows results for the object and subject queries separately to give a better overview of the precision measurements for these two query categories.

From the results in table 7.12 we see that ImSE has perfect precision except for queries 13, 21, 26 and 27.

Object query number	**# hits within top 7**	**# relevant hits (1-4) within top 7**	**Precision within top 7**
6	6	6	1.0
7	6	6	1.0
9	7	7	1.0
10	7	7	1.0
12	6	6	1.0
13	4	3	0.75
15	1	1	1.0
16	2	2	1.0
17	3	3	1.0
19	5	4	0.8
20	5	5	1.0
21	7	4	0.57
24	1	1	1.0
25	2	2	1.0
26	7	5	0.71
27	7	6	0.86
29	3	3	1.0
Subject query number	**# hits within top 7**	**# relevant hits (1-4) within top 7**	**Precision within top 7**
1	7	6	0.86
2	7	3	0.43
3	7	7	1.0
4	7	5	0.71
5	7	3	0.43
8	5	5	1.0
11	3	3	1.0
14	7	1	0.14
18	5	5	1.0
22	7	7	1.0
23	7	6	0.86
28	3	3	1.0

Table 7.12: ImSE results for the top 7 hits.

If we study the retrieval table for the query 13, we see that 3 of 4 retrievals are ranked according to the users' degree of relevance.

If we study the retrieval tables for queries 21, 26 and 27 in appendix B, ImSE retrieves all relevant images to the queries (recall measures within top 7 for these queries are 1.0) and ranks them according to the users degree of relevance, accept for query 26.

When it comes to query 26 ("*Amerikanske presidenter*"), it retrieves two non-relevant images, which are about Bin Laden, where one is a copy of the other. ImSE assigns the images high score values and ranks them as numbers 3 and 4 in the retrieval set. If we study the context the original image was used within, we see that images of American president Bush and Bin Laden were used in the same context. It explains why images were annotated with the same context, retrieved by same query and has almost equally high scores to the query.

The results in table 7.12 for the subject queries show lower precision than for object queries. If wee study retrieval tables for the subject queries in appendix B, even if ImSE does not retrieve all relevant images to the queries, it ranks retrieved images according the users degree of relevance, except for queries 4, 5, 11, 14, 22 and 23.

Query 4: "*Illustrasjoner filosofihistorie*", is an ambiguously formulated query. It is difficult for ImSE to both retrieve images that the users will perceive as relevant and rank the images according to the user degree of relevance.

In query 5: "*Antikken*" the first two images (where one is a copy of the other) have no relevance to the query but are ranked as the top hits. One of these images is annotated with the keyword that matches the query keyword. The system assigns the highest score value for the image and ranks it at the top. Since the other image is a copy, it will be also assigned the same score value and also ranked at the top.

Retrieval sets to query 11: "*Hulelignelse, linjelignelse*" includes only relevant images. Ranking order for the query is not according to the users degree of relevance - ImSE ranks non-relevant images at the top.

When it comes to query 14: "*Renessansen, nyere tid*"– ImSE has a bad retrieval performance. This is the case where the query keywords "*nyere tid*" are dependent of each other, i.e. search for the keywords partially, "*nyere*" and "*tid*", has no meaning. As a result of this, ImSE retrieves 8 images where only one is relevant and ranked as number 7.

If we study the retrieval results for query 23:"*Photo, fotobilder, photobilder*" – we see that the none-relevant image has a top rank. If we look at the image filename, it contains the keyword "*photobilde*" that matches the query keyword and thus the image satisfies the search. This is the case where the users' understandings and perceptions of image content are different, i.e. the users who annotated the images and the users who retrieved the images are not in agreement in regards to its meaning/context.

Chapter 8

Conclusion and future work

This chapter gives some conclusions made in this project, summarizes results and evaluation for the prototype of the system implemented in the thesis. It also describes experience during the work and discusses some possible future work for the master project.

8.1 Conclusion

The goal and contribution of this master thesis has been to design and implement the image search engine, ImSE. The system analyzes the context images are referred in and uses it for automatic image annotation and search. If an image is used within different contexts, it is annotated with the keywords gathered from the different context. By different contexts we mean different textual documents that are different in their textual content. Users search for desired images by typing query keywords in the search field of the systems interface.

The test collection for this thesis contains the images and HTML documents images are referred in, which were created in Fronter VLE (Virtual Learning Environment) for the philosophy disciplines at the University of Tromsø.

The users for in this thesis are the teachers at the Department of Philosophy. They have developed the educational content used in my testing. For the purpose of the testing the teachers formulated queries and defined the sets with the relevant images for each query, which are used for the test and evaluation of ImSE.

To analyze what context might be relevant to the image content and its semantics, the users publishing skills and how images are used when developing educational content in Fronter have been studied.

The hypothesis one for this thesis was that context is useful in describing image content and its semantics. In general, ImSE has a good retrieval performance with average precision pre_2 = 0.80, recall rec_2 = 0.75 and $F\text{-}score_2$ = 0.77 for the test collection. There is a clear difference in retrieval performance for the object and subject queries (summarized in table 7.11 in chapter 7.4). Retrieval performance for the object queries, F-score measurement including a relevance degree of 4, is better in average of 26%. This is because retrieval for the subject queries is more difficult and relative, then for the object category. If object queries are about specific objects, e.g., geographical places, building or persons, subject queries are rather about events, issues or categories within history of philosophy and ethics in this test. Understanding and perception of the subject and what images are relevant to it highly depends on the users background such as knowledge, age, education, etc. Good examples of such queries and discussion of the retrieval results can be found in chapter 7.5 – Ambiguously formulated queries.

Another hypothesis for this thesis was that the more contexts image is used within the more the system will “learn” about its content and semantics. Gathering information from the multiple contexts and use it in image annotation will improve the system retrieval performance.

The test collection for this thesis is a copy of the real collection with images and documents that are used in teaching of the philosophy disciplines at the University, not an adjusted collection where use of images in different contexts are fixed. It has limitations when testing the hypothesis – there are only three images in the test collection, which are used within different documents. But even so it shows good results – images are retrieved to almost all queries about different subjects or specific places and persons they are relevant to.

Assume ImSE is integrated and used for image search in Fronter. The Fronter image collection contains thousands of images with no manual annotations and titles have not proper names, but these images are used within some contexts. Based on the results ImSE achieved in the tests for this thesis gives an indication that the retrieval performance of ImSE for such collections will be nearly as good as it is for the test collection. Comparing the retrieval performance of Solr and ImSE confirms that annotating images with the context or contexts images are used within is useful in describing image content and its semantics. Even with adjustments that were made in favor of Solr search performance in this thesis, ImSE has 45% better retrieval performance (F-score measurement).

8.2 General experience with the work

This section describes some experiences that were made by working with DeStore, Fronter and ImSE implementation.

Initially, it was planned to use DeStore more than just for the file storage. By using WebDAV interface it is possible to extract and add all kinds of information about the file objects stored in the system. It includes the meta-data such as author, date of last update, unique identifier of the file objects, etc., and other user-defined information, such as properties for the file objects in Fronter.

As mentioned previously, in this version of the system prototype, DeStore is used just as the file storage. It means in practice that any other storage could be used for the test collection of images and HTML documents.

Some experiences of how Solr uses textual information to search in Fronter were made. It seems that Solr treats a query as a string and searches by exactly matching the text available with the images. As it is observed, Fronter users spend little time giving images titles that can somehow describe the image content and its semantics. Neither do the users spend their time annotating images with text in the description fields. Thus Solr’s image retrieval performance is very poor. When users search for images in Fronter they have to know at least where the images are located in the Fronter archive.

During design and implementation of ImSE, an assumption was made that copies and crops of the images might be important to the system performance. Fronter, and the web in general, have a huge amount of image versions (copies and crops). It is not unusual that copies and crops are more often referred to in contexts than originals, because of their size customized for the publishing on the web. Or only special parts of the image regions might be of more interest than the image itself, e.g., part of a map, and thus originals are cropped. It is assumed that users are always interesting in retrieving originals rather than copies, because originals have more information per pixels then copies. As was described in the chapter 5.5, image copies and crops are handled manually in this version of the system prototype. ImSE retrieves original images with its copies and crops to the queries and these images count in measurements of the system retrieval performance. The disadvantage of such an approach is that when images are not relevant to the query and are included into the retrieval sets will necessarily decrease retrieval performances.

8.3 Future work

This section will discuss some possible future work.

Integration between user interface server and ImSE will be finished in the near future to be able to run query tests by typing keywords in the search field of the system interface.

When it comes to the image annotation, a number of other approaches could be tested to see if it might improve the system retrieval performance. For example, the disadvantage of using some particular parts of the context is that sometimes it contains information that has noting to do with the image content or/and its semantics, and will make image annotation "noisy". The examples of when the near text has nothing to do with the image were given in chapter 4.3 in the subsection about the user studies. One possible approach to avoid "noisy" annotation is to exclude near text as an annotation context if there is a text in the image caption field. It is based on the observation that text in the caption is more important in describing image than near text.

If an image is poorly annotated, for example only with the words extracted from the image filename and image URL, adoption of dictionaries or thesaurus' could be used to expand the image annotation. It might give a better description of the image and improve system retrieval performance.

Another approach that could be interesting to adopt and test in ImSE is differentiation of words appearing with the specific HTML tags as it is done in ImgeRover [15]. For example keywords appearing in the ALT and TITLE fields of the IMG tag will be assigned more weight than words appearing in the near text.

Approaches that allow for searching for phrases rather than partial word search could be also tested. It would at least improve the retrieval performance for the queries where there is strong dependency between the keywords, for example "*nyere tid*", "*be or not to be*" etc.

Chapter 5.4 mentions that image copies and crops are handled manually in the current version of the system. If image collection grows and contains thousands of images, handling copies and crops manually is not a good approach. Adoption of a content-based approach in this case is a reasonable solution. It could be implemented as its own mechanism/box that communicates with *Analyzer*. For example, for each new image received by *Analyzer* it will first check with the Content-based mechanism if the image is a copy or a crop to an already existing image in *Image db*. If so, the image URL to the copy or crop will be added to the XML file created for the original image, and annotations will be added to original image instead of the copy or crop. In this case there is no need to create an own XML files for copies and crops, and no need to store them or its thumbnails in *Image db* either. It will reduce size of image and xml databases, and make indexing and searching faster and more effective. It will also mean that original images will be annotated with all contexts its copies and crops are used within. The XML file for the image will schematically have the structure as it shown in figure 5.7, chapter 5.4.

Bibliography

[1] Abby A. Goodrum. Image Information Retrieval: An overview of Current Research. *Informing Science. Special Issue on Information Science Research*, vol.3, no. 2, 2000

[2] Arnold W.M. Smeulders, Marcel Worring, Simone Santini, Amarnath Gupta, Ramesh Jain. Content-Based Image Retrieval at the end of the Early Years. *IEEE Trans.Pattern Analysis And Machine Intelligence*, vol. 22, no. 12, pp. 1349-1379, December 2000

[3] Ritendra Datta, Dhiraj Joshi, Jia Li and James Z. Wang. Image Retrieval: Ideas, Influences, and Trends of the New Age. *ACM Computing Surveys*, vol. 40, no. 2, article 5, April 2008.

[4] Young Rui, Thomas S. Huang and Shih-Fu Chang. Image Retrieval: Past, Present, and Future. *in Proc. of Int. Symposium on Multimedia Information Processing*, 1997

[5] Nuno Vasconcelos. From Pixels to Semantic Spaces: Advances in Content-Based Image Retrieval. *Computer*, vol. 40, no. 7, pp. 20-26, July 2007

[6] M.L.Kherfi and D. Ziou, and A. Bernardi. Image Retrieval From the World Wide Web: Issues, Techniques, and Systems. *ACM Computing Surveys*, vol.36, no. 1, pp. 35-67, March 2004

[7] Myron Flickner, Harpreet Sawhney, Wayen Niblack, Jonathan Ashley, Qian Huang, Byron Dom, Monika Gorkani, Jim Hafner, Denis Lee, Dragutin Petkovic, David Steel and Peter Yanker: Query by Image and Video Content: The QBIC System. *Computer*, vol. 28, no. 9, pp. 23-32, September 1995

[8] John R. Smith, Shih-Fu Chang. VisualSEEK: a fully automated content-based image query system. In *Proc. of the fourth ACM international conference on Multimedia*, Boston, MA, 1996

[9] Thijs Westerveld. Image Retrieval: Content versus Context. In *Proc. on the conference on Content-Based Multimedia Information Access*, RIAO 2000, Paris, France, pp. 276-284, 2000

[10] Heng Tao Shen, Beng Chin Ooi, Kian-Lee Tan. Giving Meanings to WWW Images. In *Proc. of the eight ACM international conference on Multimedia*, Los Angeles CA, USA, 2000

[11] Masashi Inoue, Naonori Ueda. Retrieving Lightly Annotated Images using Image Similarities. In *Proc. of the 2005 ACM symposium on Applied computing*, Santa Fe, New Mexico, USA, March 2005

[12] Michael J. Swain, Charles Frankel, and Vassilis Athitsos. WebSEER: An Image Search Engine for the World Wide Web. *Submitted to IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR´97)*, Puerto Rico, June 1997

[13] Shih-Fu Chang, John R. Smith, Horace J. Meng. Exploring Image Functionalities in WWW Applications – Development of Image/Video Search and Editing Engines. *International Conference on Image Processing (ICIP'97)*, vol. 3, Washington DC, October 1997

[14] Y. Alp Aslandogan, Clement T. Yu. Diogenes: A Web Search Agent for Content Based Indexing of Personal Images. In *Proc. of ACM SIGIR*, 2000

[15] Marco La Cascia, Saratendu Sethi, Stan Sclaroff. Combining Textual and Visual Cues for Content-based Image Retrieval on the World Wide Web. In *Proc. of Content-Based Access of Image and Video Libraries*, *IEEE Workshop*, pp 24-28, Santa Barbara, CA, USA, Jun 1998

[16] Neil C. Rowe, Brian Frew. Automatic Caption Localization for Photographs on World Wide Web Pages. *Information Processing & Management*, vol. 34, no. 1, pp. 95-107, 1998

[17] Sissel H. Eriksen. Lecture in SVF-1050 at the University of Tromso, autumn 2008

[18] Liu Wenyin, Susan Dumais, Yanfeng Sun, Hong J. Zhang, Mary Czerwinski, Brent Field. Semi-Automatic Image Annotation. *In Proc. of Interact: Conference of HCI*, 2001

[19] Rong Zhao, William I. Grosky. Narrowing the Semantic Gap – Improved Text-Based Web Document Retrieval Using Visual Features. *IEEE Transactions on Multimedia*, vol. 4, no.2, June 2002

[20] Anind K. Dey and Gregory D. Abowd. Towards a Better Understanding of Context and Context-Awareness.

[21] Ricardo Baeza-Yates, Berthier Ribeiro-Neto. Modern Information Retrieval.

[22] Amit Singhal. Modern Information Retrieval: A brief Overview. *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, 2001.

[23] Marte Karidatter Skadem, Randi Karlsen, Njål T. Borch. Delight – A P2P Storage and Processing System. *Fourth International Conference on Internet and Web Applications and Services*, pp 97-101, Venice/Mestre, Jun 2009

[24] Njål T. Borch, Audun Heggelund. A read/write distributed web server.

Appendix A – List of queries

Queries are the user-defined for the test in this thesis. The users in this test are two teachers in the philosophy disciplines at University of Tromsø. They formulated queries based on the question “*What images do I want to search for if creating a new course in philosophy disciplines*?” In total 29 queries were formulated.

1. Krig og terror
2. Heading ex.phil, filosofihistorie
3. Fagansvarlige
4. Illustrasjoner filosofi
5. Antikken
6. Athen
7. Agora
8. Sofistene
9. Sokrates
10. Platon
11. Hulelignelsen, linjelignelsen
12. Skolen i Athen
13. Aristoteles
14. Renessansen, nyere tid
15. Bacon
16. Galilei (galileo)
17. Copernicus
18. Naturvitenskap
19. Descartes
20. Hume
21. Kant
22. Maleri, painting
23. Foto, photo, fotobilder, photobilder
24. Brueghel
25. Jacques Louis
26. Amerikanske presidenter
27. Jan Steen
28. fotball, football, fotballkamper
29. Peter Brueghelden

Appendix B – Tables with the evaluating results

This appendix reports the detailed retrieval results for ImSE and Solr for the test queries attached in appendix A. Each table corresponds to each query in the list. Recall and precision values are computed for the two sets. The first one, pre_1 and rec_1 is for the retrieval sets, which include the images with a relevance degree of 1-3, the second one, pre_2 and rec_2, is for the retrieval sets, which include the images with a relevance degree of 1-4.

R.D. is the abbreviation for the image degree of relevance.

R_1 and R_2 are the numbers of relevant images to the query in the whole image collection. R_1 corresponds to the images, which includes relevance degree of 1-3, and R_2 includes images with a relevance degree of 1-4.

A is the number of images in retrieval set.

Ra_1 is the number of relevant images in the retrieved image set, which includes images with a relevance degree of 1-3. Ra_2 is the number of relevant images in the retrieved image set, which includes images with a relevance degree of 1-4.

1. QUERY: Krig og terror (subject search)			
	ImSE		**Solr**
R.D.	**Rank**	**Retrieved images:**	**Corresponding retrieved images:**
1	1	1viet_napalm.jpg	Not retrieved
1	2	2abbas.jpg	Not retrieved
1	3	apokalypse.jpg	Not retrieved
1	4	1irak_gassangrep.jpg	Not retrieved
1	5	nurnbergdommene.jpg	Not retrieved
3	6	Goyafengsel2.jpg	Not retrieved
5	7	tiger_on_sofa.jpg	–
5	8	FE_toppOrg.jpg	–
2	9	bush.jpg *(copy)*	Not retrieved
2	10	bush 340x249.jpg	Not retrieved
2	11	binladen.jpg	Not retrieved
2	12	bin_laden x 180.jpg *(copy)*	Not retrieved
R.D.		**Relevant images that are not retrieved:**	**Corresponding relevant images that are not retrieved:**

2	–	flaggbrenning.jpg	Not retrieved	
3	–	goya_prison1.jpg (*copy*)	Not retrieved	
3	–	Goya.jpg	Not retrieved	
3	–	goya fengsel1.jpg	Not retrieved	
Evaluation metrics:			**Evaluation metrics:**	
R_1 = 14, Ra_1 = 10, A=12 pre_1 = 0.83 rec_1 = 0.71		R_2 = 14, Ra_2 = 10, A=12 pre_2 = 0.83 rec_2 = 0.71	R_1 = 14, Ra_1 = 0, A=0 pre_1 = 0.0 rec_1 = 0.0	R_2 = 14, Ra_2 = 0, A=0 pre_2 = 0.0 rec_2 = 0.0

2. QUERY: Heading ex.phil, filosofihistorie (subject search)			
ImSE			**Solr**
R.D.	**Rank**	**Retrieved mages:**	**Corresponding retrieved images:**
1	1	FH_toppOrg.jpg	Not retrieved
4	2	filosofihistorie_collage.jpeg	Not retrieved
4	3	filosofer i collage 378x226.jpg (*copy*)	Not retrieved
5	4	leonardo vite_aere 250x161 (col, tegning).jpg	–
5	5	Galileo 262x32 (wb, portrait).jpg	–
5	6	agora 400x268.jpg	–
5	7	p05002 400x276.jpg	–
5	8	p05002 508x350.jpg (*copy*)	–
5	9	Blaafjell 331x248.jpg (*copy*)	–
5	10	Blaafjell 721x541.jpg	–
5	11	Blaafjell 256x192 (*copy*)	–
5	12	tiger on sofa.jpg	–
5	13	ape_lead.jpg	–
4	14	sokrates_283x435.jpg	Not retrieved
4	15	Aristoteles 283x411 (col, sculpture).jpg	Not retrieved
5	16	Hume med turban 321x289(col, portrait).jpg (*copy*)	–
5	17	Hume med turban 659x825(col, portrait).jpg	–
5	18	Hume_med_turban_4.jpg (*copy*)	–
5	19	Akademiet i Athen 590x600.jpg	–
5	20	platons akademi.jpg (*copy*)	–
5	21	FE_toppOrg.jpg	–

5	22	Skolen i Athen 721x482.jpg	–
5	23	Skolen i Aten 450x301.jpg (*copy*)	–
4	24	Platon 238x326 (wb, sculpture).jpg	Not retrieved
5	25	soeyler.jpg	–
R.D.		**Relevant images that are not retrieved:**	**Corresponding relevant images that are not retrieved:**
1		EXP_Nett_toppOrg800px.jpg 1	Not retrieved
1		FH_toppOrg800px.jpg (*copy*)	Not retrieved
Evaluation metrics:		**Evaluation metrics:**	
R_1 = 3, Ra_1 = 1, A=25 pre_1 = 0.04 rec_1 = 0.33	R_2 = 8, Ra_2 = 6, A=25 pre_2 = 0.24 rec_2 = 0.75	R_1 = 3, Ra_1 = 0, A=0 pre_1 = 0.0 rec_1 = 0.0	R_2 = 8, Ra_2 = 0, A=0 pre_2 = 0.0 rec_2 = 0.0

3. QUERY: fagansvarlige (subject search)			
ImSE			**Solr**
R.D.	**Rank**	**Retrieved images:**	**Corresponding retrieved images:**
1	1	roar.jpg	roar.jpg
1	2	roar_130x113.jpg (*copy*)	roar_130x113.jpg
1	3	oyvind2.jpg	oyvind2.jpg
1	4	Overrein.jpg	Overrein.jpg
1	5	jonas_113x130.jpg (*copy*)	jonas_113x130.jpg
1	6	jonas2.jpg	jonas2.jpg
1	7	oyvind.jpg	oyvind.jpg
1	8	oyvind130x113.jpg (*copy*)	oyvind130x113.jpg
1	9	oeyvind v2.jpg (*copy*)	oeyvind v2.jpg
1	10	oeyvind03.jpg (*copy)*	oeyvind03.jpg
1	11	Omtvedt2.jpg	Omtvedt2.jpg
1	12	baard 98x127.jpg	baard 98x127.jpg
1	13	Arne v9.jpg (*copy*)	Arne v9.jpg
1	14	asejohansenx.jpg	asejohansenx.jpg
1	15	Aase03.jpg	Aase03.jpg
1	16	aase 98x127.jpg (*copy*)	aase 98x127.jpg
1	17	Aase130x.jpg (*copy*)	Aase130x.jpg
1	18	paal_130x113.jpg (*copy*)	paal_130x113.jpg
1	19	paal.jpg	paal.jpg
5	20	Lisa.jpg	Lisa.jpg
1	21	mariann.jpg	mariann.jpg

1	22	mariann_130x113.jpg (*copy*)	mariann_130x113.jpg
1	23	Vibeke130x.jpg (*copy*)	Vibeke130x.jpg
1	24	Vibeke01.jpg	Vibeke01.jpg
1	25	Elin_2_130x113.jpg	Elin_2_130x113.jpg
1	26	Elin130x113.jpg	Elin130x113.jpg
R.D.		**Relevant images that are not retrieved:**	**Corresponding relevant images that are not retrieved:**

Evaluation metrics:		**Evaluation metrics:**	
R_1 = 25, Ra_1 = 25, A=26 pre_1 = 0.96 rec_1 = 1.0	R_2 = 25, Ra_2 = 25, A=26 pre_2 = 0.96 rec_2 = 1.0	R_1 = 25, Ra_1 = 25, A=26 pre_1 = 0.96 rec_1 = 1.0	R_2 = 25, Ra_2 = 25, A=26 pre_2 = 0.96 rec_2 = 1.0

4. QUERY: Illustrasjoner filosofihistorie (subject search)			
ImSE			**Solr**
R.D.	**Rank**	**Retrieved images:**	**Corresponding retrieved images:**
1	1	agora 400x268.jpg	Not retrieved
1	2	p05002 400x276.jpg(*copy*)	Not retrieved
1	3	p05002 508x350.jpg	Not retrieved
5	4	etikk_uke3x400.jpg	–
5	5	etikk_uke3x200x253.jpg (*copy*)	–
1	6	filosofihistorie_collage.jpeg	Not retrieved
1	7	filosofer i collage 378x226.jpg (*copy*)	Not retrieved
1	8	Skolen i Athen 721x482.jpg	Not retrieved
1	9	Skolen i Aten 450x301.jpg (*copy*)	Not retrieved
2	10	soeyler.jpg	Not retrieved
3	11	hesiod.jpeg	Not retrieved
4	12	Aristoteles 283x411 (col, sculpture).jpg	Not retrieved
5	13	Hume med turban 321x289(col, portrait).jpg (*copy*)	–
5	14	Hume med turban 659x825(col, portrait).jpg	–
5	15	Hume_med_turban_4.jpg (*copy*)	–
1	16	platons akademi.jpg	Not retrieved
1	17	Akademiet i Athen 590x600.jpg(*copy*)	Not retrieved
3	18	Platon 238x326 (wb, sculpture).jpg	Not retrieved

R.D.		**Relevant images that are not retrieved:**	**Corresponding relevant images that are not retrieved:**	
1		Ex.Phil.Collage .jpg 591x354	Not retrieved	
1		00 topside_h06.jpg	Not retrieved	
1		sokrates death 500x325.jpg	Not retrieved	
1		sokrates_death 350x.jpg (*copy*)	Not retrieved	
1		platon_aristotle 263x336.jpg	Not retrieved	
1		Picture 025 236x253.jpg	Not retrieved	
1		Agora ancient 640x480.jpg	Not retrieved	
1		Agora ancient 350x260.jpg (*copy*)	Not retrieved	
1		agora_plan 500x323.jpg	Not retrieved	
3		Blaafjell 721x541.jpg	Not retrieved	
3		Blaafjell 331x248.jpg (*copy*)	Not retrieved	
3		Blaafjell 256x192.jpg (*copy*)	Not retrieved	
1		Copernicus 558x39.jpg	Not retrieved	
3		EXP_Nett_toppOrg800px.jpg	Not retrieved	
3		FH_toppOrg800px.jog (*copy*)	Not retrieved	
3		FH_toppOrg.jpg	Not retrieved	
1		leonardo vite_aerea 250x161.jpg	Not retrieved	
3		Platon 231x326.jpg	Not retrieved	
3		Platon 268x326.jpg	Not retrieved	
1		sokrates offside.jpg	Not retrieved	
Evaluation metrics:			**Evaluation metrics:**	
R_1 = 32, Ra_1 = 12, A=18 pre_1 = 0.67 rec_1 = 0.34		R_2 = 33, Ra_2 = 13, A=18 pre_2 = 0.72 rec_2 = 0.39	R_1 = 32, Ra_1 = 0, A=0 pre_1 = 0.0 rec_1 = 0.0	R_2 = 33, Ra_2 = 0, A=0 pre_2 = 0.0 rec_2 = 0.0

5. QUERY: Antikken (subject search)			
ImSE			**Solr**
R.D.	**Rank**	**Retrieved images:**	**Corresponding retrieved images:**
5	1	etikk_uke3x400.jpg	–
5	2	etikk_uke3x200x253.jpg (*copy*)	–
1	3	hesiod.jpeg	Not retrieved
1	4	socrates death 500x325.jpg	Not retrieved
1	5	socrates_death 350x.jpg (*copy*)	Not retrieved
5	6	1viet_napalm.jpg	–
5	7	2abbas.jpg	–

5	8	1irak gassangrep.jpg	–
R.D.		**Relevant images that are not retrieved:**	**Corresponding relevant images that are not retrieved:**
1		Skolen i Athen 721x482.jpg	Not retrieved
1		Skolen i Athen 450x301.jpg (*copy*)	Not retrieved
1		Platon_Aristotle 263x336.jpg	Not retrieved
1		agora 400x268.jpg	Not retrieved
1		Agora ancient 640x480.jpg	Not retrieved
1		Agora ancient 350x260.jpg (*copy*)	Not retrieved
1		agora_plan 500x323.jpg	Not retrieved
1		p05002 508x350.jpg	Not retrieved
1		p05002 400x276.jpg (*copy*)	Not retrieved
1		Akademiet i Athen 590x600.jpg	Not retrieved
1		platons_akademi.jpg (*copy*)	Not retrieved
1		sokrates offside.jpg	Not retrieved
1		sokrates_283x435.jpg	Not retrieved
1		Platon_Aristotle 263x336.jpg	Not retrieved
1		Platon 231x326.jpg	Not retrieved
1		Platon 238x326.jpg	Not retrieved
1		Platon 268x326.jpg	Not retrieved
1		Aristoteles 283x411.jpg	Not retrieved
Evaluation metrics:		**Evaluation metrics:**	
$R_1 = 21$, $Ra_1 = 3$, $A=8$ $pre_1 = 0.38$ $rec_1 = 0.14$	$R_2 = 21$, $Ra_2 = 3$, $A=8$ $pre_2 = 0.38$ $rec_2 = 0.14$	$R_1 = 21$, $Ra_1 = 0$, $A=0$ $pre_1 = 0.0$ $rec_1 = 0.0$	$R_2 = 21$, $Ra_2 = 0$, $A=0$ $pre_2 = 0.0$ $rec_2 = 0.0$

6. QUERY: Athen (concrete place)			
ImSE			**Solr**
R.D.	**Rank**	**Retrieved images:**	**Corresponding retrieved images:**
1	1	Skolen i Athen 721x482.jpg	Skolen i Athen 721x482.jpg
1	2	Skolen i Aten 450x301.jpg (*copy*)	Not retrieved
1	3	p05002 508x350.jpg	Not retrieved
1	4	p05002 400x276.jpg (*copy*)	Not retrieved
1	5	agora_plan 500x323.jpg	Not retrieved
1	6	soeyler.jpg	Not retrieved
1	7	Akademiet i Athen 590x600.jpg	Akademiet i Athen 590x600.jpg
1	8	platons_akademi.jpg (*copy*)	Not retrieved

R.D.		Relevant images that are not retrieved:	Corresponding relevant images that are not retrieved:	
1		agora 400x268.jpg	Not retrieved	
1		Agora ancient 640x480.jpg	Not retrieved	
1		Agora ancient 350x260.jpg (*copy*)	Not retrieved	
Evaluation metrics:			Evaluation metrics:	
R_1 = 11, Ra_1 = 8, A=8 pre_1 = 1.0 rec_1 = 0.73		R_2 = 11, Ra_2 = 8, A=8 pre_2 = 1.0 rec_2 = 0.73	R_1 = 11, Ra_1 = 2, A=2 pre_1 = 1.0 rec_1 = 0.18	R_2 = 11, Ra_2 = 2, A=2 pre_2 = 1.0 rec_2 = 0.18

7. QUERY: Agora (concrete place)				
ImSE			Solr	
R.D.	Rank	Retrieved images:	Corresponding retrieved images:	
1	1	Agora ancient 640x480.jpg	Agora ancient 640x480.jpg	
1	2	Agora ancient 350x260.jpg (*copy*)	Agora ancient 350x260.jpg	
1	3	agora_plan 500x323.jpg	agora_plan 500x323.jpg	
1	4	p05002 508x350.jpg	Not retrieved	
1	5	p05002 400x276.jpg (*copy*)	Not retrieved	
1	6	agora 400x268.jpg	agora 400x268.jpg	
R.D.		Relevant images that are not retrieved:	Corresponding relevant images that are not retrieved:	
Evaluation metrics:			Evaluation metrics:	
R_1 = 6, Ra_1 = 6, A=6 pre_1 = 1.0 rec_1 = 1.0		R_2 = 6, Ra_2 = 6, A=6 pre_2 = 1.0 rec_2 = 1.0	R_1 = 6, Ra_1 = 4,A=4 pre_1 = 1.0 rec_1 = 0.67	R_2 = 6, Ra_2 = 4, A=4 pre_2 = 1.0 rec_2 = 0.67

8. QUERY: Sofistene (subject search)			
ImSE			Solr
R.D.	Rank	Retrieved images:	Corresponding retrieved images:
1	1	p05002 508x350.jpg	Not retrieved
1	2	p05002 400x276.jpg (*copy*)	Not retrieved
1	3	agora_plan 500x323.jpg	Not retrieved
1	4	agora 400x268.jpg	Not retrieved
4	5	hesoid.jpg	Not retrieved
R.D.		Relevant images that are not retrieved:	Corresponding relevant images that are not retrieved:

1		Agora ancient 640x480.jpg	Not retrieved
1		Agora ancient 350x260.jpg (*copy*)	Not retrieved
Evaluation metrics:			**Evaluation metrics:**
R_1 = 6, Ra_1 = 4, A=5 pre_1 = 0.8 rec_1 = 0.67		R_2 = 7, Ra_2 = 5, A=5 pre_2 = 1.0 rec_2 = 0.71	R_1 = 6, Ra_1 = 0, A=0 pre_1 = 0.0 rec_1 = 0.0 R_2 = 6, Ra_2 = 0, A=0 pre_2 = 0.0 rec_2 = 0.0

9. QUERY: Sokrates (concrete person)			
ImSE			**Solr**
R.D	**Rank**	**Retrieved images:**	**Corresponding retrieved images:**
1	1	socrates death 500x325.jpg	Not retrieved
1	2	sokrates_death 350x.jpg (*copy*)	sokrates_death 350x.jpg
1	3	sokrates offside.jpg	sokrates offside.jpg
1	4	sokrates_283x435.jpg	sokrates_283x435.jpg
3	5	agora 400x268.jpg	Not retrieved
3	6	p05002 508x350.jpg	Not retrieved
3	7	p05002 400x276.jpg (*copy*)	Not retrieved
5	8	Blaafjell 721x541.jpg	–
5	9	Blaafjell 331x248.jpg (*copy)*	–
5	10	Blaafjell 256x192.jpg (*copy)*	–
5	11	binladen.jpg	–
5	12	bin_laden x 180.jpg (*copy)*	–
5	13	bush.jpg	–
5	14	bush 340x249.jpg (*copy*)	–
R.D.		**Relevant images that are not retrieved:**	**Corresponding relevant images that are not retrieved:**
3		agora_plan 500x323.jpg	Not retrieved
3		Agora ancient 640x480.jpg	Not retrieved
3		Agora ancient 350x260.jpg (*copy*)	Not retrieved
Evaluation metrics:			**Evaluation metrics:**
R_1 = 10, Ra_1 = 7, A=12 pre_1 = 0.58 rec_1 = 0.70		R_2 = 10, Ra_2 = 7, A=12 pre_2 = 0.58 rec_2 = 0.70	R_1 = 10, Ra_1 = 3, A=3 pre_1 = 1.0 rec_1 = 0.33 R_2 = 10, Ra_2 = 3, A=3 pre_2 = 1.0 rec_2 = 0.33

10. QUERY: Platon (concrete person)			
ImSE			**Solr**
R.D.	**Rank**	**Retrieved images:**	**Corresponding retrieved images:**
1	1	Platon 268x326 (wb, sculpture).jpg	Platon 268x326 (wb, sculpture).jpg
1	2	Platon 231x326 (wb, sculpture).jpg	Platon 231x326 (wb, sculpture).jpg
1	3	Platon 238x326 (wb, sculpture).jpg	Platon 238x326 (wb, sculpture).jpg
2	4	Platon_Aristotle 263x336 (col, painting).jpg	Platon_Aristotle263x 336 (col, painting).jpg
2	5	Akademiet i Athen 590x600.jpg	Not retrieved
2	6	platons_akademi.jpg (*copy*)	Not retrieved
4	7	Skolen i Aten 721x482.jpg	Not retrieved
4	8	Skolen i Aten 450x301.jpg (copy)	Not retrieved
5	9	soeyler.jpg	–
5	10	socrates death 500x325.jpg	–
5	11	socrates death 350x.jpg (*copy*)	–
5	12	Blaafjell 721x541.jpg	–
5	13	Blaafjell 331x248.jpg (*copy*)	–
5	14	Blaafjell 256x192.jpg (*copy*)	–
R.D.		**Relevant images that are not retrieved:**	**Corresponding relevant images that are not retrieved:**
Evaluation metrics:			**Evaluation metrics:**

ImSE		Solr	
$R_1 = 6$, $Ra_1 = 6$, A=14 $pre_1 = 0.43$ $rec_1 = 1.0$	$R_2 = 8$, $Ra_2 = 8$, A=14 $pre_2 = 0.57$ $rec_2 = 1.0$	$R_1 = 6$, $Ra_1 = 4$, A=4 $pre_1 = 1.0$ $rec_1 = 0.67$	$R_2 = 8$, $Ra_2 = 4$, A=4 $pre_2 = 1.0$ $rec_2 = 0.50$

11. QUERY: Hulelignelsen, linjelignelsen (subject search)			
ImSE			**Solr**
R.D.	**Rank**	**Retrieved images:**	**Corresponding retrieved images:**
4	1	Akademiet i Athen 590x600.jpg	–
4	2	platons_akademi.jpg (*copy*)	–
3	3	Platon 238x326 (wb, sculpture).jpg	Not retrieved
R.D.		**Relevant images that are not retrieved:**	**Corresponding relevant images that are not retrieved:**
3		Platon 231x326.jpg	Not retrieved
3		Platon 268x326.jpg	Not retrieved
Evaluation metrics:			**Evaluation metrics:**

R_1 = 3, Ra_1 = 1, A=3 pre_1 = 0.33 rec_1 = 0.33	R_2 = 5, Ra_2 = 3, A=3 pre_2 = 0.60 rec_2 = 1.0	R_1 = 3, Ra_1 = 0, A=0 pre_1 = 0.0 rec_1 = 0.0	R_2 = 5, Ra_2 = 0, A=0 pre_2 = 0.0 rec_2 = 0.0

12. QUERY: Skolen i Athen (concrete place)

ImSE			Solr
R.D.	**Rank**	**Retrieved images:**	**Corresponding retrieved images:**
1	1	Skolen i Aten 721x482.jpg	Not retrieved
1	2	Skolen i Aten 450x301.jpg (copy)	Not retrieved
3	3	agora_plan 500x323.jpg	Skolen i Aten 721x482.jpg
3	4	p05002 508x350.jpg	Not retrieved
3	5	p05002 400x276.jpg (*copy*)	Not retrieved
3	6	Akademiet i Athen 590x600.jpg	Not retrieved
3	7	platons_akademi.jpg (*copy*)	Not retrieved
3	8	soeyler.jpg	Not retrieved
R.D.		**Relevant images that are not retrieved:**	**Corresponding relevant images that are not retrieved:**
3		agora 400x268.jpg	Not retrieved
3		Agora ancient 640x480.jpg	Not retrieved
3		Agora ancient 350x260.jpg (*copy*)	Not retrieved

Evaluation metrics:		**Evaluation metrics:**	
R_1 = 11, Ra_1 = 8, A=8 pre_1 = 1.0 rec_1 = 0.73	R_2 = 11, Ra_2 = 8, A=8 pre_2 = 1.0 rec_2 = 0.73	R_1 = 11, Ra_1 = 1, A=1 pre_1 = 1.0 rec_1 = 0.09	R_2 = 11, Ra_2 = 1, A=1 pre_2 = 1.0 rec_2 = 0.09

13. QUERY: Aristoteles (concrete person)

ImSE			Solr
R.D.	**Rank**	**Retrieved images:**	**Corresponding retrieved images:**
1	1	Aristoteles 283x411 (col, sculpture).jpg	Aristoteles 283x411 (col, sculpture).jpg
3	2	Skolen i Aten 721x482.jpg	Not retrieved
3	3	Skolen i Aten 450x301.jpg (copy)	Not retrieved
5	4	soeyler.jpg	–
R.D.		**Relevant images that are not retrieved:**	**Corresponding relevant images that are not retrieved:**
3		Platon_Aristotle 263x336.jpg	Not retrieved

Evaluation metrics:		Evaluation metrics:	
$R_1 = 4$, $Ra_1 = 3$, $A=4$ $pre_1 = 0.75$ $rec_1 = 0.75$	$R_2 = 4$, $Ra_2 = 3$, $A=4$ $pre_2 = 0.75$ $rec_2 = 0.75$	$R_1 = 4$, $Ra_1 = 1$, $A=1$ $pre_1 = 1.0$ $rec_1 = 0.25$	$R_2 = 4$, $Ra_2 = 1$, $A=1$ $pre_2 = 1.0$ $rec_2 = 0.25$

14. QUERY: Renessansen, nyere tid (subject search)			
ImSE			**Solr**
R.D.	**Rank**	**Retrieved images:**	**Corresponding retrieved images:**
5	1	agora_plan 500x323.jpg	–
5	2	p05002 508x350.jpg	–
5	3	p05002 400x276.jpg (*copy*)	–
5	4	etikk_uke6_2x700.jpg	–
5	5	Bryllup 700x525.jpg (*copy*)	–
5	6	Bryllup 300x225.jpg (*copy*)	–
1	7	leonardo vite_aerea 250x161	Not retrieved
5	8	bootboys.jpg	–
R.D.		**Relevant images that are not retrieved:**	**Corresponding relevant images that are not retrieved:**
1		Galileo 80x80.jpg	Not retrieved
1		Galileo 262x326.jpg (*copy*)	Not retrieved
1		Kepler 93x124.jpg	Not retrieved
1		Newton 280x303.jpg	Not retrieved
1		Descartes 283x400.jpg	Not retrieved
1		Descartes 584x800.jpg	Not retrieved
1		Descartes 240x330.jpg (*copy*)	Not retrieved
1		DESCARTES.jpg	Not retrieved
1		Copernicus 268x326.jpg	Not retrieved
1		Copernicus 280x400.jpg	Not retrieved
1		Copernicus 558x398.jpg	Not retrieved
1		Fransis Bacon 191x225.jpg	Not retrieved
1		Harvey William.jpg 255x303	Not retrieved
1		Robert Boyle 190x225.jpg	Not retrieved

Evaluation metrics:		Evaluation metrics:	
$R_1 = 15$, $Ra_1 =1$, $A=8$ $pre_1 = 0.13$ $rec_1 = 0.07$	$R_2 = 15$, $Ra_2 = 1$, $A=8$ $pre_2 = 0.13$ $rec_2 = 0.07$	$R_1 = 15$, Ra_1,$A = 0$, $pre_1 = 0.0$ $rec_1 = 0.0$	$R_2 = 15$, Ra_2, $A= 0$, $pre_2 = 0.0$ $rec_2 = 0.0$

15. QUERY: Bacon (concrete person)			
ImSE			**Solr**
R.D.	**Rank**	**Retrieved images:**	**Image:**
1	1	Francis Bacon 191x225 (col, portrait).jpg	Francis Bacon 191x225 (col, portrait).jpg
R.D.		**Relevant images that are not retrieved:**	**Corresponding relevant images that are not retrieved:**

Evaluation metrics:		**Evaluation metrics:**	
R_1 = 1, Ra_1 = 1, A=1 pre_1 = 1.0 rec_1 = 1.0	R_2 = 1, Ra_2 = 1, A=1 pre_2 = 1.0 rec_2 = 1.0	R_1 = 1, Ra_1 = 1, A=1 pre_1 = 1.0 rec_1 = 1.0	R_2 = 1, Ra_2 = 1, A=1 pre_2 = 1.0 rec_2 = 1.0

16. QUERY: Galileo (concrete person)			
ImSE			**Solr**
R.D.	**Rank**	**Retrieved images:**	**Corresponding retrieved images:**
1	1	Galileo 80x80 (wb, portrait).jpg	Galileo 80x80 (wb, portrait).jpg
1	2	Galileo 262x326 (wb, portrait).jpg	Galileo 262x326 (wb, portrait).jpg
R.D.		**Relevant images that are not retrieved:**	**Corresponding relevant images that are not retrieved:**

Evaluation metrics:		**Evaluation metrics:**	
R_1 = 2, Ra_1 = 2, A=2 pre_1 = 1.0 rec_1 = 1.0	R_2 = 2, Ra_2 = 2, A=2 pre_2 = 1.0 rec_2 = 1.0	R_1 = 2, Ra_1 = 2, A=2 pre_1 = 1.0 rec_1 = 1.0	R_2 = 2, Ra_2 = 2, A=2 pre_2 = 1.0 rec_2 = 1.0

17. QUERY: Copeernicus (concrete person)			
ImSE			**Solr**
R.D.	**Rank**	**Retrieved images:**	**Corresponding retrieved image:**
1	1	Copernicus 268x326 (wb, portrait).jpg	Copernicus 268x326 (wb, portrait).jpg
1	2	Copernicus 280x400 (col, painting).jpg	Copernicus 280x400 (col, painting).jpg
1	3	Copernicus 558x398 (col, painting).jpg	Copernicus 558x398 (col, painting).jpg

<table>
<tr><td>R.D.</td><td></td><td colspan="2">Relevant images that are not retrieved:</td><td colspan="2">Corresponding relevant images that are not retrieved:</td></tr>
<tr><td colspan="4"></td><td colspan="2"></td></tr>
<tr><td colspan="4">Evaluation metrics:</td><td colspan="2">Evaluation metrics:</td></tr>
<tr><td colspan="3">$R_1 = 3$, $Ra_1 = 3$, A=3
$pre_1 = 1.0$
$rec_1 = 1.0$</td><td>$R_2 = 3$, $Ra_2 = 3$, A=3
$pre_2 = 1.0$
$rec_2 = 1.0$</td><td>$R_1 = 3$, $Ra_1 = 3$,
A=3
$pre_1 = 1.0$
$rec_1 = 1.0$</td><td>$R_2 = 3$, $Ra_2 = 3$,
A=3
$pre_2 = 1.0$
$rec_2 = 1.0$</td></tr>
</table>

<table>
<tr><td colspan="6">18. QUERY: Naturvitenskap (subject searh)</td></tr>
<tr><td colspan="4">ImSE</td><td colspan="2">Solr</td></tr>
<tr><td>R.D.</td><td>Rank</td><td colspan="2">Retrieved images:</td><td colspan="2">Corresponding retrieved images:</td></tr>
<tr><td>1</td><td>1</td><td colspan="2">Galileo 262x326 (wb, portrait).jpg</td><td colspan="2">Not retrieved</td></tr>
<tr><td>1</td><td>2</td><td colspan="2">leonardo_vite_aerea250x161.jpg</td><td colspan="2">Not retrieved</td></tr>
<tr><td>4</td><td>3</td><td colspan="2">Skolen i Aten 721x482.jpg</td><td colspan="2">–</td></tr>
<tr><td>4</td><td>4</td><td colspan="2">Skolen i Aten 450x301.jpg (copy)</td><td colspan="2">–</td></tr>
<tr><td>5</td><td>5</td><td colspan="2">Soyler.jpg</td><td colspan="2">–</td></tr>
<tr><td colspan="4"></td><td colspan="2"></td></tr>
<tr><td>R.D.</td><td></td><td colspan="2">Relevant images that are not retrieved:</td><td colspan="2">Corresponding relevant images that are not retrieved:</td></tr>
<tr><td>1</td><td></td><td colspan="2">Galileo 80x80.jpg</td><td colspan="2">Not retrieved</td></tr>
<tr><td>1</td><td></td><td colspan="2">Kepler 93x124.jpg</td><td colspan="2">Not retrieved</td></tr>
<tr><td>1</td><td></td><td colspan="2">Newton 280x303.jpg</td><td colspan="2">Not retrieved</td></tr>
<tr><td>1</td><td></td><td colspan="2">Descartes 240x330.jpg (copy)</td><td colspan="2">Not retrieved</td></tr>
<tr><td>1</td><td></td><td colspan="2">Descartes 584x800 (col,portrait).jpg</td><td colspan="2">Not retrieved</td></tr>
<tr><td>1</td><td></td><td colspan="2">Descartes 382x400 (col, portrait).jpg</td><td colspan="2">Not retrieved</td></tr>
<tr><td>1</td><td></td><td colspan="2">DESCARTES.jpg</td><td colspan="2">Not retrieved</td></tr>
<tr><td>1</td><td></td><td colspan="2">Copernicus 268x326.jpg</td><td colspan="2">Not retrieved</td></tr>
<tr><td>1</td><td></td><td colspan="2">Copernicus 280x400.jpg</td><td colspan="2">Not retrieved</td></tr>
<tr><td>1</td><td></td><td colspan="2">Copernicus 558x398.jpg</td><td colspan="2">Not retrieved</td></tr>
<tr><td>1</td><td></td><td colspan="2">Fransis Bacon 191x225.jpg</td><td colspan="2">Not retrieved</td></tr>
<tr><td>1</td><td></td><td colspan="2">Harvey William 255x303.jpg</td><td colspan="2">Not retrieved</td></tr>
<tr><td>1</td><td></td><td colspan="2">Robert Boyle 190x225.jpg</td><td colspan="2">Not retrieved</td></tr>
<tr><td colspan="4"></td><td colspan="2"></td></tr>
<tr><td colspan="4">Evaluation metrics:</td><td colspan="2">Evaluation metrics:</td></tr>
<tr><td colspan="3">$R_1 = 15$, $Ra_1 = 2$, A=5
$pre_1 = 0.40$
$rec_1 = 0.13$</td><td>$R_2 = 17$, $Ra_2 = 4$, A=5
$pre_2 = 0.80$
$rec_2 = 0.24$</td><td>$R_1 = 15$, $Ra_1 = 0$,
A=0
$pre_1 = 0.0$
$rec_1 = 0.0$</td><td>$R_2 = 17$, $Ra_2 = 0$,
A=0
$pre_2 = 0.0$
$rec_2 = 0.0$</td></tr>
</table>

<table>
<tr><th colspan="5">19. QUERY: Descartes (concrete person)</th></tr>
<tr><th colspan="3">ImSE</th><th colspan="2">Solr</th></tr>
<tr><td>R.D.</td><td>Rank</td><td>Retrieved images:</td><td colspan="2">Corresponding retrieved mages:</td></tr>
<tr><td>1</td><td>1</td><td>Descartes 240x330.jpg (copy)</td><td colspan="2">Descartes 240x330.jpg (copy)</td></tr>
<tr><td>1</td><td>2</td><td>Descartes 584x800 (col,portrait).jpg</td><td colspan="2">Descartes 584x800 (col,portrait).jpg</td></tr>
<tr><td>1</td><td>3</td><td>Descartes 382x400 (col, portrait).jpg</td><td colspan="2">Descartes 382x400 (col, portrait).jpg</td></tr>
<tr><td>1</td><td>4</td><td>DESCARTES.jpg</td><td colspan="2">DESCARTES.jpg</td></tr>
<tr><td>5</td><td>5</td><td>Galileo 262x326 (wb, portrait).jpg</td><td colspan="2">–</td></tr>
<tr><td>5</td><td>6</td><td>leonardo_vite_aerea250x161.jpg</td><td colspan="2">–</td></tr>
<tr><td colspan="3"></td><td colspan="2"></td></tr>
<tr><td>R.D.</td><td></td><td>Relevant images that are not retrieved:</td><td colspan="2">Corresponding relevant images that are not retrieved:</td></tr>
<tr><td colspan="3"></td><td colspan="2"></td></tr>
<tr><td colspan="3">Evaluation metrics:</td><td colspan="2">Evaluation metrics:</td></tr>
<tr><td colspan="2">$R_1 = 4$, $Ra_1 = 6$, $A=4$
$pre_1 = 0.67$
$rec_1 = 1.0$</td><td>$R_2 = 4$, $Ra_2 = 6$, $A=4$
$pre_2 = 0.67$
$rec_2 = 1.0$</td><td>$R_1 = 4$, $Ra_1 = 4$, $A=4$
$pre_1 = 1.0$
$rec_1 = 1.0$</td><td>$R_2 = 4$, $Ra_2 = 4$, $A=4$
$pre_2 = 1.0$
$rec_2 = 1.0$</td></tr>
</table>

<table>
<tr><th colspan="5">20. QUERY: Hume (concrete person)</th></tr>
<tr><th colspan="3">ImSE</th><th colspan="2">Solr</th></tr>
<tr><td>R.D.</td><td>Rank</td><td>Retrieved images:</td><td colspan="2">Corresponding retrieved images:</td></tr>
<tr><td>1</td><td>1</td><td>Hume 718x870 (col, portrait).jpg</td><td colspan="2">Hume 718x870 (col, portrait).jpg</td></tr>
<tr><td>1</td><td>2</td><td>Hume 330x400 (col, portrait).jpg (copy)</td><td colspan="2">Hume 330x400 (col, portrait).jpg</td></tr>
<tr><td>1</td><td>3</td><td>hume med turban 659x825 (col, portrait).jpg</td><td colspan="2">hume med turban 659x825 (col, portrait).jpg</td></tr>
<tr><td>1</td><td>4</td><td>Hume med turban 321x289 (col, portrait).jpg (copy)</td><td colspan="2">Hume med turban 321x289 (col, portrait).jpg</td></tr>
<tr><td>1</td><td>5</td><td>Hume_med_turban_4.jpg</td><td colspan="2">Hume_med_turban_4.jpg</td></tr>
<tr><td colspan="3"></td><td colspan="2"></td></tr>
<tr><td>R.D.</td><td></td><td>Relevant images that are not retrieved:</td><td colspan="2">Corresponding relevant images that are not retrieved:</td></tr>
<tr><td colspan="3"></td><td colspan="2"></td></tr>
<tr><td colspan="3">Evaluation metrics:</td><td colspan="2">Evaluation metrics:</td></tr>
<tr><td colspan="2">$R_1 = 5$, $Ra_1 = 5$, $A=5$
$pre_1 = 1.0$
$rec_1 = 1.0$</td><td>$R_2 = 5$, $Ra_2 = 5$, $A=5$
$pre_2 = 1.0$
$rec_2 = 1.0$</td><td>$R_1 = 5$, $Ra_1 = 5$, $A=5$
$pre_1 = 1.0$
$rec_1 = 1.0$</td><td>$R_2 = 5$, $Ra_2 = 5$, $A=5$
$pre_2 = 1.0$
$rec_2 = 1.0$</td></tr>
</table>

21.QUERY: Kant (concrete person)				
ImSE			**Solr**	
R.D.	**Rank**	**Retrieved images:**	**Corresponding retrieved images:**	
1	1	Kant 327x400 (wb, portrait).jpg	Kant 327x400 (wb, portrait).jpg	
1	2	Kant 220x273 (wb, portrait).jpg	Kant 220x273 (wb, portrait).jpg	
1	3	Kant 285x456 (col, profile).jpg	Kant 285x456 (col, profile).jpg	
1	4	Kant 258x286 (wb, profile).jpg	Kant 258x286 (wb, profile).jpg	
5	5	steen20.jpg	–	
5	6	1steen20.jpg (*copy*)	–	
5	7	2steen20.jpg (*copy*)	–	
5	8	etikk_uke6_1x400.jpg (*copy*)	–	
5	9	etikk_uke6_2x700.jpg	–	
5	10	Bryllup 700x525.jpg (*copy*)	–	
5	11	Bryllup 300x225.jpg (*copy*)	–	
5	12	nurnbergdommene.jpg	–	
R.D.		**Relevant images that are not retrieved:**	**Corresponding relevant images that are not retrieved:**	
Evaluation metrics:			**Evaluation metrics:**	
$R_1 = 4$, $Ra_1 = 4$, A=12 $pre_1 = 0.36$ $rec_1 = 1.0$		$R_2 = 4$, $Ra_2 = 4$, A=12 $pre_2 = 0.36$ $rec_2 = 1.0$	$R_1 = 4$, $Ra_1 = 4$, A=4 $pre_1 = 1.0$ $rec_1 = 1.0$	$R_2 = 4$, $Ra_2 = 4$, A=4 $pre_2 = 1.0$ $rec_2 = 1.0$

22. QUERY: Maleri, painting (subject search)			
ImSE			**Solr**
R.D.	**Rank**	**Retrieved images:**	**Corresponding retrieved images:**
4	1	Copernicus 280x400 (col, painting).jpg	–
1	2	Copernicus 558x398 (col, painting).jpg	Copernicus 558x398 (col, painting).jpg
1	3	Platon_Aristotle 263x336 (col,painting).jpg	Platon_Aristotle 263x336 (col,painting).jpg
1	4	Hannibal 712x437.jpg	Hannibal 712x437.jpg
1	5	Hanibal 400x250.jpg (*copy*)	Hanibal 400x250.jpg
1	6	prayer ostade 300x372.jpg	prayer ostade 300x372.jpg
1	7	steen20.jpg	Not retrieved
1	8	1steen20.jpg (*copy*)	Not retrieved
1	9	2steen20.jpg (*copy*)	Not retrieved
1	10	etikk_uke6_1x400.jpg (*copy*)	etikk_uke6_1x400.jpg
1	11	Fat kitchen Heyden 380x293.jpg	Fat kitchen Heyden 380x293.jpg

1	12	Hopper Automat 700x543.jpg	Hopper Automat 700x543.jpg
1	13	gauguin 760x290.jpg	gauguin 760x290.jpg
1	14	gauguin 600x229.jpg (*copy*)	gauguin 600x229.jpg
1	15	Skolen i Athen 721x482.jpg	Skolen i Athen 721x482.jpg
1	16	Skolen i Athen 450x301.jpg (*copy*)	Skolen i Athen 450x301.jpg
1	17	Akademiet i Athen 590x600.jpg	Akademiet i Athen 590x600.jpg
1	18	platons_akademi.jpg (*copy*)	Not retrieved
1	19	Air pump_JWright 760x551.jpg	Air pump_JWright 760x551.jpg
1	20	Air pump_JWright 400x290.jpg (*copy*)	Air pump_JWright 400x290.jpg
1	21	Air pump JWright 546x396.jpg (*copy*)	Air pump JWright 546x396.jpg
1	22	etikk_uke3x400.jpg	etikk_uke3x400.jpg
1	23	etikk_uke3x200x253.jpg (*copy*)	etikk_uke3x200x253.jpg
1	24	socrates death 500x325.jpg	socrates death 500x325.jpg
1	25	socrates death 350x.jpg (*copy*)	socrates death 350x.jpg
1	26	bryllup 700x525.jpg	bryllup 700x525.jpg
1	27	bryllup 300x225.jpg (*copy*)	bryllup 300x225.jpg
1	28	etikk_uke6_2x700.jpg (*copy*)	Not retrieved
1	29	sence of taste 380x221 .jpg	sence of taste 380x221 .jpg
R.D.		**Relevant images that are not retrieved:**	**Corresponding relevant images that are not retrieved:**
1		goya fengsel1.jpg	Not retrieved
1		goya_prison1.jpg (*copy*)	Not retrieved
1		Goya-2.jpg	Not retrieved
1		Goyafengsel2.jpg	Not retrieved
1		Cicero.jpg	Not retrieved
1		cicero 450x .jpg (*copy*)	Not retrieved
1		Der Wanderer %C3%BCber den Nebelmeer .jpg	Not retrieved
1		DESCARTES.jpg	Not retrieved
1		Divana commedia.jpg	Not retrieved

Evaluation metrics:		**Evaluation metrics:**	
$R_1 = 37$, $Ra_1 = 28$, A=29 $pre_1 = 0.97$ $rec_1 = 0.76$	$R_2 = 38$, $Ra_2 = 29$, A=29 $pre_2 = 1.0$ $rec_2 = 0.76$	$R_1 = 37$, $Ra_1 = 23$, A=23 $pre_1 = 1.0$ $rec_1 = 0.62$	$R_2 = 38$, $Ra_2 = 23$, A=23 $pre_2 = 1.0$ $rec_2 = 0.61$

23. QUERY: Foto, fotobilder, photobilder (subject search)			
ImSE			Solr
R.D.	Rank	Retrieved images:	Corresponding retrieved images:
5	1	tiger_on_sofa 350x281 (col, photobilde).jpg	tiger_on_sofa 350x281 (col, photobilde).jpg
1	2	haberm 200x279 (col, photo).jpg	haberm 200x279 (col, photo).jpg
1	3	fotballkamp 2 430x307 (col, photo).jpg	fotballkamp 2 430x307 (col, photo).jpg
1	4	footballkamp 1 450x243 (col, photo).jpg	footballkamp 1 450x243 (col, photo).jpg
1	5	Kitzsteinhorn Stuetze 300x376 (wb,photo).jpg	Kitzsteinhorn Stuetze 300x376 (wb,photo).jpg
1	6	rc%5B1%5D 200x265 (wb, photo).jpg	rc%5B1%5D 200x265 (wb, photo).jpg
1	7	Popper 300x377 (wb, photo).jpg	Popper 300x377 (wb, photo).jpg
1	8	Popper 283x385 (col, photo).jpg	Popper 283x385 (col, photo).jpg
1	9	Popper 200x251 (wb, photo).jpg *(copy)*	Popper 200x251 (wb, photo).jpg
1	10	baktaman 226x340 (col, photo).jpg	baktaman 226x340 (col, photo).jpg
1	11	Carnap Rudolf 200x265 (wb, photo).jpg	Carnap Rudolf 200x265 (wb, photo).jpg
1	12	Weber Max 313x400 (wb, photo).jpg	Weber Max 313x400 (wb, photo).jpg
1	13	weber 200x319 (wb, photo).jpg *(copy)*	weber 200x319 (wb, photo).jpg
1	14	weber 345x550 (wb, photo).jpg *(copy)*	weber 345x550 (wb, photo).jpg
1	15	hempel carl 150x188 (wb, photo).jpg	hempel carl 150x188 (wb, photo).jpg
1	16	gadamer-sm 200x262 (col, photo).jpg	gadamer-sm 200x262 (col, photo).jpg
1	17	Nietzsche 212x234 (wb, photo).jpg	Nietzsche 212x234 (wb, photo).jpg
1	18	popper 150x188 (wb, photo).jpg *(copy)*	popper 150x188 (wb, photo).jpg
1	19	Skjervheim Hans 200x243 (wb,photo).jpg	Skjervheim Hans 200x243 (wb,photo).jpg
1	20	bush.jpg	Not retrieved
1	21	bush 340x249 (col, photo).jpg *(copy)*	bush 340x249 (col, photo).jpg
1	22	Durkheim2 200x257 (wb, photo).jpg	Durkheim2 200x257 (wb, photo).jpg
1	23	Durkheim2 150x193 (wb, photo).jpg *(copy)*	Durkheim2 150x193 (wb, photo).jpg
1	24	Tranøy Knut Erik 110x144 (col, photo).jpg	Tranøy Knut Erik 110x144 (col, photo).jpg
1	25	flaggbrenning.jpeg	Not retrieved
1	26	shankar.jpeg	Not retrieved

R.D.		Relevant images that are not retrieved:	Corresponding relevant images that are not retrieved:
1		1irak gassangrep.jpg	Not retrieved
1		1viet_napalm.jpg	Not retrieved
1		2abbas .jpg	Not retrieved
1		gadamer1.jpg	Not retrieved
1		Skjervheim.jpg	Not retrieved
1		Agora ancient 640x480.jpg	Not retrieved
1		Agora ancient 350x260.jpg (*copy*)	Not retrieved
1		agora 400x 268.jpg	Not retrieved
1		p05002 508x350.jpg	Not retrieved
1		p05002 400x276.jpg (*copy*)	Not retrieved
1		nurnbergdommene.jpg	Not retrieved
1		Bootboys.jpg	Not retrieved
1		charles_taylor.jpg	Not retrieved
1		che_858_1193599351.jpg	Not retrieved
1		Jon Elster.jpg	Not retrieved
1		binladen.jpg	Not retrieved
1		bin_laden x 180.jpg	Not retrieved

Evaluation metrics:		Evaluation metrics:	
$R_1 = 43$, $Ra_1 = 25$, A=26 $pre_1 = 0.96$ $rec_1 = 0.58$	$R_2 = 43$, $Ra_2 = 25$, A=26 $pre_2 = 0.96$ $rec_2 = 0.58$	$R_1 = 43$, $Ra_1 = 22$, A=23 $pre_1 = 0.96$ $rec_1 = 0.51$	$R_2 = 43$, $Ra_2 = 22$, A=23 $pre_2 = 0.96$ $rec_2 = 0.51$

24. QUERY: Brueghel (concrete person)			
ImSE			**Solr**
R.D.	**Rank**	**Retrieved images:**	**Corresponding retrieved images:**
1	1	sence of taste 380x221.jpg	Not retrieved
R.D.		**Relevant images that are not retrieved:**	**Corresponding relevant images that are not retrieved:**

Evaluation metrics:		Evaluation metrics:	
$R_1 = 1$, $Ra_1 = 1$, A=1 $pre_1 = 1.0$ $rec_1 = 1.0$	$R_2 = 1$, $Ra_2 = 1$, A=1 $pre_2 = 1.0$ $rec_2 = 1.0$	$R_1 = 1$, $Ra_1 = 0$, A=0 $pre_1 = 0.0$ $rec_1 = 0.0$	$R_2 = 1$, $Ra_2 = 0$, A=0 $pre_1 = 0.0$ $rec_1 = 0.0$

25.QUERY: Jacques Louis (concrete person)			
ImSE			**Solr**
R.D.	**Rank**	**Retrieved images:**	**Corresponding retrieved images:**
1	1	socrates death 500x325.jpg	Not retrieved
1	2	socrates death 350x.jpg (*copy*)	Not retrieved
R.D.		**Relevant images that are not retrieved:**	**Corresponding relevant images that are not retrieved:**
Evaluation metrics:		**Evaluation metrics:**	
$R_1 = 2$, $Ra_1 = 2$, A=2 $pre_1 = 1.0$ $rec_1 = 1.0$	$R_2 = 2$, $Ra_2 = 2$, A=2 $pre_2 = 1.0$ $rec_2 = 1.0$	$R_1 = 2$, $Ra_1 = 0$, A=0 $pre_1 = 0.0$ $rec_1 = 0.0$	$R_2 = 2$, $Ra_2 = 0$, A=0 $pre_1 = 0.0$ $rec_1 = 0.0$

26. QUERY: amerikanske presidenter (concrete persons)			
ImSE			**Solr**
R.D.	**Rank**	**Retrieved images:**	**Corresponding retrieved images:**
1	1	bush.jpg	Not retrieved
1	2	bush 340x249.jpg (*copy*)	Not retrieved
5	3	bin_laden.jpg	–
5	4	bin_laden x 180.jpg (*copy*)	–
4	5	1viet_napalm.jpg	Not retrieved
4	6	1irak gassangrep.jpg	Not retrieved
4	7	2abbas.jpg	Not retrieved
R.D.		**Relevant images that are not retrieved:**	**Corresponding relevant images that are not retrieved:**
Evaluation metrics:		**Evaluation metrics:**	
$R_1 = 2$, $Ra_1 = 2$, A=7 $pre_1 = 0.29$ $rec_1 = 1.0$	$R_2 = 5$, $Ra_2 = 5$, A=7 $pre_2 = 0.71$ $rec_2 = 1.0$	$R_1 = 2$, $Ra_1 = 0$, A=0 $pre_1 = 0.0$ $rec_1 = 0.0$	$R_2 = 5$, $Ra_2 = 0$, A=0 $pre_1 = 0.0$ $rec_1 = 0.0$

<table>
<tr><th colspan="5">27. QUERY: Jan Steen (concrete person)</th></tr>
<tr><th colspan="3">ImSE</th><th colspan="2">Solr</th></tr>
<tr><th>R.D.</th><th>Rank</th><th>Retrieved images:</th><th colspan="2">Corresponding retrieved images:</th></tr>
<tr><td>1</td><td>1</td><td>steen20.jpg</td><td colspan="2">Not retrieved</td></tr>
<tr><td>1</td><td>2</td><td>1steen20.jpg (copy)</td><td colspan="2">Not retrieved</td></tr>
<tr><td>1</td><td>3</td><td>2steen20.jpg (copy)</td><td colspan="2">Not retrieved</td></tr>
<tr><td>1</td><td>4</td><td>etikk_uke6_1x400.jpg (copy)</td><td colspan="2">Not retrieved</td></tr>
<tr><td>1</td><td>5</td><td>etikk_uke3x400.jpg</td><td colspan="2">Not retrieved</td></tr>
<tr><td>1</td><td>6</td><td>etikk_uke3x200x253.jpg (copy)</td><td colspan="2">Not retrieved</td></tr>
<tr><td>5</td><td>7</td><td>etikk_uke6_2x700.jpg</td><td colspan="2">–</td></tr>
<tr><td>5</td><td>8</td><td>bryllup 700x525.jpg (copy)</td><td colspan="2">–</td></tr>
<tr><td>5</td><td>9</td><td>bryllup 300x225.jpg (copy)</td><td colspan="2">–</td></tr>
<tr><td>5</td><td>10</td><td>sence of taste 380x221.jpg</td><td colspan="2">–</td></tr>
<tr><td colspan="3"></td><td colspan="2"></td></tr>
<tr><th>R.D.</th><th></th><th>Relevant images that are not retrieved:</th><th colspan="2">Corresponding relevant images that are not retrieved:</th></tr>
<tr><td colspan="3"></td><td colspan="2"></td></tr>
<tr><th colspan="3">Evaluation metrics:</th><th colspan="2">Evaluation metrics:</th></tr>
<tr><td colspan="2">$R_1 = 9$, $Ra_1 = 6$, A=10
$pre_1 = 0.60$
$rec_1 = 0.67$</td><td>$R_2 = 9$, $Ra_2 = 6$, A=10
$pre_2 = 0.60$
$rec_2 = 0.67$</td><td>$R_1 = 9$, $Ra_1 = 0$, A=0
$pre_1 = 0.0$
$rec_1 = 0.0$</td><td>$R_2 = 9$, $Ra_2 = 0$, A=0
$pre_1 = 0.0$
$rec_1 = 0.0$</td></tr>
</table>

<table>
<tr><th colspan="5">28. QUERY: fotball, football, fotballkamper (subject search)</th></tr>
<tr><th colspan="3">ImSE</th><th colspan="2">Solr</th></tr>
<tr><th>R.D.</th><th>Rank</th><th>Retrieved images:</th><th colspan="2">Corresponding retrieved images:</th></tr>
<tr><td>1</td><td>1</td><td>fotballkamp 2 430x307 (col, photo).jpg</td><td colspan="2">fotballkamp 2 430x307 (col, photo).jpg</td></tr>
<tr><td>1</td><td>2</td><td>fotballkamp 2 450x243.jpg</td><td colspan="2">fotballkamp 2 450x243.jpg</td></tr>
<tr><td>1</td><td>3</td><td>che_858_1193500351.jpg</td><td colspan="2">Not retrieved</td></tr>
<tr><td colspan="3"></td><td colspan="2"></td></tr>
<tr><th>R.D.</th><th></th><th>Relevant images that are not retrieved:</th><th colspan="2">Corresponding relevant images that are not retrieved:</th></tr>
<tr><td colspan="3"></td><td colspan="2"></td></tr>
<tr><th colspan="3">Evaluation metrics:</th><th colspan="2">Evaluation metrics:</th></tr>
<tr><td colspan="2">$R_1 = 3$, $Ra_1 = 3$, A=3
$pre_1 = 1.0$
$rec_1 = 1.0$</td><td>$R_2 = 3$, $Ra_2 = 3$, A=3
$pre_1 = 1.0$
$rec_1 = 1.0$</td><td>$R_1 = 3$, $Ra_1 = 2$, A=2
$pre_1 = 0.67$
$rec_1 = 0.67$</td><td>$R_2 = 3$, $Ra_2 = 2$, A=2
$pre_1 = 0.67$
$rec_1 = 0.67$</td></tr>
</table>

<table>
<tr><th colspan="5">29. QUERY: Peter Brueghelden (concrete person)</th></tr>
<tr><th colspan="3">ImSE</th><th colspan="2">Solr</th></tr>
<tr><th>R.D.</th><th>Rank</th><th>Retrieved images:</th><th colspan="2">Corresponding retrieved images:</th></tr>
<tr><td>1</td><td>1</td><td>etikk_uke6_2x700.jpg</td><td colspan="2">Not retrieved</td></tr>
<tr><td>1</td><td>2</td><td>bryllup 700x525.jpg (copy)</td><td colspan="2">Not retrieved</td></tr>
<tr><td>1</td><td>3</td><td>bryllup 300x225.jpg (copy)</td><td colspan="2">Not retrieved</td></tr>
<tr><td colspan="3"></td><td colspan="2"></td></tr>
<tr><th>R.D.</th><th></th><th>Relevant images that are not retrieved:</th><th colspan="2">Corresponding relevant images that are not retrieved:</th></tr>
<tr><td colspan="3"></td><td colspan="2"></td></tr>
<tr><th colspan="3">Evaluation metrics:</th><th colspan="2">Evaluation metrics:</th></tr>
<tr><td colspan="2">R_1 = 3, Ra_1 = 3, A=3
pre_1 = 1.0
rec_1 = 1.0</td><td>R_2 = 3, Ra_2 = 3, A=3
pre_1 = 1.0
rec_1 = 1.0</td><td>R_1 = 3, Ra_1 = 0, A=0
pre_1 = 0.0
rec_1 = 0.0</td><td>R_2 = 3, Ra_2 = 0, A=0
pre_1 = 0.0
rec_1 = 0.0</td></tr>
</table>