computing@computingonline.net
www.computingonline.net

ISSN 1727-6209
International Journal of Computing

VISUAL SEARCH FOR CATALOGS, COLLECTIONS AND ARCHIVES

Daniel Burckhardt [1], Pepe Jürgens [2]

[1] Marienburger Str. 32, 10405 Berlin, burckhardtd@geschichte.hu-berlin.de
[2] Winsstraße 6, 10435 Berlin, juergens@weltformat.de, http://www.weltformat.de

Abstract: *Searching is a necessity. Search should be fun. Catalogs, collections and archives contain large amounts of data ready to search. But the most common presentation of the results in the form of long lists of text seems to be rather repulsive. Various studies have shown that only very few of these results attract the users' attention. Improving the order of search results is therefore at the center of numerous research projects. Visual search is an alternative approach based on an observation from the pre-digital era that avoids these technical complexities: extensive image libraries – the slides on a light table – allow for easy scanning and ordering of large result sets. Visual search aims to bring this approach to the web. The authors first followed this concept while implementing oSkope, a visual search engine through web services such as Amazon, eBay or YouTube. The key design patterns were suitable for reuse in a number of Rich Internet Applications for collections and product catalogs since realized. By combining an intuitive and fluid navigation with a clear, meaningful and aesthetically pleasing presentation, these applications attract the users' attention to a larger set of results. Additional information can be displayed on demand. Sophisticated search options allow for an accurate search. The results can be saved for later use or be sent to a friend.*

Keywords: *Website search, Rich Internet Application, User experience.*

1. INTRODUCTION

Searching is a necessity. In 1997, Jakob Nielsen observed that almost half of the visitors of a website started their visit with a search. The users' expectations for such an interaction are well known [1]:

1. A box for entering the search terms;
2. a button labeled "search" that they can click;
3. whereupon the search engine results page (SERP) appears as a list with the best matching entries at the top.

Though these steps are well known, the outcome of this simple process is often frustrating. The results of a site-search may be worse than a similar request entered on Bing or Google. Thanks to open source search software such as Solr or Sphinx (http://lucene.apache.org/solr/, http://www.sphinxsearch.com), even sites with smaller budgets could achieve better results after proper analysis of its content and a corresponding configuration of the indexing process.

Nevertheless, keyword search remains a difficult problem, since "searching for words isn't really what you want to do. You'd like to search for ideas, for concepts, for solutions, for answers" [2]. Only skilled users are able to re-formulate these expectations into a short keyword query. Visual search can't solve this problem. But it tries to provide an alternative: Instead of focusing on an optimal result set, we spend a considerable effort to provide a better *presentation* of the results. By improving the user experience, we hope that visitors are more willing to spend their time looking through a large amount of search results in spite of imperfect ranking and a few missing matches.

The visual presentation of the SERP is not a magic bullet to overcome all the difficulties of search. It requires a suitable data set where every entry can be faithfully represented by a visual placeholder. Besides image and media archives, which are the main candidates for this approach, there are numerous databases and online shops where the choice of a product depends as much on its appearance as on further requirements such as the price or more technical constraints.

We also have to keep in mind that the skills of users visiting a site are as heterogeneous as the underlying data sets. Power users and information professionals are well trained to rapidly scan large amounts of text. In contrast, designers and more novice users might feel much more comfortable browsing through a visual set of results. Last but not least, it is important to remember that search is often not a clearly defined problem with an optimal solution. Certain requests – what is the phone number of a particular person or what are the other books by a specific author – allow a quantitative assessment of the results according to accuracy (precision) and completeness (recall). By contrast, looking for ideas for a newly designed garden or suitable pictures for a magazine report will lead to results whose accuracy and completeness seem almost impossible to quantify [3]. When starting such an explorative search, it is not clear which term should be typed into the text box. The results on or off the road to the destination can be as interesting as what was initially on the mind of the person carrying out the search. Others might be able to recommend a few matching answers. Soft constraints such as the price may be used to quickly discard some results. But in the end, only the seeker himself can decide whether a certain result is good enough to bring this search to an end.

Visual search is particularly suited for the exploratory way of searching just described. The intuitive and highly interactive interface facilitates the screening of alternative queries. A visual search consists of multiple steps similar to the opening (first search), the middlegame (orientation and refinement) and endgame (the screening of suitable results) in a chess match [4]. Ultimately, a well designed interface for such an interactive process should enable a seamless dialogue between the user and the application.

In the following chapters, we will show, how

- a creative interface well adjusted to the structure of the underlying data set,
- modern programming for highly interactive and fluid handling,
- combined with options for structured search and matching result views

can lead to a search application that is able to initiate such a dialogue. The authors first implemented this concept for oSkope (http://www.oskope.com), a visual search engine through web services such as Amazon, eBay or YouTube. As soon as this site was online, we found out that there was a large demand from museum collections and digital archives for similar interfaces. We will therefore first describe the recurring elements of our visual search applications. After that, we present the specifics of our technical implementation. Through a series of case studies, we show how to enhance the generic elements of visual search in accordance with the underlying data and the audience of a site. We will also present a case where visual search was combined with RFID tags in order to close the gap between the haptic search in the world of things and a search application running on the screen of a computing device.

2. BASIC ELEMENTS OF A VISUAL SEARCH

Search has been one of the key activities since the beginning of the WWW. Five of the ten most popular sites are focused on web search. When Google started its rise to become the dominating player in the field, two key features clearly distinguished the site from predecessors such as Altavista or Hotbot: a radically streamlined home page with the search box prominently centered in the middle of an almost empty page and a significantly improved sort order based on the so called PageRank. This algorithm counts every link on a web page as a more or less heavily weighted recommendation for the target page [5]. The rapid rise of Google has narrowed the discussion of search almost exclusively to the ranking of the results. As Morville and Callender show in their book “Search Patterns” [6], this “Best First” pattern is only one of many features that should be integrated into a successful search. In addition, they describe autocomplete, federated search, faceted navigation, advanced search, personalization, pagination, structured results, actionable results and unified discovery. Visual search is primarily an alternative to the presentation of the SERP as a linear list of text. As will be shown in the case studies below, the combination of a visual presentation of the results with additional patterns such as faceted navigation or an advanced search is often desirable.

The leading metaphor for the visual search is the lightbox for slides. The user can sort, group and freely arrange the images on this table. He is active, experimenting, looking for feedback or joint work with colleagues. This is a fast and creative process: “I see what I can find and how I can find it.” The narrow frame of a slide couldn’t accommodate more information than a short note containing the title, place and date of the slide. After transferring the diapositive into a multimedia application, we are no longer bound to these restrictions. Additional information can be displayed in a reduced form by moving the mouse over the preview image. Additional information will be loaded and displayed whenever the user clicks on the slide.

A first prototype operating on Amazon’s product database was designed by Christoph Schifferli and programmed by Peter Chylewski. It was the authors’ task to turn this demo of a visual search into a fully featured web application. The goal for oSkope was to combine the quick, fluid and simple interaction with an attractive design and smooth animations in order to attract and support the visitors. While reworking the private proof of concept into the public site, additional services such as eBay and YouTube were added as well.

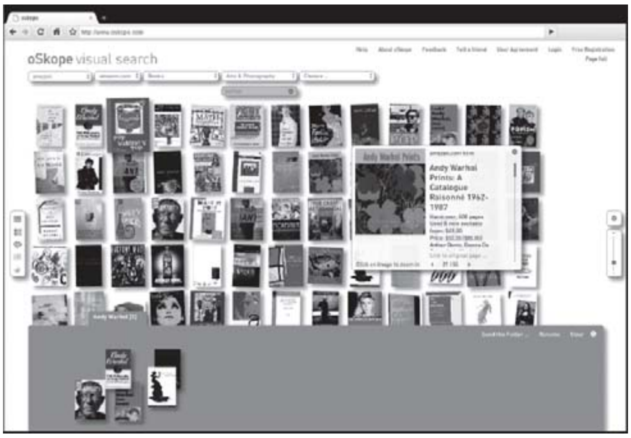


Fig. 1 – Basic elements of visual search at oSkope visual search

Single-Page User Interface

The visual search breaks with the traditional page reload paradigm of conventional web applications. Similar to AJAX-calls, only the data that is actually needed will be loaded in the background without completely rebuilding the screen. Fast response times alleviate the users' fear of losing existing results and thereby encourage them to explore additional actions. This creates a "flow" where users tend to forget the time and are drawn into much more extensive searches. By looking at the access logs from oSkope, we know that this actually works: An average user remains on the site for four minutes executing more than twelve click or search actions (search requests, opening detail information or continuing to the next result page). These values are almost twice as high (duration of the visit) or five times higher (number of clicks) than comparable sites.

Result Views

Conventional search result pages usually accommodate between five (Amazon / eBay) and ten hits (Google / Bing) in the visible area of the browser window. The users must then scroll down or click to advance to the next page. Since preview images in the visual search can be displayed not only vertically but also horizontally side by side, a lot more results fit on a single screen. Depending on the image and screen size, 50 to 150 records can be viewed simultaneously. This gives the user an overall impression. Optimally, a matching result which would have been missed with the traditional representation will be visible on the first screen of results already.

Different arrangements and groupings of thumbnail images in the result view reflect different approaches to finding and individual preferences. In addition, certain aspects such as the size of the preview pictures or the choice of the axes are customizable by the user. We will present a number of views that we have implemented in our existing applications. Further views, such as an arrangement of thumbnails on a geographical map or by adding a third dimension, should be explored in the future.

Grid View: The grid is our standard view. This representation is suitable for most data sets. In addition, visitors who are confronted with a visual search for the first time will quickly understand this arrangement. In oSkope, the strict grid is broken up by a slight random rotation of the images. By adding a small perturbation, the view appears a bit more playful compared to the fully regular grid.

Stack: The images are spread out in overlapping rows and thereby invite the user to actively browse. As soon as the mouse moves over an image, it will be enlarged and moved to the foreground. With this view, the user gets an overview of a very large set of pictures in a small space.

Pile: The images are randomly piled all over the screen similar to a deck of cards thrown on the floor. Similar to the stack, pictures are turned exactly into the grid and placed up front as soon as the mouse is hovered. The pile may lead to unexpected discoveries, can be aesthetically pleasing or just very amusing to look at: "Sometimes, it's just about fun. After an exhausting debate about order options, contemplating the chaotic piles of oSkope's visual search application can deliver valuable comic relief." [5, p. 141]

Lists and tables: In this view, the results are listed in a strictly linear list or as a two-dimensional table. When using the table, columns carry additional information such as name, price or weight. The user can sort the table according to these criteria. This view is very close to traditional result pages. This makes it easy for new users to switch to the visual search.

Groups: The pictures are arranged in groups according to a predefined classification of the data into a set of categories. This way, similar results can be spotted at a glance.

Graph: New insights are possible if the images are arranged two-dimensionally. On eBay, using duration and price as the two axes, the lower right quadrant becomes the "bargain corner" showing all the deals about to expire with a low maximum bid.

Ray: Images are arranged on a horizontal line presorted according to a certain value. On the timeline for example, older to newer images are placed from left to right. Horizontal navigation is possible either via a scroll bar or according to the Cover Flow principle familiar from Apple's iTunes program.

Preview and details

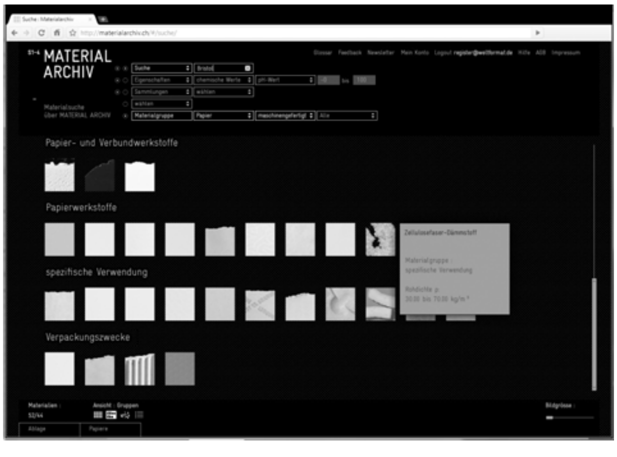
After a rough visual inspection of the results, additional information may be relevant to continue the search: Who is the author of the book with the attractive cover? Is the most favorable offer a hardcover copy or paperback book, is it used or new? This process is split into two or three stages. Depending on the application, first details such as the name of a material are shown from the very beginning in or below the preview image. As soon as the mouse cursor moves over a result, additional information can be displayed in a small window.

Further information is displayed in a popup window if a user remains with the mouse pointer over an image or clicks on it. This detail view is not limited to text. Further images, tabular data, audio or even a video can be played directly without switching to a separate application.

The search results will always remain present in the background. Therefore, the result page doesn't need to be rebuilt when the detail view is closed. Not having to wait, the user is encouraged to explore much more results in detail.

Folders

Always visible at the bottom of the screen are one or more folders acting as a watch- or wish list. Any search result can be dragged to one of the folders. Entries may just be temporarily moved to a folder or ordered into personal collections and permanently stored. Depending on the application, multiple folders can be created at the same time and then renamed and stored for future sessions. The content of a folder can be sent by e-mail as a personal reminder or as a recommendation to a friend.



Suche: Bristol
Eigenschaften: chemische Werte: pH-Wert: 0 bis 100
Sammlungen: wählen
wählen
Materialgruppe: Papier: maschinengefertigt: Alle

Zellulosefaser-Dämmstoff
Materialgruppe: spezifische Verwendung
Rohdichte ρ: 30.00 bis 70.00 kg/m³

Materialien: 52/64
Ansicht: Gruppen

Suche : Materialarchiv
http://materialarchiv.ch/#/suche/
MATERIAL ARCHIV
Glossar Feedback Newsletter Mein Konto Logout register@weltformat.de Hilfe AGB Impressum
Materialsuche über MATERIAL ARCHIV
Papier- und Verbundwerkstoffe
Papierwerkstoffe
spezifische Verwendung
Verpackungszwecke
Bildgrösse :
Ablage
Papiere

Fig. 2 – Group view combined with an advanced search (MATERIAL.ARCHIV)

Combination with other Search Patterns

The basic elements of a visual search just mentioned can be freely combined with other search patterns. Thus, for example, the entry in the search field can be expanded in the background acting as an auto-completer. Searches can be combined with a category tree to quickly reduce or enlarge the set of results. Advanced search options allow for searching specific fields or limiting parameters to a predefined set or range of values. The decision for these additional functions depends both on the underlying set of data and the audience targeted by a specific application. We will therefore explore these additional options in more detail in the context of specific case studies in the last section of this article.

3. TECHNICAL IMPLEMENTATION

Front end

The front end is responsible for the presentation of the search results. The Rich Internet Application (RIA) has been implemented in Flash. This allows for smooth animations, drag-and-drop operations as well as the integration of many media formats. In addition, most calculations needed for the rotation and scaling of images can be run directly on the client without additional server calls or resources. This minimizes the number of server requests and ensures short response times no matter how many simultaneous visitors are using the site. To facilitate the maintenance, expandability and flexibility of the application as well as to simplify its reuse, the front end was consistently structured in according to the Model-View-Controller pattern (MVC). Further result views can be added to the front end without changing the main code. Certain features such as search fields and personalization options can easily be hidden if the front end is running on terminals without keyboard interaction in an exhibition space. The separation of the view enables customer specific designs without changing the code of the model and controller.

Digression: Why Flash?

When we started our first visual search application in the autumn of 2006, there were a number of prerequisites for the application:

• The application should run in a number of web browsers current at that time (Internet Explorer 6, Firefox 1.5, Safari 2) without installing additional plugins.

• The interface should be elegant, fast and interactive (fonts, shadows).

• Meaningful animations should support the user in finding and facilitating the operation (animation and scaling of images, folders and collapse).

• The common structure of a web site as a collection of loosely linked pages should be abandoned. Instead, the user should remain on a single page during search.

Part of this requirement could have been met by combining HTML and JavaScript into an AJAX application. Smooth animations, particularly the rotation of images, font effects and shadows could not be implemented in a cross-browser compatible way without resorting to Flash. Since over 95% of the browsers currently online support Flash, this decision proved its worth. Over the past two years, Adobe has adapted the ActionScript programming language to the ECMAScript standard and significantly increased the execution speed of the virtual machine. However, the use of proprietary standards such as Flash for implementing web applications is quite controversial. With the widespread use of the iPhone and the successful introduction of the iPad, alternative implementations of the front end need to be considered seriously as long as Apple’s lack of Flash support for these two devices persists.

Depending on the client, additional requirements concerning the accessibility for people with disabilities may have to be met. Some of them may be difficult to implement in Flash. Since the two main alternatives, the canvas element from HTML5 and the W3C standard for Scalable Vector Graphics, are both not implemented in the current production version of the Internet Explorer, no valid alternative to Flash is supported across all current browsers with a wider market-share.

Middleware: the client’s connection to the database

The Flash client talks to the middleware on the web server both for searches and for the display of detailed information. This service encapsulates the access to a relational database or a third-party web service. For performance reasons, pictures are loaded by the front end through http without calling the middleware. The additional layer on the web server was initially a technical necessity: Under the so-called “same origin policy”, the security model of Flash, a Flash application that is embedded into a web page located under http://oskope.com may not directly communicate with services under a different domain such as http://rest.api.ebay.com. Looking back, this additional layer of abstraction also facilitated the maintenance and reuse of the front end. Since all the specifics of a web service are encapsulated in the middleware, the integration of additional services such as YouTube as well as API updates by eBay could be implemented without code changes in the Flash client. In addition, a well-specified interface between front end and middleware leads to a clear division between front end and server programming. This was the key for dividing the work among multiple programmers and adherence to test-driven development of both front end and middleware.

The protocol used to communicate between the Flash client and the web server is the so-called Action Message Format (AMF). Alternatives would have been JSON or XML. These two plaintext based encapsulations were significantly slower in Flash 8 than the binary format from Adobe. AMFPHP proved to be a comfortable and well performing open-source implementation and test environment for this protocol. Meanwhile, many developers and users of AMFPHP switched to Zend_Amf. Since the later is seamlessly integrated into the Zend-Framework, we also used the new library in our most recent project. Similar implementations for AMF are also available in Java, .NET, Perl, Python and Ruby. Therefore, the middleware can be seamlessly integrated into almost any existing system environment.

4. CASE STUDIES

In this chapter, we briefly discuss certain specific features of the visual searches we implemented so far. Among these applications, we find an alternate interface for third party services (oSkope), the online presentation of a legacy database (FRMA) as well as the front and back end for an archival database built up from the ground as RIA (MATERIAL.ARCHIV). We won’t repeat elements common to all of these applications that we discussed above, but focus on the characteristic features of these solutions instead.

oSkope: An alternative interface for public web APIs from Amazon, eBay and other providers

Several popular online services like Amazon, eBay or YouTube provide well documented public interfaces to their internal databases. Such an API is the technical and legal requirement for our own visual search on these data sets.

Navigation in oSkope is primarily tied to the selection of a service such as amazon.com or ebay.de. Category selections enable the drill-down of the results and can be combined with search keywords. The integration of web services is encapsulated in the middleware. Different image databases such as Fotolia, Yahoo Images and Flickr could be added to the front end without code changes.

After realizing oSkope, we discovered that most of the basic elements of a visual search could be applied to a local database as well. Since we were no longer bound to certain limitation of the web services, additional features such as facted navigation could be added.

FMRA collection: Online presentation of a legacy database with faceting

The Centre national de l’édition et de l’art imprimé (cneai) has an extensive collection of over three thousand artists' books and publications they wished to make freely available over the Internet. The database was previously managed internally as a Filemaker database. For cost reasons, this backend has been retained. We used the XML export format from Filemaker to periodically transfer the collection to the web.

For FMRA (http://www.collection-fmra.org/), we currently offer only the grid view. In addition, we feature a powerful navigation to quickly filter the results by certain categories (publication type, publication period, country of origin or artist). This faceted navigation continuously displays the number of entries matching all possible restrictions, a feature inspired by an earlier version of Apple's iTunes application. The dynamic update of the available categories after every selection warrants non-empty empty result sets for every possible action.

MATERIAL.ARCHIV: Materials Database with RFID-Integration

MATERIAL.ARCHIV (http://materialarchiv.ch/) is a cooperation of four public institutions that joined forces to come up with a collaboratively maintained database for the management and presentation of their existing collections of different materials. In addition to the scientific preservation and enhancement of the collections, the target audiences of the web presentation are designers as well as architects in training and in professional practice. The application can be used to carefully document the numerous properties of the different materials and to systematize the implicit knowledge of the curators.

There was no legacy data or application when this project was started. Thus a private backend for adding and editing the materials was jointly implemented with the public front end. The front end was planned to address the needs of an occasional user. In addition, complex searches for specific materials and certain properties were supported as well. To improve the presentation inside the archive and the exhibition spaces, the physical samples were tagged with an RFID chip connected to the contents of the database. Whenever one or more materials area placed in the scanning area next to the terminal, the entries immediately appear in the browser as results of a visual search. Direct access to the data sheets and the personalization options for storing and sending out materials are seamlessly integrated into the usage of the collection [7].

Baunetz Products: Interface Alternatives for an existing database

Heinze has a long tradition as a company providing extensive directories of products used by the building trade. These catalogs can be searched directly through the web site of Heinze. Planners, engineers and construction companies regularly use this service, while architects are clearly underrepresented.

This led to the idea to offer visitors of Baunetz architects portal – a sister company of Heinze – a more design-oriented access to the same product information. By reusing the existing data access layer, Baunetz products could be implemented in a brief timeframe of just a couple months (http://www.baunetzprodukte.de/). The site demonstrates that traditional search and visual search are not mutually exclusive. Both versions access a common search index. It is therefore less a technical than a strategic decision whether to present the two sites under a single or two different domains.

5. CONCLUSION

According to a study by Forrester Research mentioned in [3], three-quarters of the companies surveyed rated search as an extremely important feature for their web site. Less than a quarter of the respondents judged their own search to be extremely helpful. Visual search is just one of many opportunities for improving this situation.

Catalogs, archives and collections with an attractive image stock are particularly suitable for such a solution combining an attractive presentation with advanced possibilities for research and playful discovery. The focus should always remain on the simple and intuitive operation. This will lead to an increase in motivation that becomes measurable through longer visit durations and an increase in clicks. If the users thereby encounter results they wouldn't have encountered before, the visual search has hopefully fulfilled its purpose.

6. REFERENCES

[1] J. Nielsen. *Mental Models For Search Are Getting Firmer.* 2005. http://www.useit.com/alertbox/20050509.html

[2] T. Bray. *On Search*. 2003. http://www.tbray.org/ongoing/When/200x/2003/06/24/IntelligentSearch. Called up on 2010-03-20

[3] M. Hearst, A. Elliott, J. English, R. Sinha, K. Swearingen, K.P. Yee. Finding the flow in web site search. *Communications of the ACM archive*. 45 (9) (2002). pp. 42-49.

[4] K.P. Yee, K. Swearingen, K. Li, M. Hearst. Faceted Metadata for Image Search and Browsing. *Proceedings of the ACM Conference on Computer-Human Interaction* (2003), ACM Press, New York. pp. 401-408.

[5] S. Brin. L. Page. The anatomy of a large scale hypertextual web search engine. *Proc. 7th WWW* (1998). http://infolab.stanford.edu/~backrub/google.html.

[6] P. Morville, J. Callender. *Search Patterns: Design for Discovery*. O'Reilly, Sebastopol, Calif., 2010.

[7] U. Honegger, L. Glanzmann. Von der Schublade ins Netz (Titelgeschichte). *HOCHPARTERRE* (3) (2009). pp. 18-25.

__Daniel Burckhardt__, (b. 1971) received his diploma in mathematics from the ETH Zurich in 1996. In 2001, he got a MA in the history of science and technology from the TU Berlin. He has been developing content management and cross media publishing systems as well as search solutions and online communities. He is currently working at the Zentrum für Zeithistorische Forschung Potsdam in a federally sponsored project assembling an image database focusing on the visual arts in the GDR.

__Pepe Jürgens__, (b. 1971) received his diploma in grafic design and digital media from the Hochschule für Gestaltung Karlsruhe. In 2003, he founded Weltformat. Design (www.weltformat.de) which specializes on branding, user experience and interaction design as well as Rich Internet Application (RIA) development. His work has been awarded by and exhibited at, among others, the Festival Affiche Chaumont in 1998, Europrix in 2001, Feidad Award in 2002, the Red Dot Award in 2008, and Best of Swiss Web in 2008.