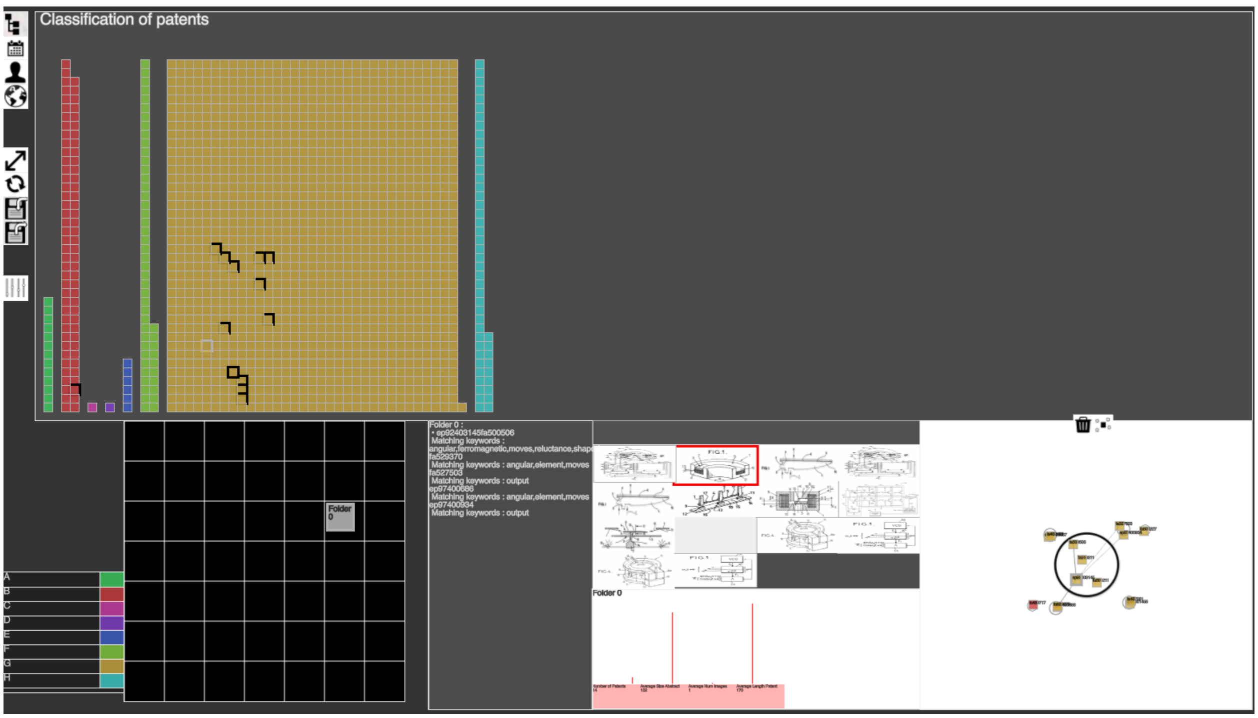
Touch based document triage with clustering of patent selections using multi-touch interactions

Technische Universiteit Delft



Kevin Allain
4424441

Touch based document triage with clustering of patent selections using multi-touch interactions

THESIS

submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

in

COMPUTER SCIENCE

by

Kevin Allain

born in Paris, France

Computer Graphics and Visualization Group Department of Intelligent Systems
Faculty EEMCS, Delft University of Technology Delft, the Netherlands
www.ewi.tudelft.nl

Research and Development
European Patent Office
The Hague, the Netherlands
www.epo.org

Abstract

The European Patent Office is the office that administrates the delivery of patents to authors making requests to them. One of their main missions is to establish the state of the art related to the applications they receive and verify the validity of the applications they receive.
The number of patents the European Patent Office has to deal with keeps increasing, and examiners need faster tools to analyse the patent applications. The most time consuming part of this process is the comparison of the patent applications with the state of the art. The current tools of the European Patent Office only allow the examiners to deal with one patent returned by their query engine at the time.
This work proposes a prototype for reviewing a collection of patents, analysing their data and metadata, and saving them into folders. The prototype also contains functionnalities to cluster the patents in a folder and display relations based on keywords matching to help the understanding of the patents contained within. The prototype is designed to work efficiently on a large multi-touch device. This prototype and the reflexions on its pros and cons give insight on the requirements and functionalities for the process of document triage.

Keywords: Document triage, patent, multi-touch, clustering, keywords-based.

Kevin Allain
Delft, The Netherlands
August 19, 2015

Preface

This master thesis is the result of my 6 months internship at the European Patent Office (EPO), for the completion of my Digital Media Technology master from both KTH and TUDelft within EIT-ICT Labs Master School (recently renamed EIT Digital). Here I describe the project I have developed during the internship for a large touch screen within the Innovation Lab within the EPO at the Hague. This work is preceded by a literature study done for a class within TUDelft.

This project is supported by the European Patent Office and TUDelft for different purposes: the European Patent Office's need to explore new tools to increase examiners' efficiency, since the number of patent applications they receive boomed lately, and as for TUDelft, they are interested in the interaction and visualisation elements of this project.

I want to thank the persons which helped and supported me during this project, my internship supervisor Barrou Diallo, my thesis supervisor Elmar Eisemann, for their advices and pertinent critics. I also wish to thank my friends Panchamy Krishnan and Jean-Baptiste Doyon, also EIT Master students, who made my days in the Netherlands brighter.

Contents

1 Introduction

The European Patent Office examiners face many issues when comparing a patent application with the state of the art. This process is called document triage. Document triage is the process where the user has to verify the relevance of the documents returned by a search query. This step is critical in the case of patent examiners, as the selected patents will require an important time to be read.
This section describes these issues and the situation that created the need for the work presented within this thesis. The European Patent Office recently drastically increased and the complexity of the patent structure for a quick analysis. A complete description of the interesting elements of a patent allows to turn our reflexion towards a solution-solver prototype within section 1.1. This project is a software for effective document triage. It was designed for a large touch screen and thus makes use of some of its functionalities.

1.1 The context

The European Patent Office's main activity is to deliver patents. The process is simple, an inventor sends a patent application describing the invention to the EPO, and the examiners verify if the invention is worth patenting by comparing it to the state of the art. Even though the EPO's examiners are experts in their field of study, verifying the state of the art is a very time consuming task. The EPO needs to deal with an increasing number of applications. Patent filings grew for the fifth year in a row to reach 274 174 in 2014. The responsibility of the European Patent Office involves that they are at the center of the process involving the commercialisation of products where the markets are worth billions of dollars. The European Patent Office is self-financing from the procedural fees and from the annual fees for pending patent applications. The budget of the European Patent Office is 2 billion euros, thus a company so large needs to efficiently manage its expenses, including the time necessary for their highly trained employees for examining patent applications.
This enormous amount of patent applications, combined with the responsibility and financial consequences, force the European Patent Office to search for new solutions to accelerate the productivity of its examiners. Among these solutions, providing better work tools stands out.

1.2 Patent structure and EPO tools

The difficulty to create an efficient software for patent analysis comes first from the structure of a patent itself. It contains numerical and categorical data, images, and often links to other patents.

Michel de Ridder developed TouchPat, a prototype for exploring large patent collections. In his master thesis, he describes some entities that are particulary

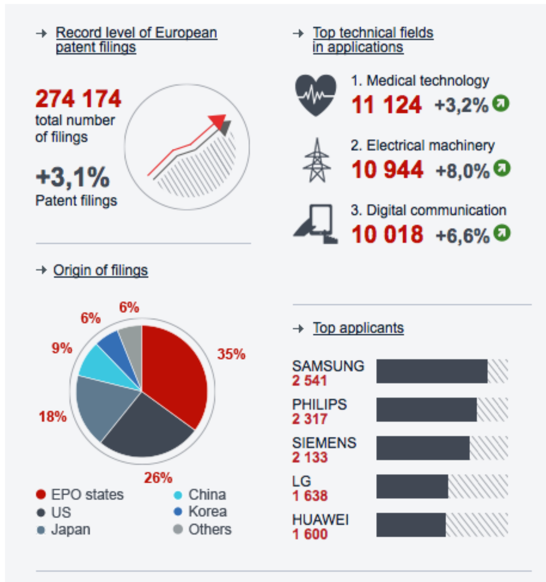


Figure 1: Statistics from the 2014 annual report of the European Patent Office

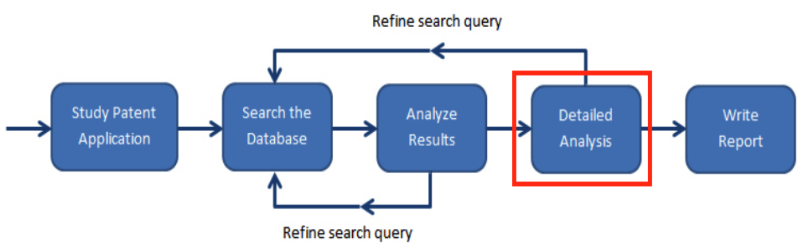


Figure 2: The process of examining a patent application. The focus of our project is on the Detailed analysis part.

important in a patent:

- Title
- Applicant: the owner of the patent
- Inventor
- Classification: The place(s) of the patent within the classification system used. This helps the user understand the meaning of a patent, and its field. A very used classification system is the International Patent Classification: IPC. The IPC is a tree structure, where each step to a child

Europäisches Patentamt
European Patent Office
Office européen des brevets

(19)

(11) Publication number: **0 392 853 A2**

(12) **EUROPEAN PATENT APPLICATION**

(21) Application number: **90303991.5**

(22) Date of filing: **12.04.90**

(51) Int. Cl.5: **G11B 7/00, G11B 7/007, G11B 7/013**

(30) Priority: **13.04.89 JP 95648/89**

(43) Date of publication of application: **17.10.90 Bulletin 90/42**

(84) Designated Contracting States: **DE FR GB NL**

(71) Applicant: **SHARP KABUSHIKI KAISHA**
22-22 Nagaike-cho Abeno-ku
Osaka 545(JP)

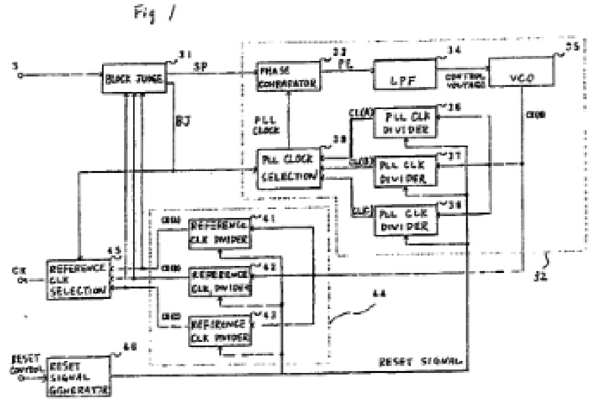
(72) Inventor: **Deguchi, Toshihisa**
C-722 Daiya Heights, 2-200-5,
Gakuendaiwa-cho
Nara-shi, Nara-ken(JP)
Inventor: **Fuji, Hiroshi**
3-1-30-206 Kataokadai, Kanmaki-cho
Kitakatsuragi-gun, Nara-ken(JP)
Inventor: **Terashima, Shigeo**
95-86 Kaminosho-cho, Nikaido
Tenri-shi, Nara-ken(JP)

(74) Representative: **Huntingford, David Ian et al**
W.P. THOMPSON & CO. Coopers Building
Church Street
Liverpool L1 3AB(GB)

(54) **An apparatus for recording and reproducing information on and from an optical disk.**

(57) An apparatus for recording and reproducing information on and from an optical disk in which at least one optical beam and a reference clock signal are used. The optical disk comprises recording tracks which are divided into a plurality of blocks and concentrically arranged along the radial direction. The apparatus comprises: a block judging unit for judging which one of the blocks is the one which is currently impinged by the optical beam; a clock signal generator for generating a plurality of clock signals which are different in frequency from each other; and a clock signal selecting unit for selecting one of the clock signals as the reference clock signal, on the basis of the judgment of the block judging unit.

Fig 1



EP 0 392 853 A2

Xerox Copy Centre

Figure 3: The first page of a patent.

describes more precisely the category of the invention. For example, Section H is ELECTRICITY, and H01 is BASIC ELECTRIC ELEMENTS.

- Publication data: This is critical data because it is for most patents when the rights owning start.
- Application number (intern to the EPO organisation)
- Abstract: short description of the invention and the claims of the invention.

In order to focus on a quick understanding of the meaning of a patent, we decided not to keep the same selection for some elements of the patent. We had the chance to have a sample of 1519 patents with the notes an examiner took during his work. He ended each analysis by writing a shorter and more understandable version of the abstract. We seized this opportunity and removed the abstract from the visualisation.
This choice had pros and cons, as not all the data from the patents was available. Losing the claims for example, did not imply a major lost for the understanding of a patent, but it is very valuable for analysing the legal situation of a patent. More will be discussed within the user feedback.
The title was not included in the visualisation since the user written description was considered explicit enough and the patent identifier was shorter. This follows our desire to provide a visualisation with only quickly readable text.
We also removed the display of the inventor and kept only the applicant, as this allows a reduction of elements to read for the user. There is no loss of a major information, due to the fact that companies can have many inventors that work on the same fields.
The examiner at the European Patent Office have several tools to search for patents.
(Ridder 11) describes the most important: Query Builder, Query Build Assistant (QBA), Viewer and Trimaran. These tools are used sequentially, and the process of examining a patent application goes through them in the following order:

1. The Query Builder tool allows the user to enter a query to search patents using keywords inside the EPOQUE, one of the European Patent Office database, and it returns a list of patents that were returned by the boolean algorithm. When facing a large collection of patents, the user refines the query to reduce the amount of patents to look at. This forces the user to face a list of patents and then has to go through them linearly for the process of document triage.
2. The QBA offers synonyms for the keywords to be entered by the user to enhance the quality of the search query. The returned list of patents is unorganized, the examiner just knows the value returned by the search

engine for his query returned true for each patent returned. No information about relevance matching is provided. If the number of returned patents is over 500, go back to step 1. This number is an approximation and depends of the examiners habits and is defined by the time constraints examiners have, and can not easily be reduced, considering the returned list is unorganized.

3. The user analyses each of the resulting patents one by one through the Viewer (cf figure 4) which allows to examine a patents content. The user saves interesting patents within folders according to their similarities in between each other, and takes notes over each patent and folder. Saving patents into folders is a part of the examiners' workflow ever since the creation of the European Patent Office, older than the use of machines.

4. With Trimaran, the user writes the final report using notes taken previously.

The current tools when analysing a group of patents resulting a search query lack spatial organisation. The usability of the software is also an issue, as the number of clicks needed to operate these is high, thus degrading the learning curve. It is interesting to note examiners go through a long process at the beginning of their employment to learn the subtitles regarding legal obligations, errors to avoid, and more interestingly regarding the use of the EPO tools they dispose.

1.3 Research Goal

We just noted the issues with the official current tools provided to the EPO examiners.
As said previously, the focus of our work is to help the user during the detailed analysis step during the process of examining a patent application. This step, recognizing the value of a document after a search query, is called document triage. It shall be described more in detail in the Design section of this paper.
Our main objective is to provide a prototype to help the process of examining a patent application.
Research goal: Investigate and provide new solutions to allow the user to deal with a large amount of patents for the process of document triage. The solutions have to allow the user to navigate through patents by using a large multi-touch input device, order the patents into a collection, modify the collections of patents, and visualise data and metadata from this collection using a keyword-based clustering.

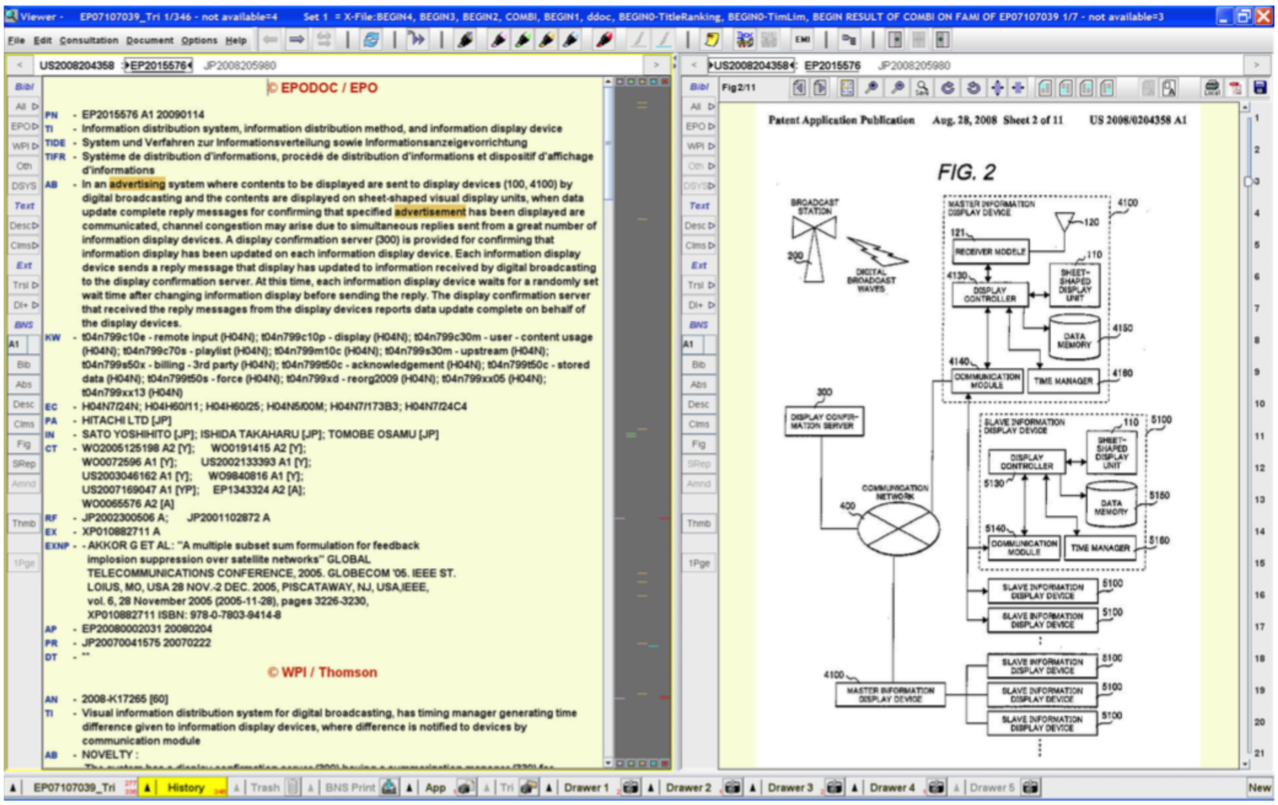


Figure 4: The User Interface of the Viewer. It is composed of three parts: Description and claims on the left, drawings on the right, and the middle bar shows where annotations were made by the examiner.

2 Literature review

The problems listed in the previous section indicate various needs for the user. In this section we describe previous work trying to solve some of these problems. Our project contains several research areas: document collection visualization, clustering of a collection of documents, patent semantic visualisation, tactile interaction and keywords generation.
This section describes the work of Anne Marijn van Ee (Marijn 12), based on the work of Michel de Ridder (Ridder 11), which both did their master thesis on the subject of visualisation and interaction of patent collections within the European Patent Office. Their work was a strong inspiration for the start of our prototype. We then describe the papers we found the most interesting concerning the visualisation methods for a large amount of documents. A section is designed to present previous work on document triage specifically designed for patent semantics. Finally, we review interesting papers regarding tactile interactions when dealing with abstract objects, which can be the representation of any kind of document.

2.1 Organization Viewer and TouchPat

Anne Marijn van Ee (Marijn 12) and Michel de Ridder (Ridder 11) did their master thesis in TUDelft and the European Patent Office. The work of Marijn is heavily based on the work of (Ridder 11), using the same code as a starting point. This allowed her to focus on the implementation of the LAMP algorithm (Joia 11), to cluster related patents together.
Michel de Ridder (Ridder 11) created TouchPat, a prototype to visualise and browse through a collection of patents using semantics multi-touch zooming. This method of display is called a Zoomable User Interface (ZUI). This method uses the spatial memory of the user when zooming and unzooming on the collection of patents. This is especially relevant since patents do not move when the user zooms in and out, the position of each patent within the grid is kept. Positions of the patents can be reordered by criteria chosed by the user. This is very efficient for navigating through a collection of up to 1000 patents. Nevertheless, the spatial organisation is linear and only provide linear metadata for quicker analysis for the user, and thus improvements can be made.
TouchPat was created using the Microsoft Surface Beta API. This API allows to code using C# and the Windows Presentation Foundation. It runs on Windows 7, and we did not find on their website if it runs on Windows 8 and 10, but we suspect it does. The official website of the Windows Surface 2.0 SDK : http://www.microsoft.com/en-us/download/details.aspx?id=26716.

The user can, using a button in the bottom of the screen, choose to display small coloured rectangles within the widgets, to indicate in which of the patents the annotation is found. As an example, if the user selects the annotation "optical" which is associated to a color randomly generated (let's say green), a small green rectangle will be visible within the widgets of the patents containing this annotation.

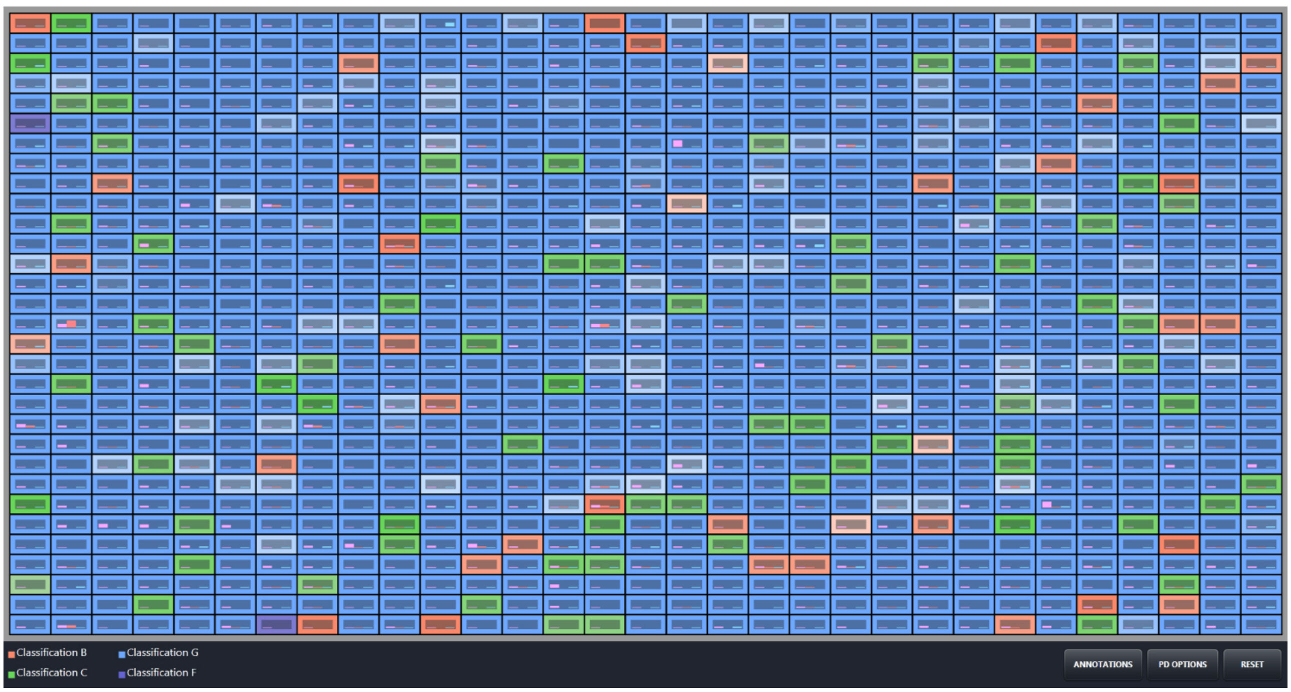


Figure 5: The TouchPat prototype. Each one of these 1000 widget represents a patent. The user can zoom in to increase the amount of data displayed within the widgets. The bottom left legend indicates the classification of the patent. The grid organisation only provides linear metadata for the user. (Ridder 11)

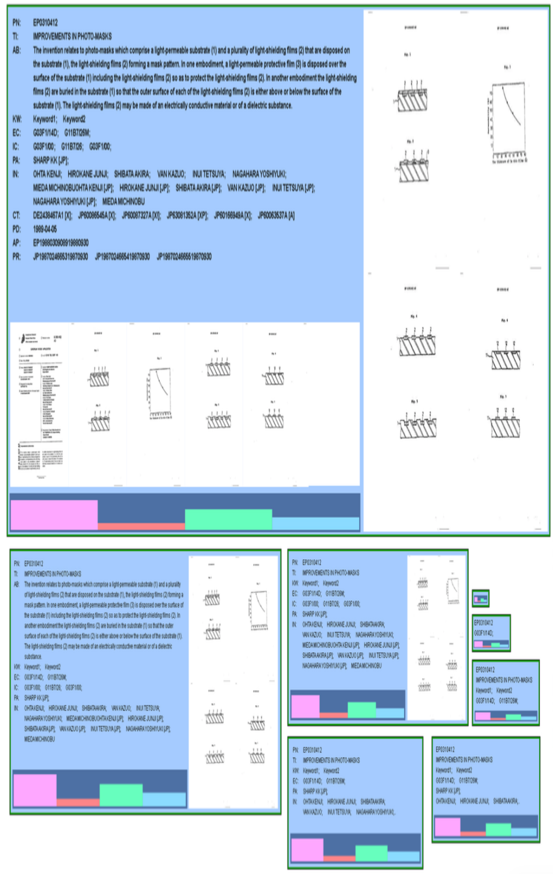


Figure 6: The data displayed by a widget according to different levels of zoom. (Ridder 11)

TouchPat also contains a Detail Viewer, which allows the user to display the informations of a patent, in a highly similar way than the official Viewer of the EPO. This viewer is cut into three blocks: full text, annotations bar, and images.
The user can also save selected patents into drawers, in order to analyse them next to each other.
The interaction tool developped by (Ridder 11) is based on a long touch, where a menu around the finger appears, and the user can go through the menu by dragging his finger over. If a submenu exists, it appears after the user first drags his finger above menu containing it.

Anne Marijn van Ee (Marijn 12), after analysing the work done by (Ridder 11), decided to start working using his code as a base, and enhance the points she judged could provide insight to the user during the document triage process.

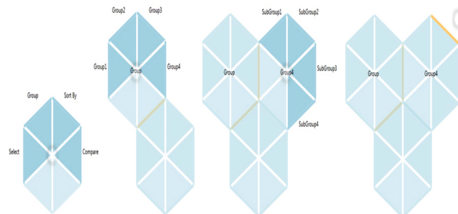
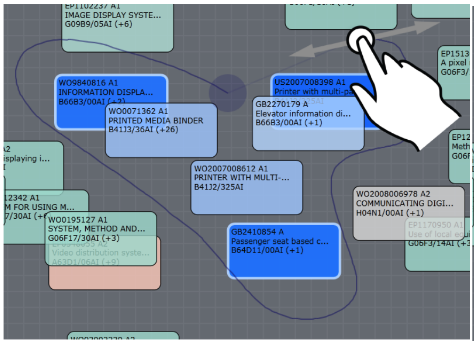


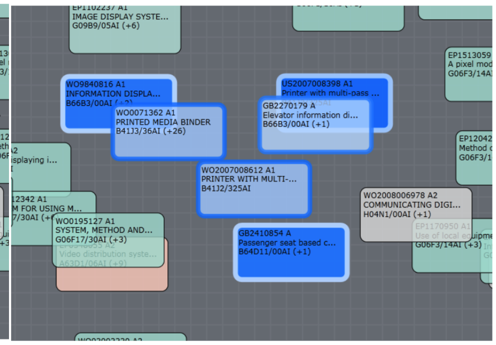
Figure 7: The context menu. The use drags his finger over a block to select it. If a submenu exists, it appears after the user dragged his finger over the block containing it. (Ridder 11)

Her research goal was the following: Investigate and develop a prototype that provides the user an overview to explore and organize a patent collection by clustering the documents based on a similarity measure. The user is able to organize and navigate the collection using a multi-touch input device.
Her research goal is driven towards being an amelioration of the work of de Ridder, with the addition of the possibility to organisze patent collections by clustering documents based on a similarity measure. Our research goal is also heavily influenced by hers, even though our focus to answer it is different. She also focused on providing freedom for manually organising the patents as desired by the user.
But before talking about her methods to organise the patents, we can first quickly observe how she designed stacks of patents. (Marijn 12) also focuses on giving the user the possibility to manually organise patents according to their taste. To do so, she implemented the lasso gesture. The lasso gesture is a simple concept: the user keeps his finger dragged around what he wishes to select, and lets go of his finger once what he draws with his finger contains what he wishes to select. Most of the time, the user does not release his finger at the same position as where it first put it down. Thus, the spatial selection has to be closed, simply by adding a line in between the beginning point and the ending point.

The cluster method of (Marijn 12) is based on a weighted Local Affine Multidimensional Projection using a similarity measure based on the classifications. The LAMP algorithm is based on the selection of patents as what (Marijn 12) calls "examples". These examples will be use as controls points to position the data. The position of the examples is thus used as an input for the projection of the position of all the other patents.
Marking patents as examples for LAMP is a way to indicate which patents have been reviewed by the patent examiner. While the results of the mix between LAMP and similarity measure provide interesting results, we would rather avoid forcing the user solutions that imply that the user already knows some relevant patents, because it is ultimately our goal and it is an important assumption. Thus, the algorithm is not within the scope of this thesis. Results only using the LAMP algorithm are not very efficient, thus (Marijn 12) creates weighted



The lasso gesture allows the user to select a group of patents by dragging his finger around it. (Marijn 12)



The patents that were within the lasso gesture are then highlighted for the user to evaluate if his movements were correct. (Marijn 12)

version, to bring better results.
The similarity can be calculated either according to text similarity or classification within the Organisation Viewer. We invite the user to read (Marijn 12) for implementation details. Even though the result look pretty promising, (Marijn 12) ends up her thesis on very interesting notes about the difficulty for users to understand the reason of the position of the patents. We also believe that bringing clear distinction in between the clusters around the example patents could ease the understanding of the meaning of patents, like (Dunne 12).

2.2 Visualisation methods for large collection of documents

The analysis of a large collection of documents is still an open problem today. Due to the high number of possible types of document, the literature review concerning can include fields that may not seem directly related to our context, but where their solutions can be a good influence for our work.
Many scientific papers focused on the relations between authors working together and citing each other rather than on the content or meaning of the scientific paper. Nevertheless, these papers provide rich insight on how to visualise networks together, hierarchical relations such as tree structure, or still sometimes focus on one document's metadata. We also read papers about data representation within a 3 dimensions space, and we will explain in section 3 why we decided to let go of a 3 dimensions environment to focus on a more efficient prototype.
(Bergstrom 09), (Dunne 12) and (Kairam 12). (Bergstrom 09) focus on providing an efficient visualisation from a paper to its citations, and from author to author, whether the relation is a citation or collaborations. This type of structure is a network. The user deals with an important amount of elements

that might or might not share common data and metadata. The objective of (Bergstrom 09) is to provide insight on the position of a paper within the context of a field research, which is a step also included within the process of document triage. While we focus on keywords-based relations, this paper still gave us a rich insight.
The circle view shows a group of circles where the centered one represents a selected scientific paper. On each circle is written the title of the scientific paper and its author. Reading the titles of papers surrounding the selected one allows the user to get a better insight on the selected paper, without reading it. While it is time consuming if this method is employed for a selection, we believe using the neighbors to get a better insight of a document without reading it is an idea with high potential. Still, some design choices, such as numbers on links are not clear and readable enough and might disturb the user's focus.

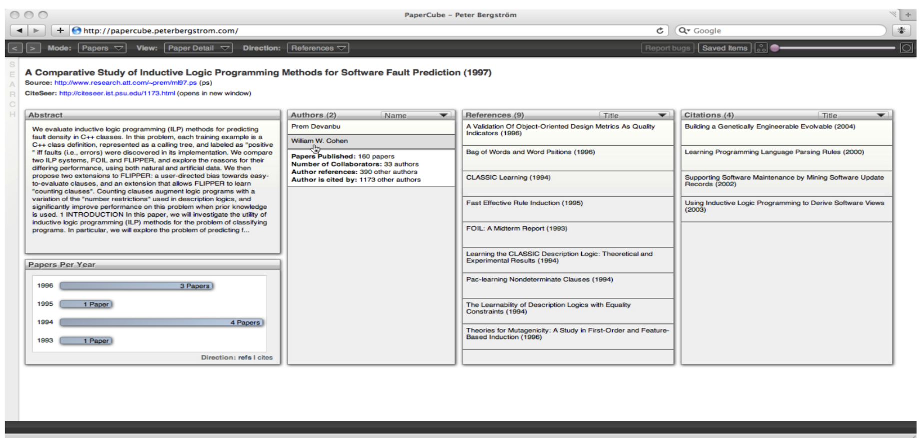


Figure 8: The paper view of (Bergstrom 09). It is interesting for us to see the similarities in between a scientific paper and a patent, such as a title, an author, an abstract, a publication date, references. Nevertheless, (Bergstrom 09) did not include a space for images within his visualisation. The importance of images within a scientific paper is not a standard like for patents.

(Dunne 12) created a software for helping analysts to generate surveys on fields they are unfamiliar with, no matter the audiences and levels. Action Science Explorer (ASE) is also based on a node-link visualisation, but the data is not only clustered in a group, but interestingly the clusters are clearly visible for the user thanks to a circular shape surrounding each cluster. The calculation of the clusters is not within the scope of our analysis, but we refer the interested reader to (Perer & Shneiderman, 2006). Each paper is identified by a small rectangle containing an ID for the user to recognize it. This visualisation is very rich and allows the user to establish the quality of the results given by a search query. Indeed, if the visualisation is composed of a high number of cluster containing relatively small amount of papers, it thus means the selection is too

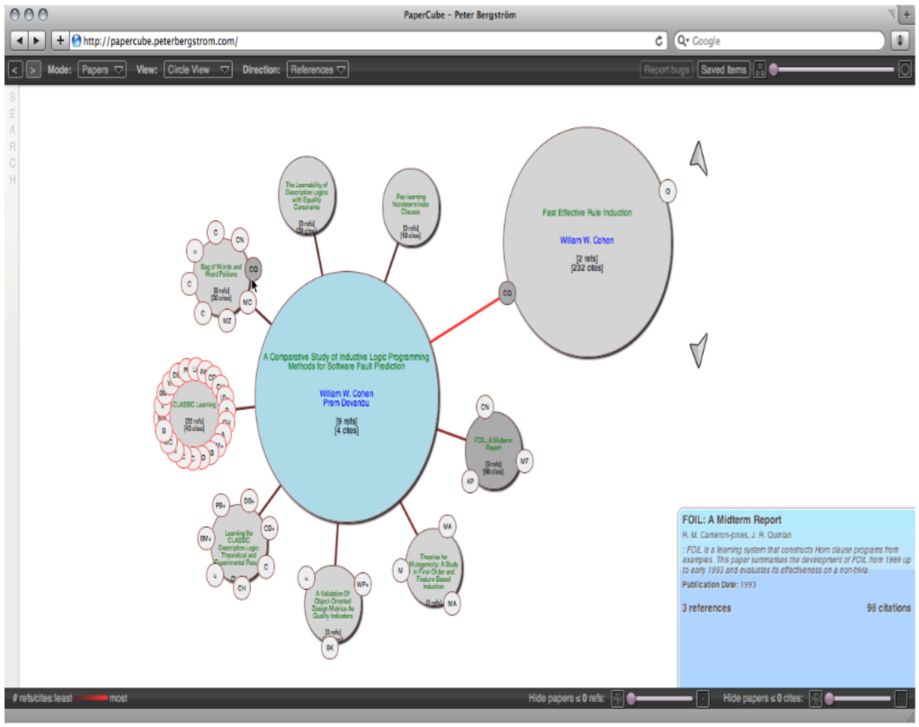


Figure 9: The Circle view shows a visualisation where each circle represents a scientific paper. The size represents the number of time another paper got cited, and the color of the relationship indicates the strength of the connection in between papers (here, the same author). (Bergstrom 09)

broad, and thus the user did not manage to focus on one field and its neighbors. The visualisation contains the list of papers' labels with colors indicating the ranking of each paper according according to the citations. The visualisation also contains a section called the citation context, which is text displaying the citations within their sentences, which allows the user to understand the nature of the connection. While some papers are heavily based on one another, some have small citations only to invite the user to read it if they feel interested while noticing the content of the citation is different enough from the paper for the user not to need explanations on it. This is valuable inspiration as it shows that connections do not mean shared meaning. This pushed us towards keywords-based visualisation, as our objective is maybe not as much as to understand entirely the invention of a patent, but to understand if the field of a patent is relevant to the document triage process. It is unfortunate that, with such a good basis for an idea, the visualisation only highlights the text concerning the citation. It can force the user to spend a lot of time reading if the user enjoys the potential of this feature.

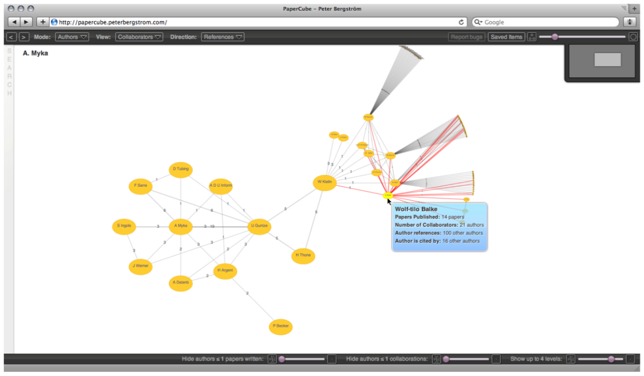


Figure 10: The Collaboratr view displays the collaborators for author in the middle of the circle up to 4 levels. The user can thus display the collaborators of collaborators. (Bergstrom 09)

Figure 11: The Treemap view focuses on the citations of papers starting from the one the user selected. The selected paper is the one on the top of the screen. The titles written on the closest citing papers help the user to understand the field of research of the paper. (Bergstrom 09)

All the citations are displayed at once within the visualisation, which we consider an overload for the user's focus, or we believe the user should at least have the possibility to reduce the amount of links to the connections starting from a single paper. While the clusters provide a good overall idea of the quality of the return, the links do not help the user to properly identify the field he is working on.

(Kairam 12) developed GraphPrism, a technique to summarize large graphs corresponding to co-authorships. While we believe in our use case the co-authorship is not very relevant, the tools set up by (Kairam 12) are interesting. The software developped by (Kairam 12) consists of a view of GraphPrism, a stacked multi-scale histogram, a classical node-link diagram, and a selection menu. GraphPrism is inspired by a modified version of the adjacency matrix, the B-matrix. A B-matrix is a histogram where each cell B(l,k) indicates by its color the number of nodes within the network that can reach k other notes in exactly l hops.

Several measures are available for the graphs with GraphPrism, to measure the distance in between elements of the network. While we will not use these measures methods within our prototype, it showed us the necessity to indicate to the user the strength of connections within a collection of documents.

Even though the use of GraphPrism can seem a little bit difficult at first, we believe there is potential in dynamic visualisation of query over a node-link diagram in order to identify relations in between its elements. Finally, the

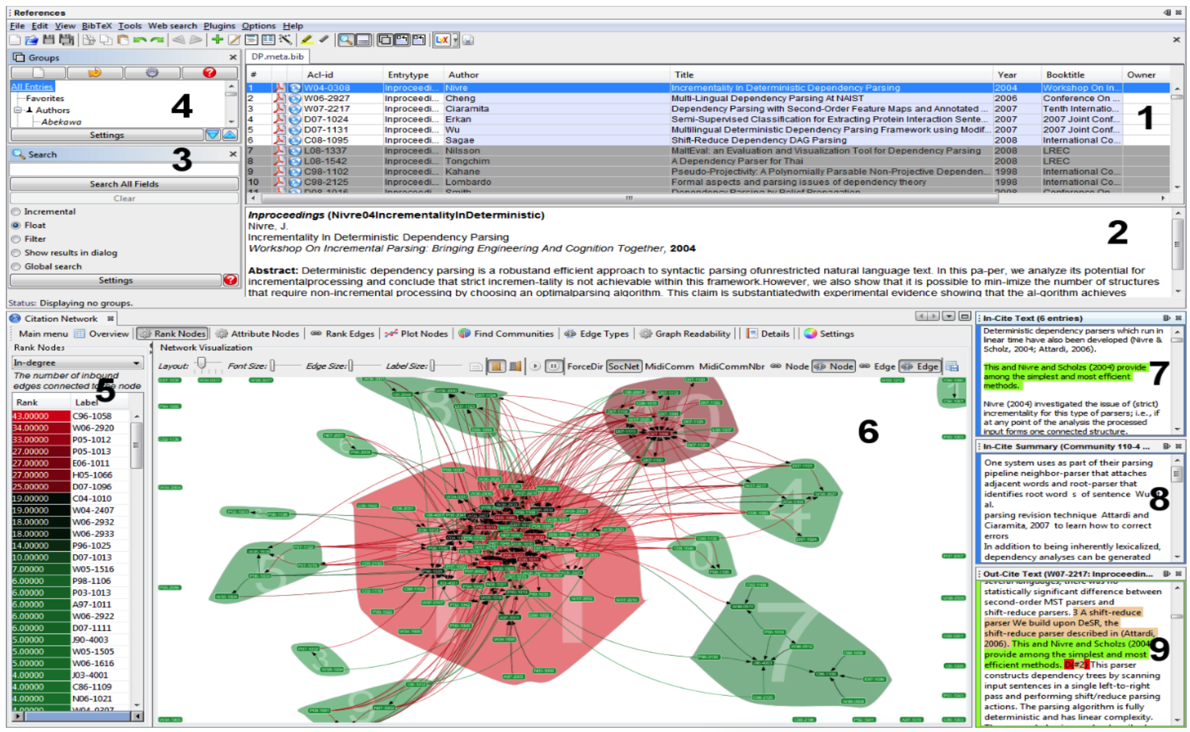


Figure 12: The main views of ASE: Reference management (1-4), Citation Network Statistics Visualization (56), Citation Context (7), Multi-Document Summaries (8), and Full Text with hyperlinked citations. The Citation Network Statistics Visualization view is calculated according to citations in between papers.

following papers, while not critical for our prototype, are interesting.

We also invite the reader to look at (Paulovich 08) for the generation of a hierarchical cluster tree that then can be visualised in a 2D space, and (Van der Maaten 08) for the technique called t-SNE, which gives each point of a high-dimensional dataset a location within a 2D or 3D space.

Analysing these papers pushed us towards the design described in Section 3. While our focus did not always match the ones presented in the papers (focusing on relations such as citations and co-authorship), many of our ideas are inspired from the presented work in this subsection.

2.3 Patents semantics

The previous section described scientific papers that inspired us towards our objectives and our design concerning organisation and navigation of networks.

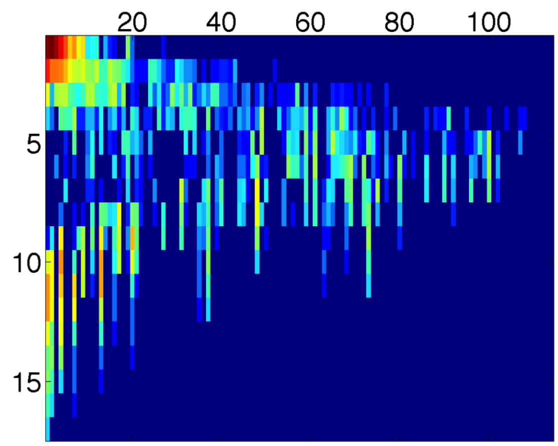


Figure 13: An example of a B-Matrix: Within a network, for each cell B(l,k), the color indicates the number of nodes that can reach exactly k other notes in exactly l hops. Color representation is a classic heat-map in this case. (Kairam 12)

This is specially useful within our widget clusters, part of widget visualisations, which design is detailled in section 3.5. Nevertheless, patents are a specific kind of documents. A patent being a document containing caterogical and numerical data, it is necessary to think of a visualisation that can adapt to the whole package of meaningful information it can provide. Thus, we searched for scientific papers describing projects focused on the work with patents. The two papers that got our attention the most are (Giereth 08) and (Koch 10). We will describe their content and the lessons to be learned from it.
(Giereth 08) developed an extension to integrate visualisations into a semantic wiki. This work combines content of the result of a search query within the project PATExpert (described in this section) with user made semantic annotations of information, rating, linking with prior art, review, translations and discussions, based on the organisation of a wiki, where every user can modify the data for enhanced relevance.
PATExpert is a project that started in 2006, its goal was to change the paradigm of processing textual information to semantic. (Giereth 08) is an interesting extension to it as it focuses on user content, which implies adding content that went through a process of reflexion, even if we do not know it, it brings a benefit a computer can't easily bring. This paper is focused on three aspects:

- Use a semantic model (a visualisation) to represent data information and user annotations.
- Use wiki technology to annotate original patent data to add semantic information
- Integrate the visualisations regarding classificatory, temporal, geographic, and other patent information aspects to work together.

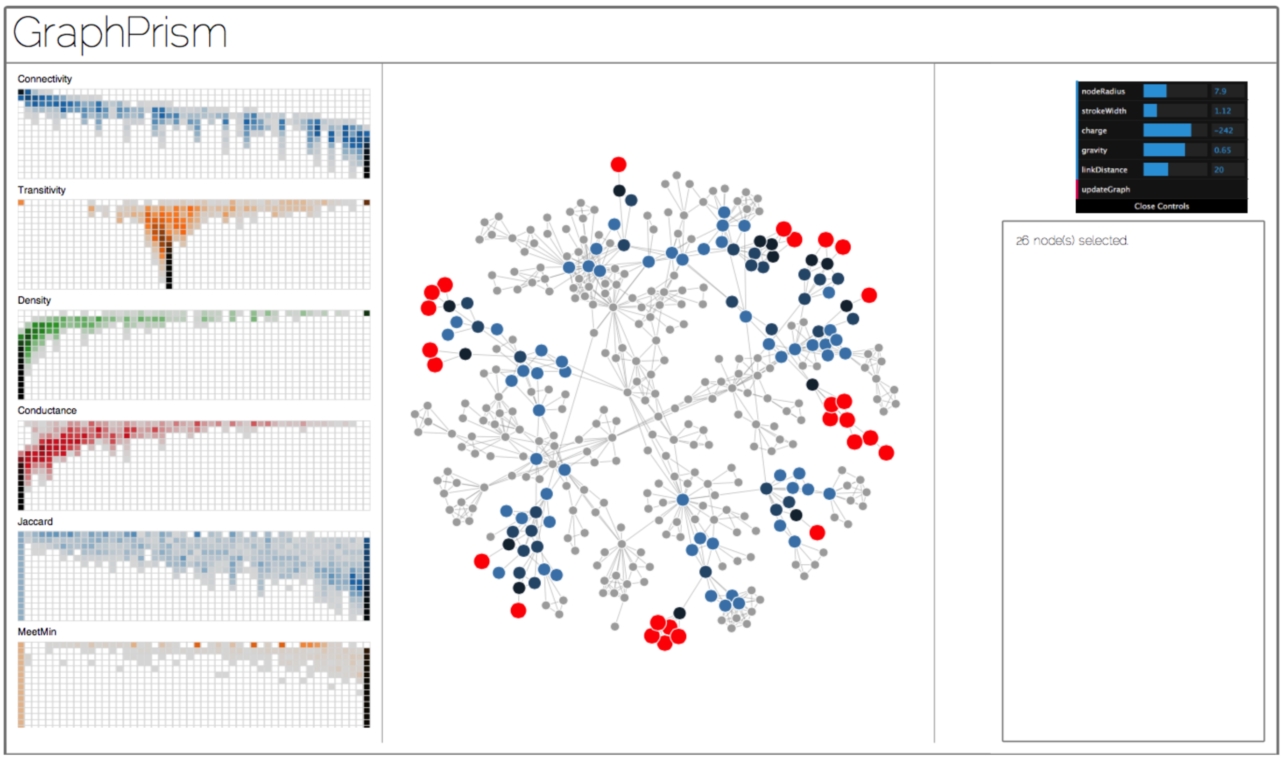


Figure 14: GraphPrism: The histograms on the left define several measures: connectivity, transivity, density, conductance, Jaccard, MeetMin (defined in section 2.2). In this figure, the user selected the nodes that reach the fewest other nodes within 3 nodes. The red elements on the network thus represent "outsiders". (Kairam 12)

Bach developed Cubix in (Bach 14). Cubix is based on a series of adjacency matrices aligned to each other to indicate the evolutions of the relations through time. Large blocks on the 3 dimensions visualisation allow the user to see an important cluster of connections through time. (Fig. 16) The interaction cube we wanted to use within our first 3D prototype was inspired from the Cubix user interface. We believe this paper has high potential for inspiring future work, as patent examiners are likely to appreciate this prototype to quickly see an interesting cluster. This will be developed in Section 8.

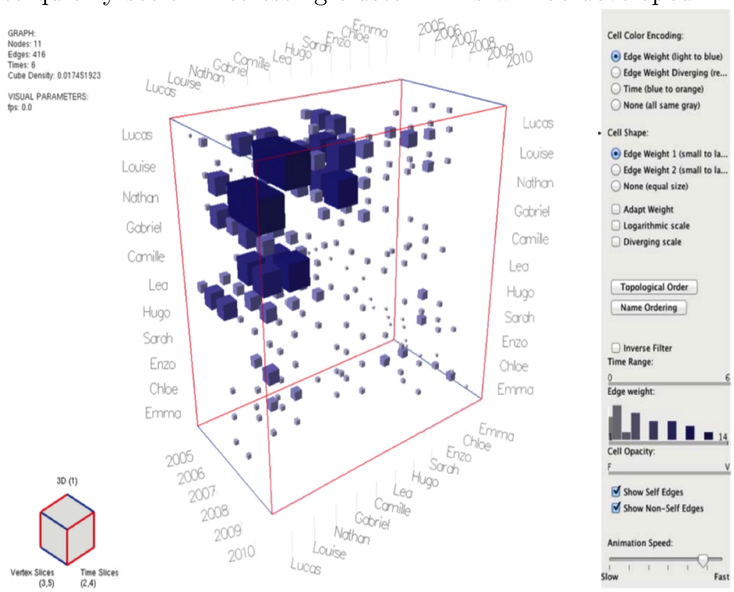


Figure 15: Cubix, developed by (Bach 14). The user can see the time evolution several adjacency matrices lined up according to time evolution. The user can navigate through the matrix using the small cube on the bottom left.

As stated by (Giereth 08), patent databases, whether they are private or public, lack the tight integration of users within a community process for commenting and reviewing patents. The author also underlines the difficulty to face different languages for patents, and the complications it brings, such as a well accepted ontology for patents terms. Ontology is a formal naming and definition of an entity that exists for a particular domain. It is necessary to make use of ontology to efficiently add the semantic data provided by the user. It is interesting for us to highlight how complex understanding the meaning of a text with complicated verbose. The author also notes the heterogeneity of the users, and thus the needs they might encounter when looking at a patent. The needs of the users were saved during a study that was produced at the beginning of the PATExpert project. What resulted from this study are the requirements for tools for the users and a desire for team-work. We believe that our prototype has potential for team-work on the large screen in future development, as the tools we use for analysis could be duplicated according to the number of users (the widget visualisation). Our reflexion on the idea will be described within Section 8.

(Giereth 08) focuses his work on combining patent search, patent visualization

Question	Yes	No
1. Do you have the necessity to change the layout of the result presentation?	75,0	25,0
2. Do you want to include comments into the results?	68,8	31,2
3. Do you have to exploit the results in cooperation with other persons?	46,2	53,8
4. Do you make the inquiries in cooperation with other persons?	30,8	69,2
5. Do you have the necessity for storing the original results of an inquiry?	82,4	17,6
6. Do you want to store meta data from the relevant patents?	81,3	18,7
7. Would it be helpful to have some kind of tool-tip within the images which can show an explanation of details?	100,0	0,0
8. Some tools are offering the possibility to highlight the hits. Is this helpful for your daily work?	69,3	30,7

Figure 16: The result of a study performed at the beginning of the PATExpert project. What results of this study are some needs for the analysis of a patent, and a desire for team-work. (Giereth 08)

and patent annotation in a semantic wiki system. This implies that each of the three elements the user faces when going through a patent analysis (patent search, patent collection, patent details) need to offer the possibility to annotate semantic data. even though our prototype tries to include as little text as possible, we believe this attitude towards adding human generated semantic data can greatly enrich the process of analysing a patent or a selection of patents by accelerating the quality assessment of a patent or selection of patents. Our advise on future work towards this direction will be discussed in section 8.

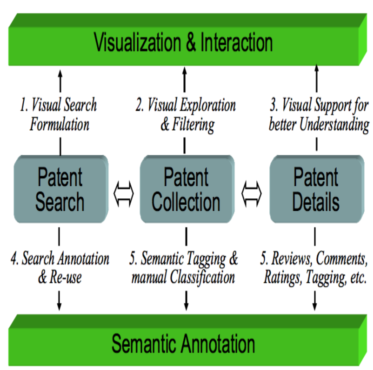


Figure 17: The support of a patent search process in order to support semantic additions from the user. (Giereth 08)

The visual framework of this extension is PatViz, and supports the analysis of temporal, geographic, classificatory, and term distributions of large patent sets. The prototype contains different visualisations set next to each other for the user to analyse different metadata at once. The top right graphic shows the distribution of the patent set according to the International Patent Classification (IPC) using an ordered treemap. The color indicates the number of classified patents within a specific IPC category. The more saturated the color, the higher the number. Zoom in is available by clicking to have a more precise view. The graphic at the center on the right is a searchable and filtrable table of the content of the patents. It is interesting to notice this is still a list, which forces the user back to a very linear analysis. There is in the right bottom a colored word map, where the colors indicate the amount of patents in a certain country. It goes from grey to red to indicate the number of patents from within a country. The visualisation contains a section with a combination of 2D visualisations applied over a 3D structure, similarly to textures on faces of a cube. These 2D visualisations are a colored world map (similar to the one previously described), a tree map, and a node-link diagram representing citations in between patents. Hovering over an element will hide the other ones within the visualisation. Linking the data representation of several visualisations at once results in a better understanding of the metadata of the patents. Still, we believe that the spatial layout of these visualisation does not feel very comfortable. Within this 3D environment, one of the 2D faces in the bottom is not easy to read, and the user can't zoom in the treemap, resulting in a difficulty to read precisely the classification. Finally in the bottom left of the visual interface, an alphabetically ordered tag cloud shows the terms appearing the most within the selection. A term listed in this space thus appears often, and in order to emphasize its importance, the size of the word depends of the number of times it is seen within the selection. The number of appearances is listed next to it for the user to have precise information. We believe the tag cloud brings in an important value to the visualisation as the user can more easily determin the subject of a group of patents by overlapping the meaning of these terms.

Databases are accessible through Esp@cenet, a service of the European Patent Office which contains more than 60 million patent documents. With such a large amount of patents, the common process examiners go through is to do a first search query, look at the results, try to evaluate as fast as possible its quality, and if necessary, which can often happen, modify the search query for a better result. This can become problematic for users as the size of the query can become too important to efficiently read it or remember all of it.

(Koch 10) describes evolutions of PatViz, which contains several interesting visualisations, and provides interesting focus on how to analyse a patent, which metadata is relevant during the patent search process. It also focuses on a visual query module to add an abstraction layer for when the user needs to retrieve patents from a database. While implementing such a functionality was out of the scope of our project, we will describe its functioning as we believe it is a good first approach towards a solution where there is a lesser need to be highly

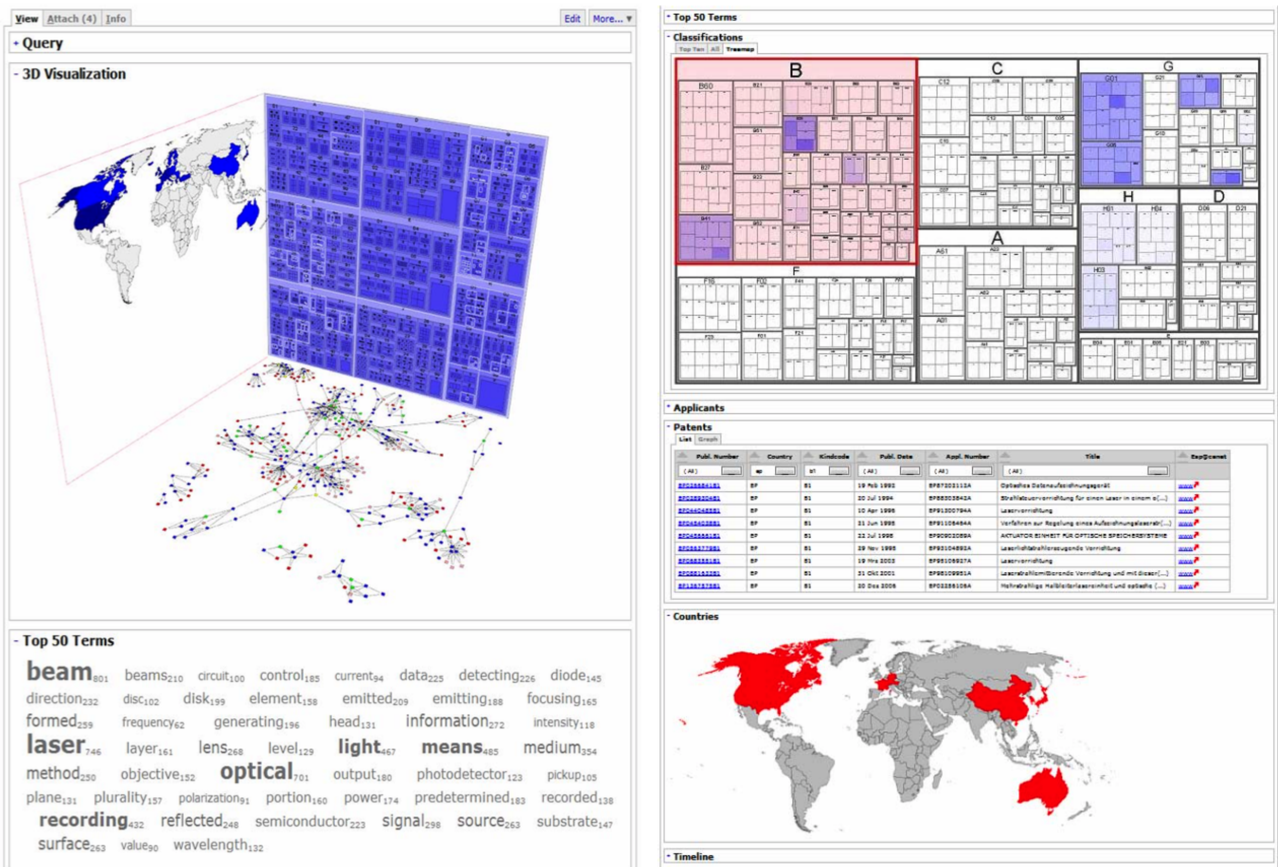


Figure 18: The visualisation of a PatViz search result at the publication time of (Giereth 08). It is composed (from top left to bottom right) : A 3D grouped visualisation composed of a colored map, a treemap, and a node-link diagram that can be dynamically brushed by the user; a could tag indicating words appearing often; a treemap indicating repartition of classifications; a list of patents with indications on semantic data; and finally a color map indicating the countries of the returned patents.

trained, and more interestingly towards our large multi-touch device solution where we want to avoid the user to be forced to type, or at least as little as possible.
(Koch 10) brings a tool for graphically representing combined search expression using a graph. While this is still a query and thus can end up complex, the spatial organisation allows for a clearer and faster search query.

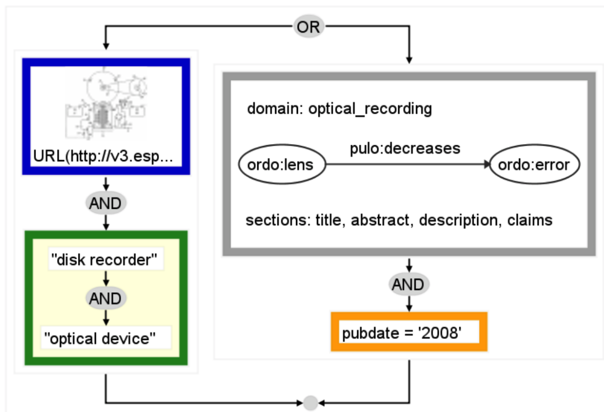


Figure 19: The graphical representation of a search query. PatViz allows the user to mix a search regarding keywords (green), metadata (orange) and also image similarity (blue).(Koch 10)

The evolutions of the visual interface of PatViz are numerous, and it is interesting to notice the 3D visualisation of (Giereth 08) was not kept during the development process. The visual interface is now composed of eleven windows, each containing a different visualisation. The visualisations that are left are the result of a longer reflexion, including on the data judged relevant for examiners. Thus we will examine each of these windows and analyse their pros and cons.

- Patent Graph: a configurable graph view that shows various connections between entities of the result set
- World Map: a distribution of the patent documents over the filing countries. The user can zoom in for a higher precision and not miss countries where patents in the search result were filled in. While map seems interesting at first sight, and we believe might feel enjoyable to use, justifying such a large space for data that can fit a very small table seems odd within this context.
- 3D IPC Treemap: This 3D treemap shows the IPC classification of resulting patents. We do not believe this view brings a real value to the visualisation, as the perspective makes it harder for the user to properly see the links from one classification to the others.

- 2D IPC Treemap The same treemap as the 3D IPC treemap but reduced to 2 dimensions. It is more readable than the 3D treemap.
- Aggregation Tree: This tree view shows the classification .
- Text View: A viewer for patent document texts. This view only contains text and is thus time consuming for the user to analyse a patent.
- Term Cloud: A cloud of words displaying the most frequent terms. Next to the alphabetically ordered terms the number of appearances is displayed. This tool provides a quick way for the user to understand the content of a patent quickly by overlaying them together. Still, the use of displaying the number of appearances of the terms seems very limited as there is no need for precise data here, the importance of a term already conveyed by the size of each word.
- Geo-Timeline: A scatterplot showing for each patent the filing country on the y-axis, along with the filing date on the x-axis. While showing the filling date of the results is interesting to ensure the examiner the query result is not too old or too young to be considered relevant, the random position of a patent on the y-axis can lead to overlay and does not help to estimate the quantity of patents at a specific time and country.
- Bar Charts: A bar chart to indicate the choosable metadata field. This visualisation seems to present very little interest as we can see it only displays five metadata fields while there are a very large number of them. This visualisation would be better off without this window as it is difficult to use, read, and takes space which other windows could benefit from.
- Table: A table containing the most important data of the patent documents like number, title, and applicant. While the displayed data is highly rich to the examiners, the table is very large when looking at the result of a search query.
- Selection Management: A graph based tool to store, combine and adjust selections. This tool is especially interesting as it points one need: even once a selection seems interesting, an analysis of small selections allow the user to determine with higher accuracy the quality of a patent with a better efficiency by using the visualisations with less elements to deal with.

2.4 Tactile interactions with abstract objects

We stated earlier the issues with interacting with a large amount of documents, and how the current tools at the European patent office are time consuming for the interactions.

Tactile interactions can bring more natural and richer ways to navigate through data and metadata. Even though our focus was not on the tactile interactions,

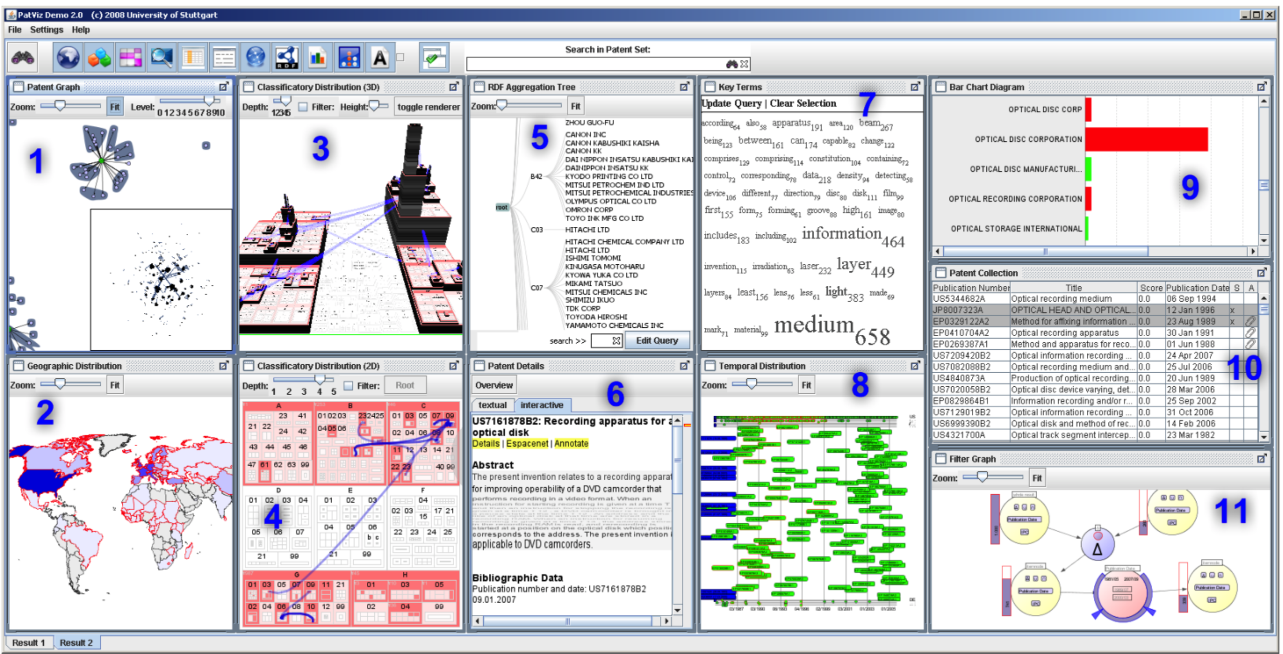


Figure 20: The views of the PatViz desktop, from top to bottom and from left to right: 1. Patent Graph; 2. World Map; 3. 3D IPC Treemap; 4. 3D IPC Treemap; 5. Aggregation Tree; 6. Text View; 7. Term Cloud; 8. Geo-Timeline; 9. Bar Charts; 10. Table of important data; 11. Selection Management. These parts are described with precision in section 2.3. (Koch 10)

it is interesting to look at the state of the art concerning them. While our interactions are relatively simple, the analysis of scientific literature allowed us to design a prototype that could be extended and enriched with new and original interactions. In this section we will quickly describe the papers that allowed us to set boundaries that should allow for multi-touch extension without needing to sacrifice our contributions.
(Schmidt 10) describes touch interactions with a node-link diagram, and especially touch input to allow the user to interact with the nodes or the links. These interactions allow for the user to get rid (at least temporarely) of the overload of the node-link diagram in order to interpret without error the visualisation.
(Schmidt 10) presents TouchStrumming, consists on touching a node creating after action a visible vibration of all the links related to it, or by directly interacting with a link, vibrating after interaction.
Another interesting point presented in (Schmidt 10) is PushLens, where by touching a node, the user creates a zone where only links of the current node are allowed. The other are deviated in order. This allows the user to focus more easily the beginning and end points of each link. Nevertheless, this method only removes nodes overlay at the start of each link, and overlay is still possible past the zone forbidden to other links.

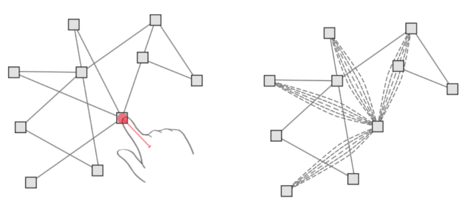


Figure 21: The effect of interacting on a node with TouchStrumming. The vibrations allows the user to efficiently determine the ending node of each links of the touched node. (Schmidt 10)

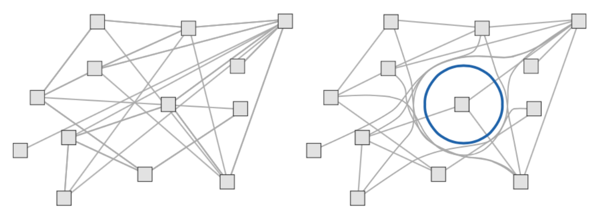
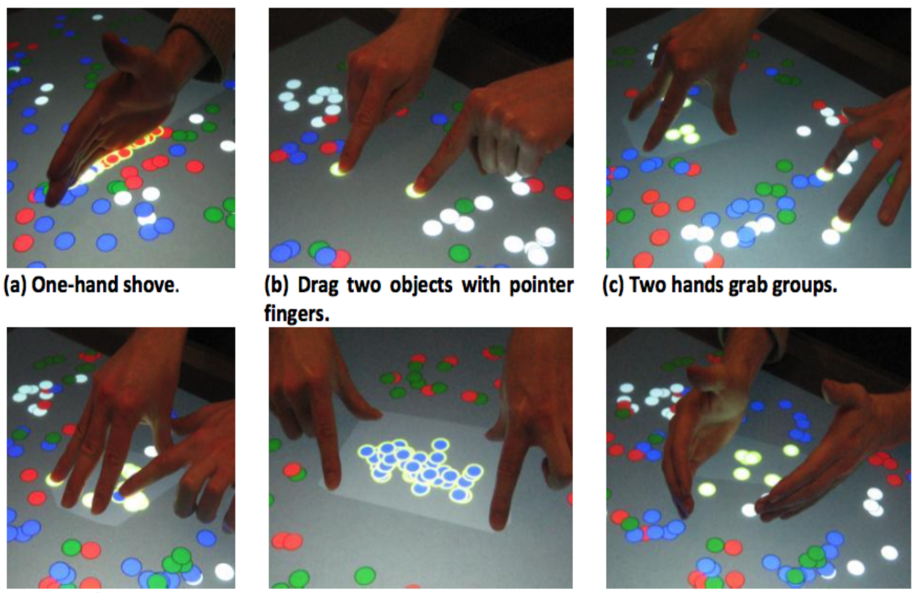


Figure 22: PushLens allows the user to remove overlay of unrelated lines around a selected node. (Schmidt 10)

(North 09) is a study on the efficiency of multi-touch interaction compared to classic mouse interaction and physical objects. In order to establish the most efficient interaction method, the users performed a sorting of circles that they had to cluster together according to color as fast as possible. During this study, the users were interacting with the same software except for the hardware. We invite the user to read (North 09) if they desire to discover the whole design of this prototype. It is curious to notice that the fastest interaction method was the one based on physical objects, followed by multi-touch interactions, and last were classic mouse interactions. There are two specific points we want to focus on:

- The users lost time with the multi-touch surface as they were, without indications, unclear about the set of actions they could perform, which showed that users do not go through the same reflexion process about interacting with tactile screens.

- The users were more likely to try tactile interactions using multiple fingers at once after the first test with physical object, and resulted with faster results than users that started with the mouse or multi-touch screen. While the consequences of such results are still unclear, we can presume it indicates that reminding the user of real-based paradigms can push the user to naturally try more complicated interactions.



(a) One-hand shove. (b) Drag two objects with pointer fingers. (c) Two hands grab groups.

(d) Add/remove from selection. (e) Two-hand transport. (f) Both hands coalesce large group to small.

Figure 23: Interactions user tried to sort colors circles on a multi-touch device without instructions during the study of (North 09). The users were more imaginative on the multi-touch device after performing a similar task with physical objects.

(Bederson 10) explains the benefits of the creation of Zoomable User Interface (ZUI) within some restrictions, as he underlines it is a great solution for specific solutions but can not be adapted to any problem. Thus, we decided to limit the zoom functionalities as when it seemed necessary, and focused the rest of the design in solutions that did not need the user to scale them. (Bederson 10) also underlines that they should be used smartly as they are time consuming for the user. Nevertheless, the author notes a surprising result: the user is more efficient even though animations are time consuming, the number of errors is re-

duced when including the animations. Finally, (Bederson 10) warns the reader when designing a ZUI that visual overviews always require significant visual search to find something, and thus, to remember that finding back data in a large collection, even when zoom is available, always requires reflexion from the user.
Another interesting visualisation in (Fanea 05) is presenting a system of 2D parallel coordinates extended into 3D coordinates. While not relevant with ordinal data, this part can be used to improve visualisation on metadata when selecting a folder.

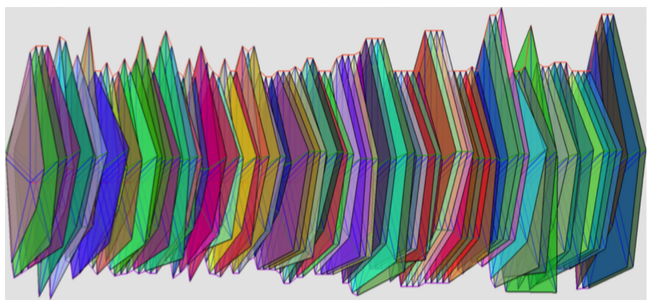


Figure 24: 3D representation of parallel coordinates. Here the data representes the same object evolving through time. The height allows to visualise point of interest for data different from the group. (Fanea 05)

3 System requirements and design

This part describes the philosophy that drove us towards our choices, the system requirements that were set by applying our way of thinking with our objective described previously, and gives an overview of the global organisation of the prototype.
For the process of document triage, two key factors shadow the others: the efficiency and the speed. While it is common to try to have a process as balanced as possible, but while the EPO can afford to make inventors wait for their verification, it is critical if a mistake is made and a patent is granted while it shouldn't have been, thus the balance in between speed and efficiency is tilted to efficiency. The goal of our new visualisation is to make use of the spatial memory of the user to help him take into its reasoning the keywords-based relations within the organisation of a collection of patents.
Designing the prototype to work efficiently on a large multi-touch device brought many questions to answer concerning interaction. The position of the screen was set up already when the work began, the screen is put against the wall, and the user has to stand to use it. Thus, the need to have as little input as possible is important, because the user can suffer from arm stiffness in such a situation.
As described earlier, it is necessary to limit the amount of text in order to reduce the reading time necessary for the user during document triage. This drove many of the following choices we describe in the later sections, especially section 3.5.
Our work was heavily influenced by the work of (Marijn 12), which itself is based on the work of (Ridder 11), and our requirements are similar, the differences come in our way to tackle the issues met during the document triage process.

3.1 Requirements

- Visualize a collection of patents
- Cluster patents in a two dimensionally space to display similarity in between group of patents according to a keyword-based K-means algorithm
- Display and filter group of patents
- Display implicit and explicit relationships between patents
- Offer tactile functions to interact with patents and patents metadata

3.2 Creation process of the metaphor

Establishing how to fulfil our requirements took us quite a bit of time, as we went through several ideas. (Agarawala 06) and his virtual desktop highly drove us towards the metaphor of an office, with interactions that feel close to the reality. Nevertheless, it was important to keep the advantages of the virtual environment too.(Giereth 08) and the several visualisations used to display data and metadata of a patent, were a proof of that.

The focus towards a close-to-reality interaction first drove us into the creation of a 3D environment representing an enhanced desktop. Even though a first 3D prototype was already started, we decided, for reasons described within the appendix, to provide a 2D solution.

Modifying our project into 2 dimensions implied many implementations changes, along with design choices. Changing from cubes to squares (precisely widgets) was obvious, but it forced us to rethink the global design of the prototype as well as its interactions.
It is also very important to underline one choice we made: the whole prototype works without any typing from the user. This choice has a huge impact on the design of the prototype, but it appeared as necessary. We think working with a portable keyboard is a solution that could be investigated.

We tried to keep the prototype light in the number of buttons displayed at once to push them to try it and thus help the learning curve.

When the user starts the program, he is facing three main blocks: the widget scene at the top center of the screen, the workspace, grilled, at the bottom left of the screen, and the widget visualisation, at the bottom right of the screen (cf Figure 26). It is also important to notify the small widget at the most left in the bottom, the widget legend: this widget displays information explaining the meaning of the color of each patent in the scene.
On the left, there are small buttons, which allow spatial organisation, box selection, reload of the search query, saving and loading of the workspace and the widget menu. These will be described in detail in Section 5. We believe the small number of buttons solves the problems of complicated interactions stated previously with the current tools of the EPO. The functioning of the buttons will be described in Section 5, but we already want to note the importance of dropping text for icons with the buttons. While we do worry the meaning of each button might not always be completely clear at first use, we believe an experienced user can more quickly recognize the meaning of an icon.
Dimensions of each of the main blocks (scene, workspace, and widget visualisation) were a difficult choice to make, as it was based on unrelated criteria:

- The number of patents returned by the search query : If a small amount of patents is returned by the search query, it implies a small need of space for the scene.
- The number of patents saved within a folder : The number of patents saved within a folder is critical for our group visualisation of patents saved. Many choices about data and metadata displayed, or not, were based on the available space for the widget Visualisation.
- The examiners' habits concerning the number of folders they usually will

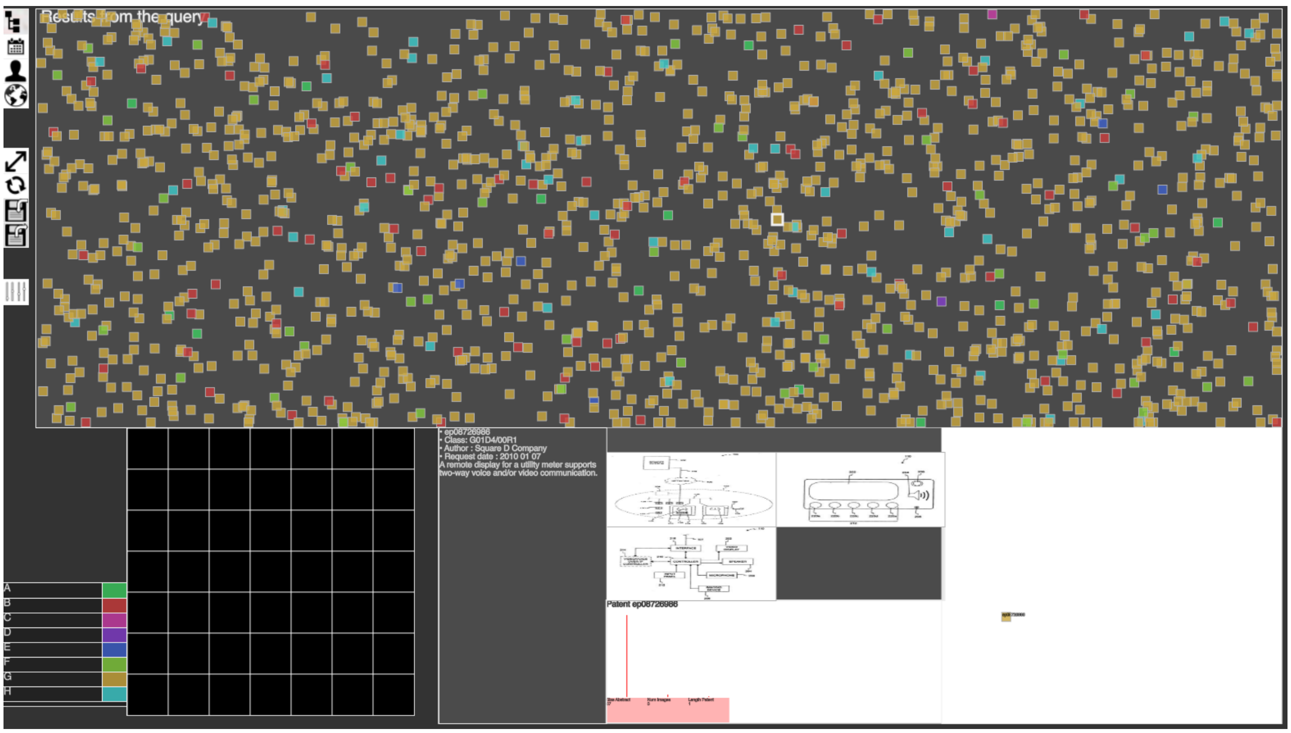


Figure 25: The prototype when the user starts it. Each widget represents a patent. The view is first unorganised for the user to keep it simple. The large widget on the middle of the screen is the scene, where the results of the search query are displayed. The grilled widget in the bottom left is the workspace, area to save patents and create folders, and the widget in the bottom right is the widget visualisation, used to display data and metadata of a patent or a folder.

make during their document triage process : If the user is used to create many folders during the document triage process, then we need a large space for him to order them. Users habits are heavily random, thus we have to go for the safe choice and offer a large grid of 49 spaces where the user can save a patent or a selection of patents.

- Our desire to keep the software usable for less large multi-touch devices, or even regular devices: Thanks to the important size of the screen we were working on, displaying a large amount of widgets was easy without losing the user's attention because of difficulty to discern them. Still, adaptability on a less large screen implies widgets big enough for the user to easily visualise and interact with the widgets. It is especially important within the scene and the widget visualisation.

The issues listed above are restricting one another. We can not ensure a situation where everything is displayed at once. Still, working with a multi-touch device allows the user to modify size of widgets, and thus gives him more freedom concerning the visualisation. We made a choice that works with all devices, but especially with a multi-touch device. The only one of the three main blocks that can be moved and scaled is the widget visualisation. While the scaling is impossible with a classic configuration of keyboard and mouse, this issue is not a big problem with a large screen, and folders do not contain a too high number of patents. On a tactile screen, even with a large number of patents within a folder, it is always possible to scale the images at will, ensuring better visualisation. The widget visualisation is rich in content and will be described in a later section, but it is important to note that because it is the richest widget for analysis, the choice of providing freedom to the user on only this widget came naturally.
Finally, we want to underline the highlight brought on patents within the scene for the user to keep track of them and the stage of interaction they are at (more details about interactions in Section 5).

- The last patent touched by the user will be surround by a bright white square, so that the user can easily distinguish it
- A selected patent is surrounded by a white (if last selected) or grey square.
- When selecting a folder, the patents in the scene are surrounded by a black square. This way, the user does not have to go through reanalysis of these patents. It is important to keep in mind though that selecting a folder means selecting the patents within it.

3.3 The Widget Scene

The scene is the space where each widget represents a patent returned by the search query. The design of this space was inspired by (Agarawala 06) as we

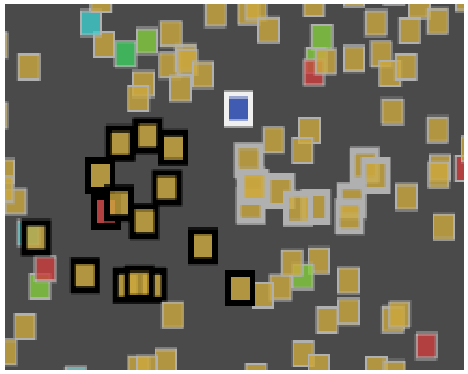


Figure 26: Color surrounding the patents indicate their state: white is the last selected by the user, grey is selected, and black is within a folder selected (which also counts the black patents as selected).

first focused on providing an experience similar to the one provided by a desktop. (Marijn 12) offers the possibility to create a group of patents, as a stash of papers, like (Agarawala 06), while we create folders. We decided to create a specific space (the workspace) for this action, as users often put apart the folders, and the patents as they can be considered as still part of a work in progress.
Some buttons spatially and visually reorganise the patents within the scene (details in Section 5). The user can interact with a patent by tapping or dragging it, but we decided for the patents in the scene to always be pushed back to their position pre interaction, in order to keep the meaning of the spatial ordering. The animations to push back a patent to its position is very simple : at each frame, the velocity of the widget that is not at its previously calculated rightful position is set to a constant positive value and the direction goes from the current position of the widget to the calculated one. This can result in the widget being pushed farther than its calculated position, and thus time for animation longer, but examiners enjoyed the animation, and more importantly (Bederson 10) reminds us that even though they are at first time consuming, animations results in less mistakes in interactions, and thus more efficient interactions, thus we decided not to put our focus on this issue.

3.4 The Widget Workspace

The workspace is a grilled widget for the user to think about the relevance about what he wants to organise patents that seemed interesting, but is yet not sure about their quality. Each square is a spot for the user to either put a patent, or a selection of patents, resulting in a folder. Once put in the workspace, the user can not change the spot of a patent or a folder. This is done to ensure the safety of spatial memory. If a user considers a patent or a folder irrelevant, he can destroy them, which keeps the state of the patents in the scene.

It is possible, within this space, to append a patent or a selection of patents, or a folder to a patent in a spot, creating a folder of the containing all the patents in the selection. Interactions within the workspace are explained in Section 6. Because of the color code already exisiting in the scene, we decided to keep the folder grey, in order for the user not to misinterpret the color code as a signification of the data or metadata of the patents contained within the folders.
Still, another important point we had to decide on was if we wanted to keep a stack metaphor like (Marijn 12). While we do believe in the interest of this method when the user considers all of the scene as his desktop, we took another direction for our metaphor, as our distinction from the specific elements the user wants to work on are clearly separated. Plus, in our case, clicking on the folder highlights the patents within the scene, providing metadata information, which is an advantage. The final argument for creating a stack was for the user to have a hint about the number of elements of a stack thanks to the number of patents visible "coming out of the stack", and also the height and width of the stack. The size of the stack can not be properly estimated from the number of patents "coming out", and our workspace forces to have a limit on the size of the stack. Thus, we believe it was for the best not to use stacks in our prototype.

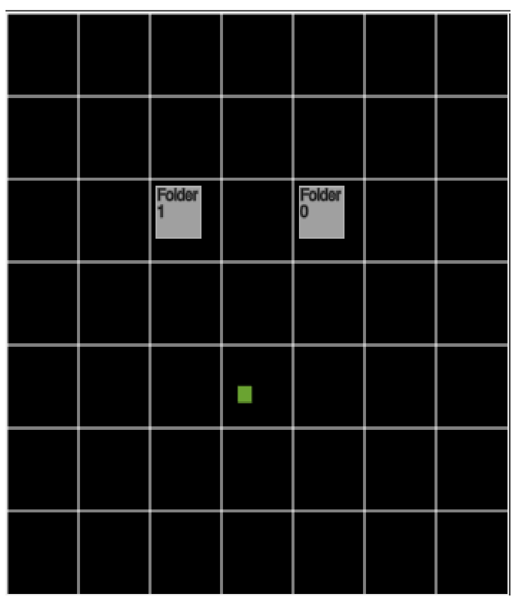


Figure 27: The workspace, where each spot can contain a patent or a folder. Here the user has saved two folders and one patent.

3.5 The Widget Visualisation

Displaying data and metadata of a selection, whether it is a patent or a folder, is a critical step in evaluating its value. Currently, the Viewer at the European Patent Office displays a lot of text at once, which is time consuming, interactions with images are not comfortable as the user needs to use a mouse and many clicks to interact with them, and finally the Viewer does not provide information about elements within folders.
Thus, we need to display important textual information, need to display the images, need to display metadata about the patent, and if the last selected element is a folder, the relations in between the patents within it. To do so, the widget visualisation is cut into four smaller widgets: the widget text, the widget images, the widget metadata, and finally the widget clusters. triage process.

- The widget text :

  - After selecting a patent : The widget text displays text for data the examiners we talked with consider critical for the patent : the application number, which we also call identifier, the classification, the applicant (called author in the visualisation for simplicity), the publication date, and the notes taken by examiner.
  - Folder : Identifier number of the folder, and the number of patents contained within it.
    It is also important to consider the representations of the patents within the widget clusters. These widgets are the same have the same colours as the widgets from the scene, but clicking on them when a folder is last selected displays data and metadata about the relations with other patents in the other folders. We call these representations of the patents within a folder clones.
  - Clone: Identifier number of the folder, identifier of patent, identifiers and list of matching keywords in between this patent and the ones with whom the selected clone share keywords.

- The widget images : (cf Figure 29) This widget displays the images contained in the last selected object. If the last selected object is a patent, then all of its images are displayed, and if it is a folder, all the images of all the patents contained within the folder are displayed. We order the images in a square arranged array.

- The widget metadata : The widget metadata is a widget displaying the following metadata for a patent : size of the description in characters, number of images contained in the patent, and length of the patent in characters. These metadata are similarly calculated with a folder, but we display only the average. While the interest of this feature on the visualisation aspect is not very important, keeping the space for this widget offers the possibility to reveal more relevant metadata for future work.

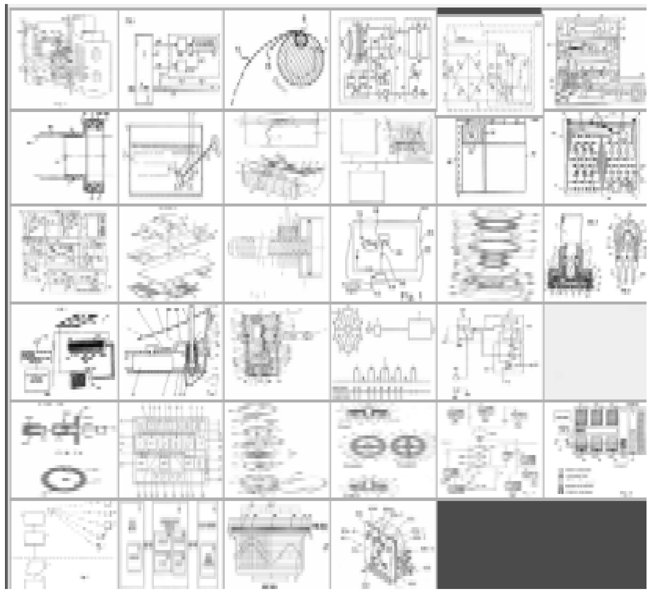


Figure 28: The widget images displays all the images of the last selected patent or folder. The square root of the number of images defines the number of images per line.

The height of the bars is calculated as follows: the highest value among the length of the description, number of images, or length of the whole patent, get all the allowed space as a height, and thus this creates a ratio among values for the other lines to be drawn.
The main interest in this visualisation resides in giving the user to quickly decide, according to the ratio of the description and the whole patent, if reading the whole patent is too time consuming. Even though in our case, the data is not completed, we believe this is very important for future work and thus keeping this was necessary. Another point was inspired by (Kandel 12) analysis tool for assessing data quality issues. While it is not the case with recent patents, some old patents can be stored unproperly within the EPO database, and it is important for the user to see missing data that might degrade the quality of his work. This can easily be seen if the length of a patent is 0.

- The widget clusters : This widget is composed of copies, or clones, of the widgets contained in the latest object selected, whether it is a patent or a folder. If the user last selected a patent in the scene, the widget clusters simply shows a widget with the same color as the original, and the identifier being drawn upon it. If the last selected object is a folder, the widget clusters thus contains copies, also called clones, of each widget representing patents contained within the folder.

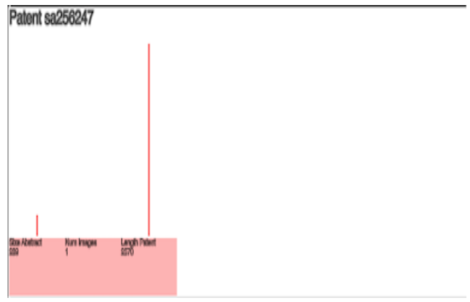


Figure 29: The widget metadata. The lines represent the value of the length of the description, the number of images, and the length of the whole patent. In the case of a folder selected, the average is displayed.

The clones representing the patents are grouped into circles representing

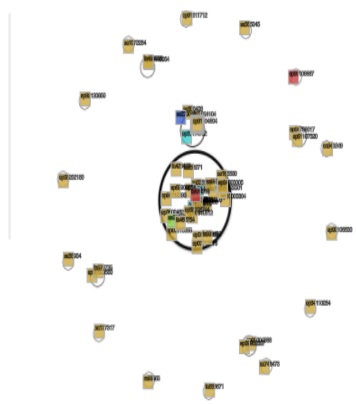


Figure 30: The widget clusters. Each widget represents a copy of the selection. Here a folder was selected, and each widget represents a patent inside the folder. The patents within the same circle are similar according to the keywords-based K-means algorithm, described in Section 5.

their clusters. The algorithm producing the clusters and their spatial organisation will be explained in Section 5. Nevertheless, the design is to be noted. We decided to keep the patents within simple black circles, with only shades of grey to describe the connections in between the patents, in order not to have conflict with the colors of the patents. Because of a lack of time, we only implemented straight lines instead of convex hulls, yet we believe that by giving the user the possibility to move around the clones, he can get a good understanding of the destination of each of these lines. (Dunne 12) proposes a visualisation of communities in an interesting way, as relations are represented by arrows,instead of lines, each paper

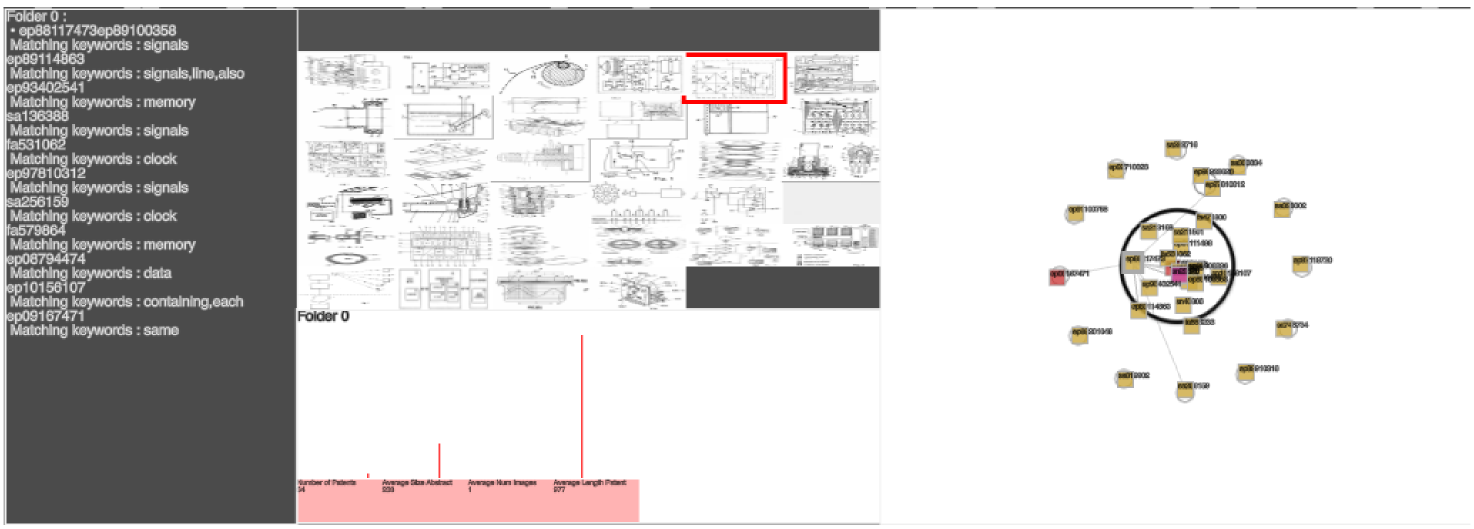


Figure 31: The widget visualisation after selecting a folder, then a clone. The selected clone is highlighted in grey in the widget clusters, with lines drawned to the other patents which share common keywords. The darker the line, the more important the number of patents they share. The image of the patent is highlighted in red in order to keep track of it easily. In the widget text, the related patents with the common keywords are written in a list.

referencing another one. Unfortunately, the ambition of drawing all the relations at once makes it difficult to analyze the citations precisely. Thus, we decided to only display relations from the last selected clone to related patents in the widget clusters. If the user moves clones, and then selects a folder, the clones are put back within the circle they belong to, for the user can not to misinterpret the position of a clone.

3.6 Conclusions

We based the requirements of our program on a combination of the expectations and issues faced by the examiners at the European Patent Office, which gave us boundaries and objectives. Document triage process being complex, we solved the issue of separating the several steps the user go through using different spaces: the scene, the workspace, and the widget visualisation.
Designing a prototype on a large multi-touch device implies to have an original approach, regarding the interactions, which have to be natural and few, and also regarding the presentation, which can display many elements at once.

4 Tactile interactions on a large screen

Multi-touch interactions offer to the user a larger and more natural set of possibilities than the classic mouse. Tactile interactions are now wildly used by billions of people everyday that own smart phones. This provides an opportunity to enhance many interactions still done with mouses on computers. Nevertheless, desktop sized and large multi-touch screen are not widely spread, and the users are used to the interactions they find on small touch devices: dragging, zooming and rotating. Phones are limited in the number of fingers they can follow at once, and because of the size of the screen, augmenting the number of input points is not something users find necessary. Larger multi-touch devices, whether we talk about regular desktop size or larger ones, are more expensive and thus are not widely available for the users. In the mean time, companies, like the European Patent Office, do see this opportunity and want to be up to date with the latest technologies. Thus our project was focused for a multi-touch large device, offering us new problematics and possibilities. In this section we shall describe the advantages and inconveniences of a large multi-touch screen, the hardware we worked on with its possibilities and issues, and how we designed our tactile interactions according to it. The interactions are a crucial element for any software where the user has to produce intense reasoning, as the more natural interactions are, the more the user can focus on his task. Multi-touch interaction are a set up many users are used to thanks to the wide use of smart phones and tablets in the world. It is already recognised as a well known and intuitive way of interacting with a program (Marijn 12). As stated earlier, the most important aspects of our prototype is for the user to be able to browse and interact with a large collection of patents. We make the hypothesis a large multi-touch device with classic tactile interaction is an efficient combination for it.

4.1 The advantages and inconveniences of a large multi-touch screen

Our project was made for a large 70 inch multi-touch device. The size of the screen allows for the display of many elements at once, and also allows to do presentations easily to a group of people. Nevertheless, the size of the screen bring in many issues, such as the positions available for the screen. Such a large screen is limited spatially: it can be set up vertically or horizontally, but it can not viably be set in inclined, as the movements for the users would be very big and some areas of the screen might be impossible to reach for some users. Thus only vertical and horizontal positions seem convenient. Unfortunately, a better analysis of the dimensions of our screen show that it has a width of 1 meter, which while reachable for us, does not feel comfortable, and close to impossible to some users. Thus we set the screen in the vertical position, fixed against a wall, at a height designed for the user to interact while standing. While

Figure 32: Our prototype was developed to work on a CTOUCH Laser screen with a diagonal size of 70 inches (1778mm). The user has to stand in front of the screen to interact with it.

the interactions are this way easy to proceed, it is interesting to notice that interactions have to be designed in order to be relatively short so that the user does not feel arm tiredness after keeping it stretched for long.

4.2 Tactile design

The multi-touch screen we use within our project is capable of handling inputs of up to 10 fingers at once. Unfortunately, licences issues, described in the appendix, block from using more than 3 fingers at once, and another issue is that the functions that are supposed to take one tap as input only react to two quick taps.
In this subsection, we will describe how the user can physically interact with the screen in order to execute actions described in Section 4.
Since we were limited within our number of fingers used at once, we implemented interactions that would be easy for the user to understand. It is also designed so that the attention of the user is focused on the data, and not on the interaction. Thus, our interactions are based on a mix of classical copies of interactions within a virtual desktop: tap (for a click), drag-and-drop, and box selection, and classical interactions taken from smart phones: zoom and rotation.

- Tap: The user has to double tap an object to select it. If he does so twice, a menu of actions appears on top of it. While this is highly common, it is interesting to keep in mind we need objects to be big enough on the tactile screen for the user not to need many tries to successfully select an object.

- Drag-and-drop: Pressing the finger down over an object, keep the finger pressed and release it at a specific position to trigger an action.

- Box selection: The box selection is an interaction inspired by virtual desktops. The principle is simple, the user selects this mode of selection (the icon is highlighted as long as it is selected) and then taps twice to set up the points that will define the selection. We save the x and y position of both taps, and then select all the patents horizontally in between the two values x, and vertically in between the two values y.

Figure 33: The icon of the box selection.

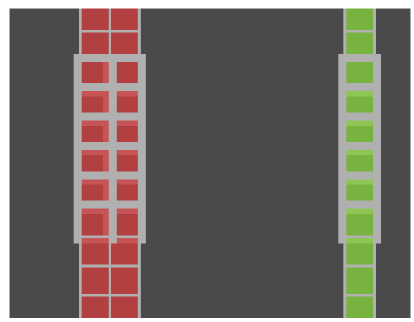


Figure 34: Here the we can imagine positions the user tapped to select these patents.

- Zoom: The zoom is an interaction users are completely used to nowadays and naturally try to use when facing a tactile interface. It works by pressing two fingers at the same time either closer to each other to zoom, or further to each other to unzoom.

- Rotation: Rotation is another interaction users are accustomed to thanks to smart phones. By pressing both fingers at once, and lifting one up and the other one down. The direction of the rotation is going left if the right finger goes up at the same time the left finger goes down.

5 Interactions and visualisations for reflexion on the scene patents.

The previous section described the design choices we did for our prototype. Another important point is the need for the user to smoothly interact with the data and metadata, navigate through the patents in order not to feel restrained by the software. The more natural and rich the interactions, the easier the reflexion on the content for the user.
In this section, we will describe the design choices and functionalities of all the buttons of our prototype allowing the user to interact with the patents within the scene, and the widget menu in section 4.6. Our inputs for interactions were designed to aim for the following objectives:

- Small number of input for interactions
- Avoid overlay of data as much as possible
- If overlay happens, always allow the user to set back the prototype to a situation without it.
- Imply possibilities of actions with icons rather than text

The elements within the scene convey metadata about the search result to the user thanks to their colours and positions. While the patents are randomly positioned when the user starts the program, it is only for him to first establish an order of size for the number of patents returned. We decided never to display the exact number of patents displayed in the scene as it is just a large number, and we believe if the number is too high, the user is likely to lose motivation and go again through the process of editing a query, which can be time consuming and brings risks of skipping a document relevant to the document triage process.
In order to offer tools precise selection of the results from the search query, a widget, called the Widget Menu, allows the user to create precise filters on the classification, date of publication, authors, and languages. This widget is described in Section 5.6.

5.1 Interactions with a collection of patents using buttons adapted to the selection

When the user presses twice on a patent, folder, or clone, a set of actions appears. These actions are represented and the ones available to the user depend of the object first selected.

Our objective is for the user to rapidly understand the action performed when pressing an icon, and let the icons be recognisable and memorable. For a fast learning of the use of the prototype that the icons are reused, even though the action might be slightly different according to the context. Easily identifiable icons implies a gain of time not wondering the consequences of pressing them,

Figure 35: Buttons after double clicking on a patent widget.

Figure 36: Buttons after double clicking on a folder widget.

Figure 37: Buttons after double clicking on a clone widget.

thus we used icons the user is likely to know from another program, or tried to make them self-explanatory.
The number of actions the user can do to an element depends of the nature of the element. While the actions are the same or very similar no matter the object, some actions are not available to clones, and some actions are only available to patents. Thus, we invite the user to look at figure 31 from left to right to see all the buttons possible. We will describe them from left to right, with changes according to the element selected before their appearance.

- The unselect button: Available for patents and folders, it unselects the last touched element.
- The unselect selection button: Button also available for patents and folders, it unselects all the elements previously selected, visible by the fact they were highlighted, and no longer after pressing this button.
- The delete button: This button is available for patents, folders, and clones, but actions will be slightly different: after selecting a patent, it removes it from the scene; after selecting a folder, it removes it plus all the patents within it from the scene; and when last object selected is a clone, it removes the clone from the widget clusters, and removes the patent from the folder structure.
- The select only button: Available for patents, folders and clones, it unselects all the objects except the last touched patent or patents contained within the last touched folder, or the patent the clone represents. This functions is quite useful after clicking on a clone as it allows to quickly display the patent metadata by interacting with the clone, and thus quickly estimate its value.
- The keyword button: This button is only available for patents. Pressing this button displays all around text widgets with the keywords of the patents written. Organising the keywords around the icon often implies some of the keywords are above the screen limit and thus the user can not see them. The user can drag the whole group by dragging the keyword icon, and then look at all the keywords. When the user selects one of the keywords, all the patents that share this keyword are then selected. The process to generate the keywords is extremely simple, we simply take the short description, which is already made of well thought of terms, and filter them. The EPOQUE database provides a list of terms used to filter the databases. We remove these terms from the words and then assign

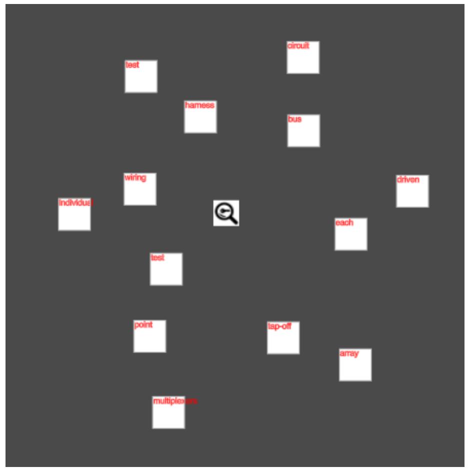


Figure 38: The result after selecting the keyword button. Each white widget have written one of the keywords of the patent. Selecting a keyword will select all the patents sharing this keyword.

each word remaining as a keyword of the patent.

The interactions described above are designed to solve problematics established earlier: how can we provide natural non restrictive interactions, how can we create interactions without going through a time consuming tree menu, and how can we ensure the user remembers the interactions accessing to the functionalities implemented ?
We solve these challenges by providing tax tiles interactions with dynamic display of buttons, use of intuitive and well known icons, and tactile interactions to provide a certain freedom for the user to organize the widgets within the program.

5.2 Organising according to classification

IPC classification of patents can be an important metadata for the examiners during their document triage process. For example, it is very likely that patents that are part of an unexpected classification among the query results are a misuse of a term either from an imprecise search query or an ill written patent. The colour of the patents depend of the classification they are contained within. If a patent is contained within several classifications, the first one given by the file, the most important one, is used. As stated earlier, the IPC is a tree structure. A complete classification would be for example A61K31/10. We decided to only separate the patents according to the first level of the IPC, that is separating patents from classification A and B, and not A01 and A02 for example. The user can see the classification matching each colour by looking at

the widget legend. This way, we assign for each first level of the IPC a distinct hue. While we believe some solutions can be implemented to inform quickly the user of a patent's classification, we ensure with this method for the user does not to misread metadata, and this solution allows animations to move to other positions according to the metadata interesting the user.
This method allows to evaluate the proportion of each classification among the results, and can quickly notice minority among the classifications which are likely to be outside the scope of interest of his search.

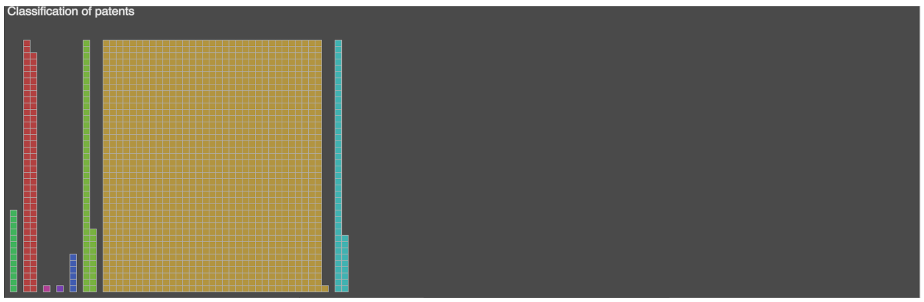


Figure 39: Patents spatially organised next to each other according to their classification. An unique hue for each first level of the IPC easily indicates the classification of a patent. This method allows to evaluate the proportion of each classification among the results, and immediately notice patents that hold an unexpected classification.

5.3 Chronological organisation

The date of publication is of the highest importance as a patent can not be granted if the same invention was already published. It is also sometimes irrelevant for the examiner to look at very old patents as they are likely to present technology that is no longer relevant.
The spatial position of the patents is based along the scene on the x-axis, where patents on the left are old, and patents on the right are recent. The patents are colored according to their publication, with the widgets of the oldest patents resulting the search query being completely black, and the widgets of the most recent patents are bright red. The transition is done linearly by changing the value of the HSV colour of each widget. While this view does not allow the user to precisely know the exact date of each patent in the scene, by looking at the widget legend the user can picture a time period precise to a few months of the date of publication of each patent. If the user desires a very precise time selection, the widget menu described in Section 5.6 allows to filter the scene.

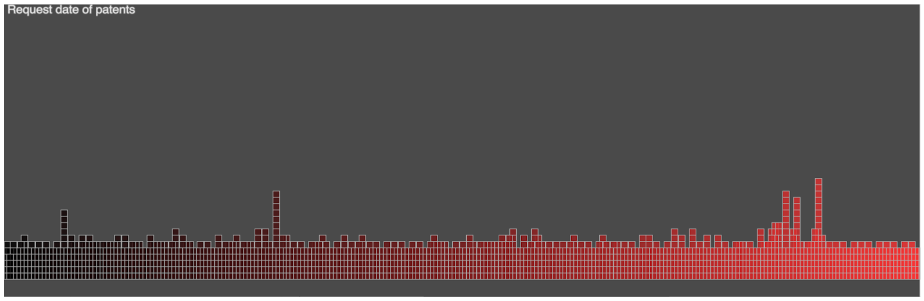


Figure 40: Patents spatially organised on the x-axis according to their publication date. Patents sharing the same date of publication are set up on top of each other to form a histogram. The user can thus also estimate an idea of the amount of patents published on a certain day. Black widgets on the left indicate the oldest patents and bright red widgets on the right indicate the most recent patents resulting the search query.

5.4 Focusing on the authors

As described earlier, the term author here actually describes the legal tenant of the patent. The point of this choice is to allow for grouping of widgets sharing the same author when the user selects the author organisation. This is very valuable information as some authors are focused on very specific fields of study and thus the examiner can establish what is the global area of a group of patents, and thus remove them if their field does not completely match the user's objective.
The widgets patents are organised alphabetically, and grouped together when they share the same author. Thus, each group has a different author, including groups of one widget patent only. In order for the user to rapidly investigate the authors that published the most of the search query result, we set up a color code. The color linearly changes from dark blue for authors that published a small number of patents in the result to bright blue for the most published ones.

5.5 Analysing available languages

In order to establish the state of the art, the examiners have to look through patents that can come from anywhere around the world. While all the European Patent Office examiners speak several languages (the official languages are English, French and German), the examiners often find japanese or chinese patents returned by the search query, and they have to translate them before establishing their value. It is thus interesting for the users to be able to separate the patents they can not understand directly, and take care of them later in the process. This pushed us to also create a spatial and visual organisation of the patents according to the language they are available in.

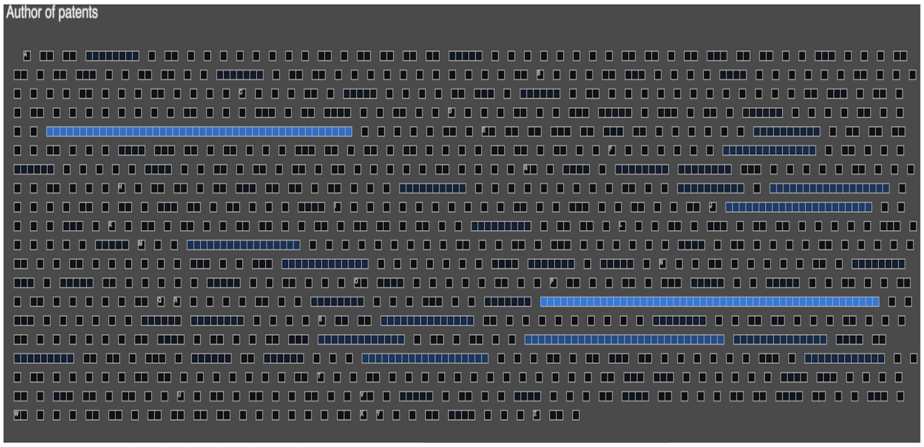


Figure 41: Patents are alphabetically ordered from left to right on each line. The visible letters on widgets indicate a change of the first letter of the author. Patents grouped together share the author, which means that each space indicates a change of author. Colours vary from dark blue to bright blue according to number of publications.

The spatial and visual organisation of the patents according to the language they are available in is very similar to the one when displaying patents according to the classification. Colour is coded by giving one specific hue for each language, and widgets are grouped according to the language for the user to have an order of idea concerning the languages of the result of the search query.

Figure 42: Patents visually and spatially according to the language they are available in. The hue values characterizing the colours are unique for each language. The patents are grouped together for the user to have a first insight about the result of the search query.

5.6 The Widget Menu

The menu selection and how it affects the remaining patents. Explain the colours and positions of each widgets. The buttons previously detailed help the user interact and get a good insight on the metadata of his search result. Still, if the user wants to proceed with the results in a methodical way in order to be certain not to miss any relevant patent. Thus, we must provide tools for precise filter of the scene and thus reduce the amount of data has to deal with at once. This motivation made us create the widget menu. If the user selects it, a widget appears with four tabs to precisely filter the patents according to the date, classification, author, and language. We will describe each of these tabs and the way the user can interact with them.

- Widget time: For the user to precisely select the time periods he wants our of the filter, the user has to move two widgets along a line to define the start and the end of the time filter the patents can go through. Thus, if one widget is positioned for 01/01/1995 and another widget is positioned for 31/12/1999, the user will select all the patents within 1995 to 1999 included. Since the number of possible dates is very big, even on a large multi-touch device, the user has to be very precise with his input during this selection. In order to reduce the precision, we decided to remove the dates where no patent is published within the query search result. This allows for larger spots for each time date. Still, future work includes dividing this idea into 6 widgets on 3 different lines to have widgets to select distinctly year, month and day.

- Widget classification: The widget classification is very simple, it simply contains widgets representing the first levels of the IPC tree the user can select to keep through the filtering. There is one widget for each possible first level of the IPC tree, and these widgets are highlighted when selected. If the user unselects one, it is no longer highlighted.

- Widget author: The widget author allows the author to select the patents that will go through the filter. There is one widget for each author, and since the number of authors can still be important, we decided to organise the widgets alphabetically, with the name of the author printed (even though there can be some overlay of text), and also we decided to implement the same color code as within the scene. If an author published many patents, the background color of his widget would be bright blue, while authors that published little were represented by a widget with a dark background. The bottom buttons work the same way as for the widget classification.

- Widget language: This widget is very similar to the widget classifications, but with the languages that can be found within the patents resulting the search query. Once all the selections are done, the user can press on filter, click again on the widget menu button, and see the filtered patents within the scene.

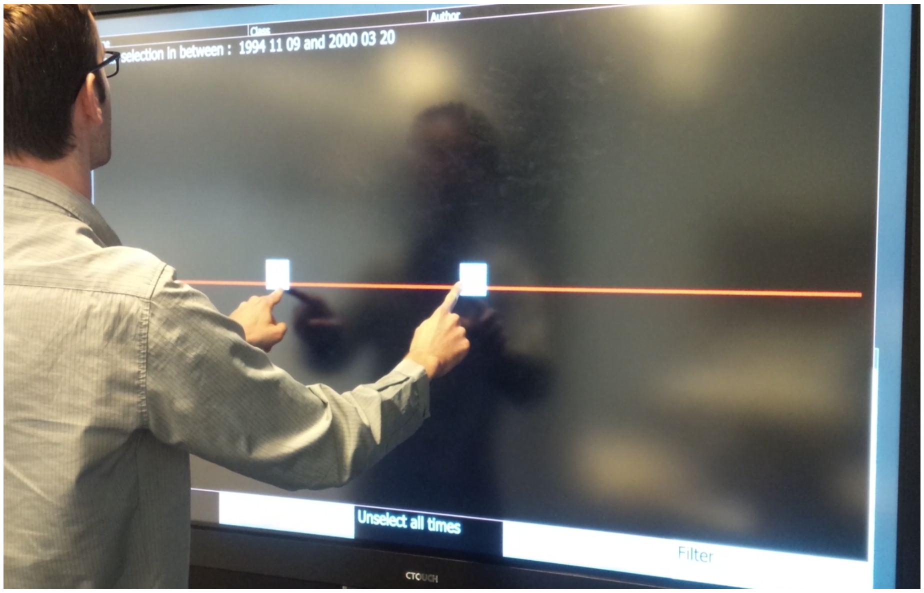


Figure 43: The time selection tab of the widget menu. The user can select the time selection he wants to stay within the scene after the filtering. There are selection indications on the top of the widget, and also on the two white widgets, to make sure the user can focus on a precise selection.

5.7 Saving a workspace and loading back

As we intended for the user to still feel free on his possibilities to work with his workspace, we also added two functions, save and load, accessible by pressing the icons of the floppy disk, with an arrow going in to indicate saving, and the one with an arrow going out to indicate loading. If the user wants to make some space within his workspace, but has found some patents or selections of patents he does not want to lose, he can press the button save. The patents and folders will be saved within the program, and the workspace will be emptied. The user can then keep on working, and when desiring to look back at his previous work, he can press the load icon, and the widgets will reappear. If the user used a space previously owned, the patent or folder that was there will go to the next space available on the right of its previous spot, and will go down a line if necessary.

5.8 Conclusions

Our objective was to provide natural interactions for the user to be able to use all his focus on the document triage process. To do so, we removed all text for the interactions, as reading is likely to stop the focus of the user and used icons instead, limited the number of icons for the user to remember them after his

Figure 44: The classification selection tab of the widget menu. The user selects the first level of the first level of the IPC tree he wants to save from the filtering. The buttons at the bottom allow the user to select all widgets or unselect all of them.

first use.
Several design decisions were taken for the interactions not to block the display of data, such as buttons outside of the scene for global interactions, and buttons appearing after double tapping located on top of all the widgets of the scene.
There is no solution for displaying all data available at once when dealing with a large number of documents. Thus, the interactions have to give solutions for the user to navigate through the documents. Several examples of our solutions for this type of issue are given, such as the possibility to move the keyword button, allowing the user to position the set of keywords where he can analyse them within good conditions, and the possibility to move and zoom and unzoom the widget visualisation and its images.
The user can select several visualisations to focus on different metadata he can find interesting in his research, and through a menu according to the level of precision he desires within the documents he wants to analyze. In order for the user to immediately understand the meaning of the menu, we keep the colour codes the user is used to from the scene within the menu. Thus, even though interaction is different, the colour codes associated with the highlight of selected elements involves fast selection over specific metadata the user finds relevant.

Figure 45: The author selection tab of the widget menu. Each author is represented by a widget ordered alphabetically with the background of the widget indicating the amount of published patents. A dark background indidcates few publications while a bright blue background indicates many publications.

Figure 46: The language selection tab of the widget menu. Available languages are represented by widgets. Once all the selections are made, the user can press on the filter button in the bottom right, press again on the widget menu, and see the filtered patents within the scene.

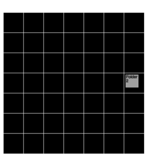


Figure 47: The user desires to save an interesting folder.

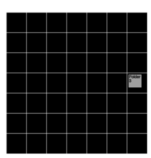


Figure 48: After saving, he uses the same space for another folder he works on.

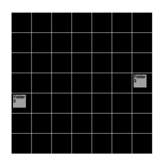


Figure 49: When pressing the load icon, the loaded folder moves through the workspace until it finds a free spot.

6 Visualisation of clusters within a selection

Clustering is a multi-objective optimization problem, it is a problem where the decision is based on multiple criteria. Clustering algorithms are heavily dependant on the parameters used to set them up, which are dependant of the nature of the data. It is most of the time unique to a problem. It is important to underline that clustering is a process of knowledge discovery which involves errors and trials, as quite often there are no perfect solutions to the problems solved.
The role of the visual clustering is crucial within our prototype, as it allows for the user to face a lot of metadata to understand the subject of several patents at once.
In order for our use of a clustering method to be justified, we need to think about these key questions:

1. When is it relevant for the document triage process to cluster elements ?
2. How should the clusters be represented to the user: as blocks, or displaying all the elements at once ?
3. Should the processus be entirely automatic or have the user interact with it ?

In order to explain the clustering of the patents within folders, we will first quickly explain our choice of design choices inviting to create clusters, how the user can creates folders, append elements to folders, and remove patents from folders. Then we will describe our method, based on K-means, to calculate the clusters of patents, and how we spatially organise them.

6.1 Creation of a folder and patents appending

Clustering can bring an overload of data, which implies to deal with it either several elements are grouped into one, which sometimes means losing the display of some data, or display all the elements at once, but this can bring issues to read them if they are hiding each other. Our focus towards the process of document triage is to focus on keywords-based relationships. If we allow clusters of a large collection of documents, it is very unlikely that the user can read it and visualise connections in between patents. Thus, we designed our prototype for the user to create folders according to criteria such as author, classification, or dates to obtain a relatively small folder of patents, which makes the clustering relevant. Once the folders are created, representing each elements within it is necessary for the user to be able to recognize precisely (as stated earlier, efficiency is critical in our case) each patent within the folder, and even within a cluster.
The interaction to create a folder is very easy, and results of our desire to separate the scene from the objects the want user wants to ponder. Thus, if the user wants to create a new folder containing the current selection, he simply has to drag one of the patents of the selection and drag it over a spot of the

workspace. When the user releases his finger, the folder is created. If the user wants to add another a patent or a selection of patents within the folder, he simply has to drag it over the folder spot within the workspace. It is also possible for the user to add the patents of a folder into another one, by dragging the folder we desire to add onto the space of the other one. If the user intends to filter the content of a folder, he can easily do so by double tapping on clones within the widget clusters, and select the delete icon. The patent related to the clone is still within the scene, but it is removed from the list of patents contained in the folder.

6.2 Keyword-based relationship to convey insight

Seeing patents grouped together allows the user to quickly grasp the meaning of a patent thanks to the metadata he reminds from the previously clicked and linked patent. The links in between the patents indicate they share keywords: a folder contains a list of patents, and each patent contains a list of keywords. As the computational cost to verify the shared keywords can become high with a folder containing many patents, the calculation is done once each time a folder is created, or when patents are added to a folder, or when patents are removed from a folder. This implementation is costly for the computer's memory, but the gain is higher than the loss. We save the links within a structure to allow faster interaction for the user. The folder thus contains metadata organised about the collections of patents it contains. The folder also contains a structure to save all the appearances of each keywords among all the patents it contains. This structure allows us to use the K-means algorithm to calculate clusters.

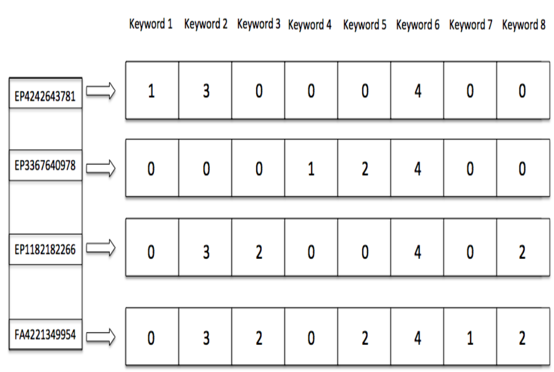


Figure 50: The data structure used in order to use the K-means clustering algorithm. The vector on the left is the input given to the K-means algorithm. Each element of this vector contains a vector as long as the total number of keywords within all the patents. The number of keywords thus defines the dimensions of the K-means algorithm.

6.3 Clustering a selection of patents according to their keywords using the K-mean algorithm.

The K-means algorithm is a classic algorithm for clustering. It takes a set of of feature vectors and separates it into a user defined number of clusters. In our prototype the feature vector is a vector as long as the total of keywords contained within the patents inside the folder. This vector indicates, for each index, the presence of the keyword specific to this index. If a keyword is not present, the value is 0 at this index, and else it is equal to the number of patents it belongs to. Thus, the dimensions of the K-means algorithm are equal to the total number of keywords within the folder.
K-meanss also requires the user to enter the number of clusters desired by the user. We do not allow the user to do this choice. Instead the square root of the number of keywords within all the selected patents combined. The square root of the number of keywords is selected in order for the number to adapt and grow with the size of the selection, but providing relevant clusters thanks to a small number.
The K-means algorithm is made of 4 different steps:

```
Step 1: The initialisation of the initial "means" which are randomly  chosen
points within the data domain.
Step 2: The K clusters are created by assigning each point to its closest
mean points.
Step 3: Calculate the new centroids of the clusters.
Step 4: If the mean of the distance beween the means and their respective
centroids is bigger than the threshold, return to step 2.
```

Considering K clusters, and N elements (the patents of the folder), the complexity of the K-means algorithm is equal to KN. The step 2 is the assignment step. It assigns to each point to a cluster by verifying the closest mean point. Consider the vector where each index indicates a patent, which we call x. For each mean point m, a cluster C is defined as follows:

$$C_i = \{x_p : ||x_p - m_i||^2 <= x_p - m_j||^2 \forall j, 1 <= j <= k\}$$

The step 3 is the update step, where for each cluster cluster C, the new centroids c have to be calculated. Considering the feature vector x described earlier, here is how it is computed:

$$m_i = \frac{1}{|S_i|} \sum_{x_j \in S_i} x_j$$

The algorithm returns an array containing arrays with indexes of the feature vector within it, pointing to patents saved within the folder. This implies that the N dimensions space is cut in K clusters, with patents related to each other as the distance calculated from their keywords indicate similarity. It is interesting

```
[ [ [ 0, 1, 2, 3, 6, 11, 14, 17, 18, 19, 21, 22, 24, 25 ] ],
  [ [ 8, 16 ] ],
  [ [ 10 ],
    [ 13 ],
    [ 26 ],
    [ 20 ],
    [ 4 ],
    [ 5 ],
    [ 7 ],
    [ 23 ],
    [ 12 ],
    [ 27 ],
    [ 9 ],
    [ 15 ] ],
  [ [], [], [] ] ]
```

Figure 51: The structure returned by the K-means algorithm. Each array in between the symbols [ ] contains intern indexes of patents within the folder. The number of patents within a cluster will be used to calculate the position of the cluster; the bigger the cluster, the closer to the center of the visualisation.

to notice we can easily recover the size of the clusters, which be valuable for our spatial organisation.

K-means provides clusters of patents belonging together within a N dimensional space. As our prototype is all in 2 dimensions, we decided to set up the visualisation of the clusters in 2 dimensions. The results of K-means do not provide us with information on spatial information, like the area taken by a cluster for example. Thus, it is out of the metadata of the K-means result that we decided to set up the spatial visualisation.
Our goal is for the user to immediately see the important clusters, which are the most likely to help him understand the subject of a high number of patents at once. We decided to set in the middle of the visualization the cluster with the highest number of patents contained within it.
This choice being made, we still had to decide how to organise spatially the other groups of patents. This was an important reflexion as the space of the widget clusters is relatively small, and all of its elements will grow bigger if the user zooms on the widget visualisation. Thus, tactile interactions could not solve the issue of space, and our focus had to be on providing a way to display all the clusters at once, while displaying their patents, and to avoid unclear separation by the user like (Marijn 12). The solution to unclear clusters we chose is rather easy: we draw a circle around the patents clustered together. This solution does have one important consequence, it requires complete spatial separation of all the clusters, as an overlay would make it impossible for the user to distinguish which clusters a patent belongs to, or even worse the crossing circles could mis-

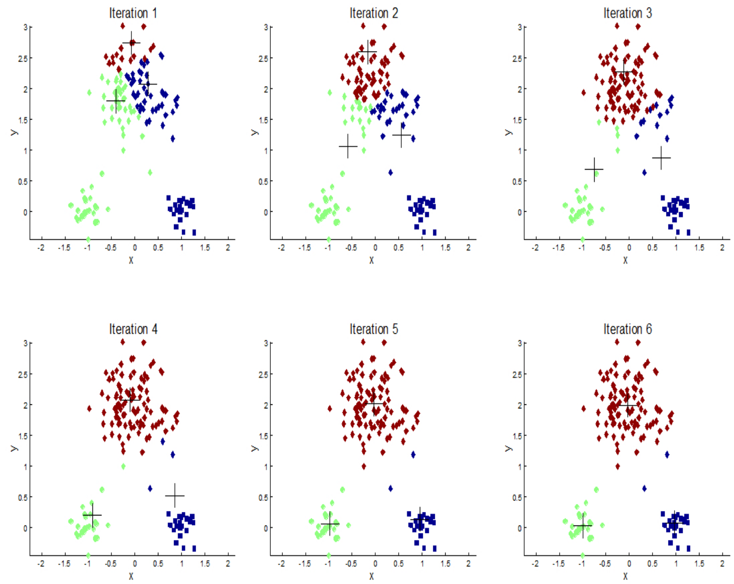


Figure 52: The evolution of the K-means through time. The mean points are modified by setting their position at the centroid of the cluster. The belonging of points within is then recalculated according to the closest mean point around them. This process is continued until the distance in between a mean point and the calculated centroid is smaller than a predefined threshold.

lead the user to believe patents within the crossing are part of both clusters. Thus, we decided to accept the possibility of widgets patents overlaying each other within a cluster. Patents are thus randomly positioned within the circle , and while the biggest cluster has a fixed size in the center of the visualisation, the other clusters are set in an organised way to convey metadata about them. The size of the other clusters depend of the number of patents within them. Their distance from the center is defined according to their size, as the bigger clusters are closer to the center. This hierarchy is easily understood by the user as he can see the largest cluster in the center and all the alone patents organised in a wheel far away. Thus, we define some layers around the center, depending of the size of the clusters. Those layers can be understood as invisible circles around the center indicating level of shared relationships. In order to avoid overlay, the angle from clusters of the same size and the center are evenly shared. That is for example if there are three clusters of the same size, the angle separating each of them from the center will be 60 degrees, with a unique starting angle for this layer. While this is not necessary for the largest clusters, as layers are set up far away enough from each other, there can be in some very big folders cases where there is no longer room to separate all the clusters by distance only (both patents and clusters can not go out of the widget clusters), and thus there are still relatively small chances of overlaying clusters.

Another point we want the reader to notice is that while users can move clones for a better visualisation of the keywords-based links, we did not allow the modification of the position of clusters, position, conveying the size of the cluster, allows to make the user does not miss a patent hidden by other widgets within the cluster.

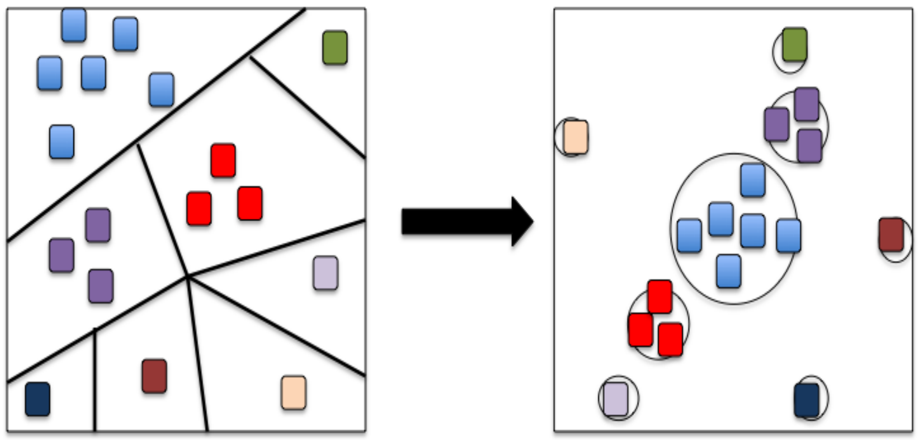


Figure 53: The change from the K-means space on the left (here in two dimensions for simplification) to our visualisation on the right: we select the elements according to the number of elements in the clusters, and set the distance to the center according to it. The angles in between the clusters are set to spread them evenly at an equal distance around the center of the visualisation.

6.4 Conclusions

In this section, we explained the motivation for the clustering of patents within a folder, and the display of keywords-based relationships. These chocies are reflected within the widget visualisation when interacting with a folder.
The widget visualisation when interacting with a folder allows the user to see the relations in between the patents it is composed of. As stated earlier, displaying all the data and metadata at once is a complicated, and even in some cases impossible task. In this situation, we decided to display keywords-based relationships in between the patents, with two tools, K-means based clustering, and when selecting a patent, lines displaying according to their color the links to other patents. We only display the relations from the touched patent and the related ones, and in order for the lines to be readable when there is a high number of relations, the user can move the patent to establish the ending point of all the lines. When pressing back the folder, the patent returns inside their circles to avoid the loss of the information given by the clustering.
Separating the N dimensions space where the distance in between documents is calculated and its visual representation allows for more effective representations

of a group of documents while displaying their metadata.

7 Conclusion and future work

This section describes our conclusion towards the work of this master thesis. It also describes the small but valuable user feedback we received, and from its lessons and our analysis of previously described work, possible directions and advises for future work.

7.1 Conclusion

Document triage is a complex process in which the user has to evaluate the value of a collection of documents according to his search. The European Patent Office examiners are daily faced with this task and do not dispose of efficient tools for this step. This is the issue that motivated the work of this thesis. Our research goal is the following:
Research goal : Create a prototype for the process of document triage. This prototype has to allow the user to deal with an amount of up to 1500 patents, allow the user to navigate through patents by using a large multi-touch input device, order the patents into a collection of selected patents, modify the collections of patents, and visualise data and metadata from this collection using a keyword-based clustering. This research goal gives clear directions on the objectives but close to none regarding the design of the prototype. Thus, we studied the state of the art concerning document collection visualisation, patent semantic, and multi-touch interaction. The literature review indicated to us the pros and cons of several directions for our work. We used these papers as inspiration, and provided a solution designed for a large multi-touch device, which brought advantages regarding the number of elements the user can see at once, and allows the user to interact in a natural way. Still, the hardware is not comfortable for the user to type upon, thus our design had to be focused towards a solution where the user does not have to keep his arm stretched for a long time. The prototype we created was developed using Javascript and the Multitouch Cornerstone SDK, upon a CTOUCH Laser 70 inches touch screen.
Our starting point was heavily influenced by the work of (Ridder 11), for its several layers of data showed according to the situation and especially for the possibility to divide a collection of patents into a smaller one in order to analyze them more efficiently. (Dunne 12) also heavily influenced our work for the clear separation of elements not belonging to the same cluster when looking at patents saved within a folder.
Another point of focus we want to underline here is our desire to provide a solution with little text, focusing on efficient data filters so that data is displayed when the context justifies it. Demonstrations of use of the European Patent Office official tools were very greedy in attention needed to operate them. The issues with the European Patent Office tools influenced heavily as which problems should be avoided: Large amount of text displayed at once, large number of clicks needed to operate, and separations of tools within several applications for the same process. This justifies the usage of an examiner made database

with notes done after examination, adding high value to a quick understanding of a patent, as abstracts are written in a very formal and inconvenient way.

We believe the prototype we developed offers valuable solutions to the problems encountered during document triage. The benefits of our solution are the following :

- Quickly establishing major metadata of a collection of documents like classification, date of publication, language, or the importance of the author.
- Limiting the amount of text to its minimum to keep the user focus.
- Understanding of the subject or field of a document thanks to its keywords and its relations to other patents sharing the same keywords.
- Natural tactile interactions to feel closer to the data.

Thus our prototype brings new ways to visualise metadata of a collection of documents, and keywords-based relationships in between patents within a folder.

7.2 User feedback

It was particularly difficult for us to find examiners to test our prototype, due to the very important amount of patents they have to deal, and because we finished our prototype during the month of July, when many of the contacts we made within the European Patent Office were not available. Thus, we only managed to have one examiner to try our prototype. While an user study would be pointless with such a small number of participants, his feedback after trying the prototype was still very valuable. Thus, we will explain what resulted of the conversation with the examiner when asked his opinion about elements such as the interactions, the scene visualisation, the semantic of patents, the visualisation of a folder, the screen, and more.
The feedback from the examiner was overall positive, especially thanks to the pleasure of using the tactile interactions with colourful visualisations. He said it reminded him of a science fiction movie because of the large amount of data displayed at once on a large tactile device. The author view was appreciated by the examiner, as he likes to sort the patents according to authors together to examine them. The language view was declared a good idea because, within his field of work, he often encounters patents from China, which he needs to separate and have translated, and so he prefers to first disregard them. The menu to select which authors to filter was also appreciated as he said to usually know some authors and thus already know their fields of work. We were also pleased to hear that the examiner had no issue understanding the subject of a patent when reading the description, and even sometimes understand the functioning of the patent by reading the images and the description together. The keywords links within the widget clusters were also appreciated by the user.

Still, some elements were missing or not understood without further explanations. While the text displayed when selecting a patent is clear enough for the user to understand the meaning of a patent, the lack of claims was an issue for him as claims define legal boundaries, and help examiners also disregard patents with similar subjects but claims with no possible opposition to the ones done in the patent application. The impossibility to name the folders was also disappointing for him, as examiners are used to this functionality and got used to name smartly in order to find them back immediately and know the reason that pushed them to organise a selection of patents together without needing to look back at them. Finally, the viewer has a system that allows to point from annotations on images into their descriptions, and can be useful for subjects where the images are especially important.

7.3 Future work

The prototype we realized during our 6 months internship, contains a high potential for the document triage process for the examiners at the European Patent Office. Nevertheless, there are functionalities or content we did not consider necessary or did not have time to add to our prototype. This section will describe the ideas we consider best for future work.
Firstly it is important to underline what user feedback taught us: the claims are necessary for the examiners to determine the legal boundaries of a patent. Adding this data that is not within our database means linking it with the EPOQUE database, which we expect to be a tricky problem to solve with the proxy limitations within the European Patent Office.
Another element examiners use a lot and is very difficult to replace is the keyboard. While our solution allows the user to select the keywords from a patent to select a whole group, directly typing a keyword to highlight it within the visualisation could be a quick way to interact too, especially at the beginning of the analysis of a search query result analysis. The position of the keyboard can seem problematic, as the user needs is very likely to naturally move close then further from the screen to adapt to the number of elements he wants to look. One solution could be to have a keyboard on a sliding table, thus the keyboard could be pulled close to the user when necessary, and pushed under the screen otherwise.
This prototype is based on the personal database of an examiner, and thus provides user generated content with high semantic value. We believe following an approach like (Giereth 08) where can save annotations and descriptions on the patents online can drastically enhance the semantic quality of the patents. This idea does raise a lot of issues regarding the intern structure of the European Patent Office, which we do not know well enough to estimate its adaptability to implement this solution.
We also believe the visualisations can be enhanced. For example, (Bach 14) and his 3 dimensions row of adjacency matrices showing the evolution of citations through time could be implemented to quickly find authors often cited and their evolution through time. The time evolution is interesting to us because it can

show that an author might have a lot of patents resulting the search request, but if they are all very old, they are unlikely to be relevant.

Our first prototype was designed to be a 3D environment, with cubes within this space each representing a patent returned by the search query, and each face of the cube would display a visualisation of the metadata of the patent. Interactions were planned to be done using a special cube where each face would display metadata for each patent. We believe such a concept is particularly beautiful, it brought many issues regarding the view, the position of the camera, and others. Also, reading again (Cockburn 01) reminded us that 3D, while preferred, was less efficient. While we believe there is potential in such an idea, too many questions needed an answer before focusing on the process of document triage, thus we decided to set back to a 2D visualisation. (North 09) showed us that the closer to reality the environment was, the more efficient the sorting made by the user. Thus we believe a solution in 3D still has a lot of potential, but is likely to be achieved only if intermediary questions are solved first.

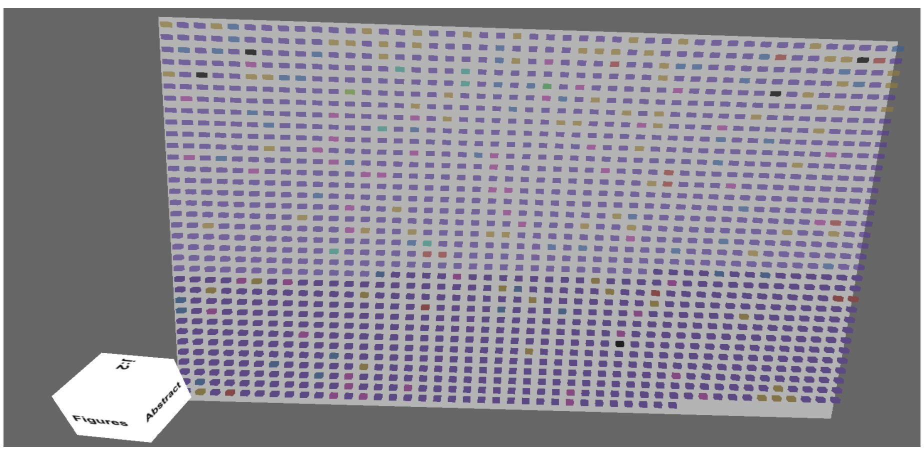


Figure 54: Our prototype when our focus was on the 3D. Each cube represents a patent returned by the search query. Clicking on the faces of the white cube would modify the data displayed for all the cubes on the desktop. We dropped this idea mainly for the reason of the complexity it brought for very little to no real contribution.

Finally, an important element we had to disregard during the development of our prototype is the possibility to assess, with a collection of patents, their legal status. The rights of patents can be expired, bought by another person, but they also can be extended. While it is not necessary to understand the meaning of an invention, examiners need to disregard patents which no longer are under a legal protection and fall in public domain. Currently, this step is executed at the same time as the document triage process. Thus the future version of this prototype should contain this kind of information in order for the examiners to be able to complete each step of a patent application analysis with it.

References

[Agarawala 06] Anand Agarawala & Ravin Balakrishnan. *Keepin it real: pushing the desktop metaphor with physics, piles and the pen.* In Proceedings of the SIGCHI conference on Human Factors in computing systems, CHI 06, pages 12831292. ACM, 2006.

[Bach 14] Bach, B., Henry-Riche, N., Dwyer, T., Madhyastha, T., Fekete, J. D., & Grabowski, T. (2015, May). *Small MultiPiles: Piling Time to Explore Temporal Patterns in Dynamic Networks.* In Computer Graphics Forum (Vol. 34, No. 3, pp. 31-40).

[Bederson 10] Benjamin B. Bederson. *The Promise of Zoomable User Interfaces.* In Proceedings of the 3rd International Symposium on Visual Information Communication, 2010.

[Bergstrom 09] Peter Bergstrom. *Augmenting Digital Libraries Using Web-Based Visualizations Augmenting Digital Libraries Using Web-Based Visualizations.* Master of science in computer engineering thesis, Santa Clara University, 2009.

[Chuang 12] Chuang, J., Ramage, D., Manning, C., & Heer, J. (2012, May). *Interpretation and trust: Designing model-driven visualizations for text analysis.* InProceedings of the SIGCHI Conference on Human Factors in Computing Systems (pp. 443-452). ACM. Department of Computer Science, Stanford University Department of Computer Science and Technology, Tsinghua University

[Cockburn 01] Cockburn, A., & McKenzie, B. (2001, March). *3D or not 3D?: evaluating the effect of the third dimension in a document management system.* In Proceedings of the SIGCHI conference on Human factors in computing systems (pp. 434-441). ACM.

[Dunne 12] Cody Dunne, Ben Shneiderman, Robert Gove, Judith Klavans & Bonnie Dorr. *Rapid understanding of scientific paper collections: integrating statistics, text analysis, and visualization.* University of Maryland, Human-Computer Interaction Lab Tech Report, 2012.

[Fanea 05] Fanea, E., Carpendale, S., & Isenberg, T. (2005, October). *An interactive 3d integration of parallel coordinates and star glyphs.* In Information Visualization, 2005. INFOVIS 2005. IEEE Symposium on (pp. 149-156). IEEE.

[Giereth 08] Giereth, M., & Ertl, T. (2008, July). *Visualization Enhanced Semantic Wikis for Patent Information.* In Information Visualisation, 2008. IV'08. 12th International Conference (pp. 185-190). IEEE.

[Joia 11] Paulo Joia,FernandoV.Paulovich,DaniloCoimbra,Jose A.Cuminat & Luis G. Nonato. *Local Affine Multidimensional Projection.* IEEE Transactions on Visualization and Computer Graphics, vol. 17, no. 12, pages 25632571, 2011.

[Kairam 12] Kairam, S., MacLean, D., Savva, M., & Heer, J. (2012, May). *GraphPrism: compact visualization of network structure.* In Proceedings of the International Working Conference on Advanced Visual Interfaces (pp. 498-505). ACM.

[Kandel 12] Kandel, S., Parikh, R., Paepcke, A., Hellerstein, J. M., & Heer, J. (2012, May). Profiler: Integrated statistical analysis and visualization for data quality assessment. In Proceedings of the International Working Conference on Advanced Visual Interfaces (pp. 547-554). ACM.

[Koch 10] Steffen Koch, Harald Bosch, Mark Giereth & Thomas Ertl. *Iterative In- tegration of Visual Insights during Scalable Patent Search and Analysis.* IEEE Transactions on Visualization and Computer Graphics, vol. 17, no. 5, pages 557569, 2010.

[Liu 13] Liu, Z., Jiang, B., & Heer, J. (2013, June). *imMens: Realtime Visual Querying of Big Data.* In Computer Graphics Forum (Vol. 32, No. 3pt4, pp. 421-430). Blackwell Publishing Ltd.

[Marijn 12] Anne Marijn Van Ee, *Touch-Based Organization of Patent Collections.* Master of computer science thesis, Delft University of Technology, 2012

[North 09] Chris North, Tim Dwyer, Bongshin Lee, Danyel Fisher, Petra Isenberg, George Robertson & Kori Inkpen. *Understanding Multi-touch Manipula- tion for Surface Computing.* In Proceedings of the 12th IFIP TC 13 Inter- national Conference on Human-Computer Interaction: Part II, INTERACT 09, pages 236249, 2009.

[Paulovich 08] Fernando V. Paulovich & Rosane Minghim. *HiPP: a novel hierarchical point placement strategy and its application to the exploration of document collections.* IEEE Transactions on Visualization and Computer Graphics, vol. 14, no. 6, pages 12291236, 2008.

[Ridder 11] Michel de Ridder. *TouchPat: A Touch Based Zoomable User Interface for Exploring Large Patent Collections.* Master of computer science thesis, Delft University of Technology, 2011.

[Schmidt 10] Sebastian Schmidt, Miguel A. Nacenta, Raimund Dachselt & Sheelagh Carpendale. *A set of multi-touch graph interaction techniques.* In ACM International Conference on Interactive Tabletops and Surfaces, ITS 10, pages 113116, 2010.

[Van der Maaten 08] Van der Maaten, L., & Hinton, G. (2008). *Visualizing data using t-SNE.* Journal of Machine Learning Research, 9(2579-2605), 85.

List of Figures

Appendix

The prototype was developed using a System Development Kit (SDK) in order to have already implemented tactile functions. Here we will describe the SDK we chose, why we made this choice, and the consequences this choice implied on the code structure and thus on the prototype.

The environment

We developed this prototype coding in Javascript using the Multitouch Cornerstone SDK [1] on a CTouch 70 inches multi-touch screen. The SDK, first developed for a C++ application, easily allows the creation and definition of widgets that the user can translate, rotate and enlarge using predefined tactile interactions. We decided to work with Javascript as it is a quick coding language to prototype, and thus it allowed us to test ideas rapidly. Coding our prototype in Javascript also allowed us to use npm, a package manager that easily integrates functions, like the K-means algorithm we use within our program. Nevertheless, it is important to notice some issues with the Multitouch Cornerstone SDK, as we faced some bugs that made little sense, like the single tap predefined functions that required double tap to be recognized. While it may be minor for this case, this had consequences on our freedom of interactions. According to the Cornerstone customer service this is probably due to the fact we do not own a license, even though our functions respect the boundaries defined within the free version. According to customer service, the issue with the single tap functions requiring a double tap, is due to the screen being recognised by the SDK as a third party hardware, and should be solved with the purchase of a license. It is also important to highlight that the lack of license limited our interactions to three fingers upon the screen at once maximum. Thus this limited our possibilities for designing specific interactions with several fingers at once. While we are satisfied with the interactions of our prototype because of their simplicity and the users' habits with them thanks to smart phones, we believe adding new, more rich functionalities with more fingers at once presents a very high potential.
Still, we decided to use the Cornerstone SDK instead of the Microsoft Surface SDK used by (Ridder 11) because of an important limitation with the Microsoft Surface SDK: it only allows programs working under a Windows OS, which we believe is a drawback compared to a Linux OS regarding development possibilities, as the user can enjoy a richer and freer environment to develop future work.

Specificities of the SDK

Working with the Multitouch Cornerstone SDK gave us tools to implement tactile interactions within our program. Still, some specifities of the SDK forced

[1] https://cornerstone.multitouch.fi/

us to change the structure of our program in order to improve the performances. Indeed, we were quite surprised to notice the number of frames per second (fps) drop heavily when calling what we considered a relatively small amount of computationally light functions. While we are not sure if this is due to the SDK or Javascript not taking advantage of the whole capacities of the machine, this forced us to drastically limit our use of predefined functions called on rendering of widgets.
Another important point with the prototype is that the program does not handle very efficiently the creation or deletion of widgets in the scene, and also is very slow when there is a need for a loop within the children of a widget (the widgets contained within another one are defined as children). Thus, the program had to be a mix of CSS for the fixed design, such as the highlight around a selected object, and some functions defined within the SDK, such the color of the widgets when it needed to be dynamically changed according to the data because while the SDK allows to use some predefined CSS classes for the widgets, including colors, it would be too long to generate all the type when we are dealing with colour gradually changing according to values like date of publication.
Finally, it is interesting to notice that in order to link our program to the objects created in the scene, we had to create a whole system of hashmaps taking the widget's identifier pointing towards the object we created, but also a system of hashmaps to keep the user actions saved, in order to dynamically adapt the selection, and thus adapt the CSS of objects.