Similarity Preserving Snippet-Based Visualization of Web Search Results

Erick Gomez-Nieto, Frizzi San Roman, Paulo Pagliosa, Wallace Casaca, Elias S. Helou, Maria Cristina F. de Oliveira, *Member*, *IEEE*, and Luis Gustavo Nonato, *Member*, *IEEE*

Abstract—Internet users are very familiar with the results of a search query displayed as a ranked list of snippets. Each textual snippet shows a content summary of the referred document (or webpage) and a link to it. This display has many advantages, for example, it affords easy navigation and is straightforward to interpret. Nonetheless, any user of search engines could possibly report some experience of disappointment with this metaphor. Indeed, it has limitations in particular situations, as it fails to provide an overview of the document collection retrieved. Moreover, depending on the nature of the query—for example, it may be too general, or ambiguous, or ill expressed—the desired information may be poorly ranked, or results may contemplate varied topics. Several search tasks would be easier if users were shown an overview of the returned documents, organized so as to reflect how related they are, content wise. We propose a visualization technique to display the results of web queries aimed at overcoming such limitations. It combines the neighborhood preservation capability of multidimensional projections with the familiar snippet-based representation by employing a multidimensional projection to derive two-dimensional layouts of the query search results that preserve text similarity relations, or neighborhoods. Similarity is computed by applying the cosine similarity over a "bag-of-words" vector representation of collection built from the snippets. If the snippets are displayed directly according to the derived layout, they will overlap considerably, producing a poor visualization. We overcome this problem by defining an energy functional that considers both the overlapping among snippets and the preservation of the neighborhood structure as given in the projected layout. Minimizing this energy functional provides a neighborhood preserving two-dimensional arrangement of the textual snippets with minimum overlap. The resulting visualization conveys both a global view of the query results and visual groupings that reflect related results, as illustrated in several examples shown.

Index Terms—Multidimensional projection, web search visualization

1 Introduction

Searching for information on the web is routine task to millions of users. The typical procedure consists in providing textual queries to a search engine, which returns a ranked list of textual snippets each containing a content summary and a link to the referred document (or webpage). A ranked list of snippets is quite simple, straightforward to interpret, and it turns out to be effective in focused search tasks that require locating a particular webpage or document [1]. Nonetheless, it also has limitations likely to hamper user experience when exploring and analyzing search results in other scenarios. In fact, ranked lists fail to provide an overview of the collection retrieved, making it difficult and time consuming to figure out how documents relate content wise. For example, if a user queries Google's search engine on the keywords "jaguar features," the first page returned includes snippets on at least four distinct subjects, namely, the animal, the car brand, a fan club of old Jaguar cars, and a video game console. Surely users may refine the search; however, if s/he needs a global picture there is no other option but navigating through the pages in the list and manually group the snippets according to their topic.

Information visualization offers users more flexible mechanisms to inspect and navigate the result of textual queries. Some existing methods preserve the snippet list paradigm while enhancing it with visual resources such as color glyphs and tag clouds, adding information on the contents of the returned documents. Although interesting and potentially useful, those visual resources reveal no information on document relations, that is, which documents share similar content and how many different subjects appear in the search results. Other classes of methods replace the ranked list paradigm with alternatives such as thumbnails to favor better understanding of document content. However, those methods tend to be visually more intricate and demand greater user effort to locate and inspect specific documents. Moreover, current visualization methods use the full content of each document, therefore relying in pre-processed data not readily made available by standard search engines, thus preventing their usage as independent plug-ins.

Multidimensional projection techniques may be employed to generate visualizations that favor the perception of groups of similar documents. Such methods typically represent documents as points in a two-dimensional visual space, where neighboring points correspond to documents with similar content. However, points only

- *E. Gomez-Nieto, F.S. Roman, W. Casaca, E.S. Helou, M.C.F. de Oliveira, and L.G. Nonato are with Instituto de Ciências Matemáticas e de Computação (ICMC), Universidade de São Paulo, Avenida Trabalhador São-carlense, 400 - Centro, São Carlos, SP 13566-590, Brazil. E-mail: erick.gomez@ucsp.pe, {frizzi, wallace, elias, cristina, gnonato}@icmc.usp.br.*
- *P. Pagliosa is with Faculdade de Computação, Universidade Federal de Mato Grosso do Sul, CP 549, Campo Grande, MP 79070-900, Brazil. E-mail: pagliosa@facom.ufms.br.*

Manuscript received 22 Oct. 2012; revised 3 May 2013; accepted 30 Sept. 2013; published online 10 Oct. 2013.
Recommended for acceptance by M. Chen.
For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org, and reference IEEECS Log Number TVCG-2012-10-0235.
Digital Object Identifier no. 10.1109/TVCG.2013.242.

convey information on neighborhood relations. Replacing them with other geometric entities introduces overlapping, which impairs readability. The problem of arranging geometric entities in a two-dimensional visual space so as to ensure that similar objects remain neighbors while avoiding overlap is a recurrent problem in many different visualization contexts such as word-cloud construction, graph drawing, and label placement. Since finding an exact solution to the problem is computationally intractable, heuristics have been proposed, which nevertheless do not guarantee neighborhood preservation and visual space occupation. Therefore, techniques capable of arranging geometric entities in a visual space by taking into account their underlying object similarity while avoiding overlap are highly desirable, as they may benefit many distinct applications.

The technique introduced in this work provides an overlap removal mechanism that overcomes the drawbacks just discussed. More precisely, we propose an energy functional that considers both the overlapping between snippets and the neighborhood structure provided by a multidimensional projection. The minimum energy of such functional gives an arrangement of geometric entities in the visual space that preserves neighborhoods with minimum overlap.

We apply the proposed overlap removal mechanism in the context of snippet-based textual query web search result visualization, enabling two-dimensional layouts that preserve the simplicity and usability of textual snippets while emphasizing groups of content related documents. The unique combination of a similarity-based layout with the textual snippets brings out a powerful mechanism to organize and present textual search results that retain the familiar snippet paradigm, thus avoiding complex interfaces and visual metaphors. Moreover, the visualization is created only from the information in the textual snippets, rendering the proposed method computationally efficient and easy to plug into conventional search engines.

In summary, the main contributions of this work are:

- *2D snippet layout:* A new method, which we call *ProjSnippet*, to display the results of textual queries that integrates textual snippets and a multidimensional projection layout into a simple and intuitive visualization.
- *Energy functional:* The *ProjSnippet* layout relies on a new overlap removal energy functional that considers both the neighborhood relations between snippets and their overlapping in the visual space.

To assess the efficiency of the proposed technique toward facilitating user analysis of results from web queries, we have conducted two controlled user evaluations. The effectiveness of the proposed overlap removal mechanism is also assessed through comparisons with four well-known heuristics.

2 Related Work

Visualization techniques to support textual searching can be split into two major groups: methods for visualizing free text queries, which provide visual tools to assist the querying process, and techniques for visualizing the outcome of a particular query. We focus our discussion on the latter group to contextualize the technique proposed in this paper. Specifically, we discuss methods tailored to provide visual representations of the results of a textual search process, disregarding approaches aimed at visualizing document collections in general. Albeit potentially applicable in the scenario considered here, they typically neglect the specificities of this kind of application. A comprehensive survey that addresses aspects not covered here may be found in the works by Yao et al. [2] and Marchionini [3]. The book by Hearst [4] also surveys contributions on visual interfaces to support general search tasks.

Existing techniques for visualizing textual search results may be organized based on their underlying visualization paradigm.

Augmented list-based techniques rely on conventional list-based representations, but augmenting textual snippets with visual resources to facilitate interpretation. The webpage preview mechanism built into Google's search engine is a typical example. Another is TileBars [5] that places a colored bar next to each list entry to visually convey information on document length and the frequency of the query terms within the document. Similarly, the scheme by Heimonen and Jhaveri [6] places a small document-shaped icon on the left of each snippet to indicate query term frequency. HotMap [7] shows a more explicit visualization of query term frequencies as a color coded heat scale while still enabling a zoomed out view of the returned list. PubCloud [8] employs a different, domain specific augmentation mechanism. It enriches the conventional list-based visualization with tag clouds built from abstracts returned from searching the PubMed database, allowing users to select a specific topic in the tag cloud to filter the returned list accordingly. WordBars [9] provide an overview of the search results based on term frequency histograms and also allows interactive refinement of the search result, resorting the returned list. ResultMaps [10] enriches text listings with a Treemap-based visual representation that provides an hierarchical view of the repository during page navigation. Although useful in specific contexts, the reliance of augmented list-based methods on the sequential list paradigm hampers their effectiveness to convey document similarity relations and does not favor content-driven exploration.

Image-based techniques replace textual snippets by thumbnails that visually summarize content, for example, by displaying textually enhanced screenshots of the documents [11], [12], or a combination of images and textual information extracted from them [13], [1] or yet a mix of keywords and external images obtained by querying the web using those keywords [14]. Image-based methods also require access to the full document content. Moreover, their effectiveness in finding new documents (webpages) is arguable, being more appropriate to support re-finding tasks [1]. Again, gathering thumbnails according to the similarity of underlying documents is not addressed.

Plot-based visualizations depict search results through two-dimensional graphic plots that replace or complement the snippet list. Existing techniques vary greatly in the

visual resource employed to assist search. Nguyen and Zhang [15], for example, adopt a solar system metaphor that places the user query in the center of the system, around which returned documents orbitate based on their similarity with the query. Spoerri [16] organizes the documents in a spiral shaped list placing the better ranked ones closer to the spiral center, while icons identify their originating search engine. Nizamee and Shojib [17] adopt scatterplots to complement the snippet list and support filtering according to file type and publication year. Nowell et al. [18] propose a more powerful scatterplot-based method that combines glyphs, a coloring scheme and icons in a customized search interface. VisGets [19] combines information visualization widgets for searching and visualizing RSS feeds, enabling multiple visual facets that allow for simultaneous geographic, temporal and content visualization. As it requires geographic information, extending such a multiple facets approach to general search engines is not straightforward. Similarly, Fluid Views [20] integrates dynamic queries, semantic zooming, and dual layers to provide a visual overview of the information space while enabling direct access to individual results and supporting geographic and temporal visualizations.

Although effective to assist specific searches, none of the previous methods or metaphors show a global view of the results that favors the identification of groups of similar documents. This is the focus of the visualization in the PEx-Web tool [21], which employs multidimensional projections to enable users to identify and interact with groups of content-related documents.

Sallaberry et al. [22] combine graph layout and information visualization tools into a multilevel mechanism to show clusters of similar documents. Multidimensional projection and graph layout methods are quite effective for content-focused navigation. However, displaying a document summary in a manner that favors exploration in the context of web searching is not straightforward. Moreover, those visualization techniques, as most plot-based methods, do not incorporate the textual snippets, thus relinquishing their good properties and their already established usage as a mechanism for handling search results.

The visualization technique proposed in this work is innovative in its ability of displaying groups of similar documents and their rank in the search, while preserving the simple and familiar snippet paradigm.

3 Motivation and Design Rationale

The standard approach of displaying query results as a linear list of snippets is quite effective for most tasks performed by users of search engines. However, when users carry out an exploratory search on a broad topic or subject, linear lists are not so helpful, demanding additional effort toward gathering and mentally organizing the relevant information. The visualization technique introduced in this paper, the ProjSnippet, has been designed to assist users in these exploratory scenarios. As such, it is not intended as a substitute for lists of snippets, but rather as an additional resource to improve user experience in specific situations. Therefore, the proposed visualization system aims at helping users to gain a more comprehensive view of the query results, highlighting related documents and webpages while still retaining, as much as possible, the good properties of the conventional list-based paradigm, namely, the rank information and the summary content provided by the snippets.

Design rationale. To achieve the above desired properties, the visualization technique should comply with three major design goals:

Arrangement by similarity. A major requirement is to easily identify documents with similar content. A straightforward way to accomplish this is to build layouts where similar documents recovered by a search engine are placed close to each other. Such a layout may be naturally obtained with multidimensional projection techniques. Moreover, some multidimensional projection methods handle neighborhood structures explicitly, making it easier to keep control of those structures while computing the layout, justifying their adoption in the proposed solution.

Ranking identification. The success and effectiveness of search engines rely on a ranking mechanism that sorts documents according to their relevance. Therefore, any search result visualization technique must convey the document ranks. We incorporate such information into our visualization by controlling the size of the geometric entities (rectangles) that represent the documents returned by the query. This choice relies on the fact that human beings can easily discriminate objects according to their size.

Uncluttered layout. Visual clutter is prone to occur when arranging geometrical entities in a two-dimensional layout, mainly due to overlap. ProjSnippet avoids overlapping making use of a novel overlap removal mechanism that arranges the geometric entities representing documents while preserving the neighborhood structure provided by the multidimensional projection.

In the following, we detail the technicalities built into ProjSnippet so as to realize the design decisions just described.

4 Neighborhood Preserving Snippet Layout

The proposed technique comprises three steps as shown in the pipeline in Fig. 1: preprocessing of search results, multidimensional projection, and optimization. In the first step, each entry returned from a textual query is processed and its term frequency vector extracted (see [23] for details on term frequency extraction). Stemming and stopword removal are applied and Luhn's lower and upper cuts [24] established to compute the *tf-idf* vector representation of each snippet. Only the summary texts are processed, rather than the full content of the referred documents or webpages, which renders the visualization algorithm fast. Although considering the full document content might improve cluster quality, handling only the summary text favors interactivity and makes it easier to plug the proposed solution into existing standard browsers, which typically do not make available the full preprocessed content data.

Each term frequency vector may be handled as a point in a high-dimensional space that can be mapped to the visual space with a multidimensional projection technique. Albeit our current implementation adopts the *least squares projection* (LSP) [25]—due to its good accuracy in terms of

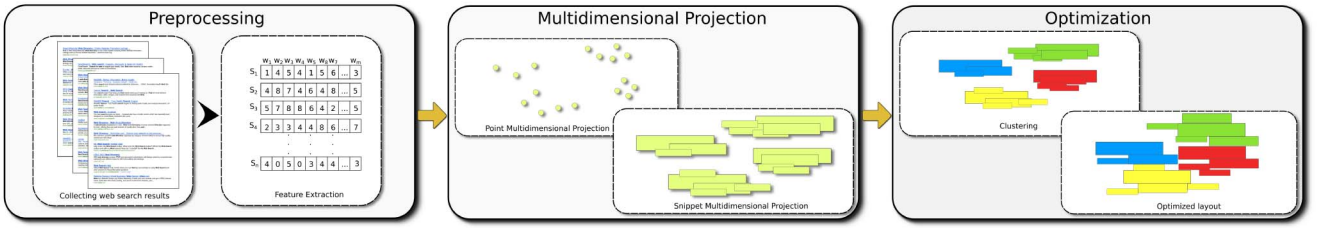


Fig. 1. Pipeline to generate the neighborhood-preserving snippet visualization. Snippet textual content is processed and represented as high-dimensional data points (left). The high-dimensional data are mapped to the visual space and snippets are embedded into rectangles (middle). Optimization is applied to avoid overlap while preserving neighborhoods.

distance preservation and low computational cost—any projection technique with similar properties might be employed (see [26] for an up-to-date survey of efficient multidimensional projection methods). The projection preserves much of the neighborhood structure of the original data, ensuring that similar instances are placed close to each other in the visual space.

The following step is to embed the content of each snippet within a rectangle whose bottom left corner is placed in the snippet's (or its high-dimensional data point) projected position. A rectangle's height and width are settled to reflect the rank of its corresponding snippet in the retrieved document list, so that better ranked snippets are assigned larger rectangles. The k-means++ algorithm [27] is applied to the projected layout to identify clusters of similar documents (taking as metric the euclidean distance in the visual space), and rectangles in the same cluster may be assigned the same color to highlight groups. The benefits of highlighting clusters when visualizing and analyzing textual search results have been pointed out by several authors [28], [29], motivating our choice.

A major drawback at this stage of the pipeline is that rectangles enclosing the snippets overlap considerably, impairing identification of individual entries and the perception of the document neighborhood structure. The final step (rightmost box in Fig. 1) optimizes the placement of the snippets so as to avoid overlapping while preserving data neighborhoods as computed by the projection. The optimization is driven by an energy functional, detailed next.

4.1 The Energy Functional

The energy functional E comprises two components, one that considers the overlap of snippets, denoted by E_O, and a second component related to the neighborhood relations resulting from the projection step, denoted by E_N. In mathematical terms, the energy E is written as

$$E = (1-\alpha)E_O + \alpha E_N, \tag{1}$$

where the parameter $\alpha \in [0,1]$ balances the relative contributions of both E_O and E_N in the total energy.

Energy E, as well as E_O and E_N, are functions of the coordinates of the bottom-left corners of the rectangles embedding the snippets, which initially correspond to the projected coordinates of the high-dimensional snippet vectors. We omit the independent variables from the equations to simplify the notation.

Overlapping energy. Aiming at enhancing overall visibility and readability of the visualization, the energy E_O must be defined so as to minimize the overlap/intersection of nearby snippets. This is achieved with a function that measures the distance between the left corners of the rectangles. This function is smooth, attains its minimum value when no overlapping takes place and takes higher values when rectangle overlap is greater. Smoothness is an important property here, as it allows resorting to simple and efficient optimization methods, which are mandatory for quick generation of the final visualization.

Let $\vec{x}, \vec{y} \in \mathbf{R}^n$ be the coordinate vectors of the bottom left corner of each rectangle and $\vec{v}, \vec{h} \in \mathbf{R}^n$ be vectors whose components are the vertical and horizontal dimensions of each rectangle. We first define two auxiliary functions to simplify the presentation:

$$[x]_+ = \begin{cases} x & x \geq 0, \\ 0 & x < 0, \end{cases}$$

and

$$O_{i,j}(\vec{x},\vec{h}) = \begin{cases} \frac{1}{h_j^4}\left[h_j^2 - (x_i - x_j)^2\right]_+^2 & x_i \geq x_j, \\ \frac{1}{h_i^4}\left[h_i^2 - (x_i - x_j)^2\right]_+^2 & x_i < x_j, \end{cases}$$

where x_i, h_i and x_j, h_j denote, respectively, the x-coordinates of the bottom left corner and the lengths of rectangles i and j. Notice that $O_{i,j}(\vec{x},\vec{h})$ is zero when there is no horizontal overlapping of rectangles i and j and attains its maximum value of 1 when the x-coordinate of the left corners of both rectangles coincide. Function $O_{i,j}$ works similarly if y-coordinates and heights are used as arguments, i.e., $O_{i,j}(\vec{y},\vec{v})$.

From definitions above, we set E_O as

$$E_O = \frac{2}{n(n+1)} \sum_{i=1}^{n} \sum_{j=i+1}^{n} [O_{i,j}(\vec{x},\vec{h}) O_{i,j}(\vec{y},\vec{v})], \tag{2}$$

where n is the number of projected points. The definition of $O_{i,j}$ clearly guarantees that E_O is continuously differentiable and it ranges in the interval $[0,1]$.

Neighborhood energy. The minimization of E_O spreads textual snippets in the visual space so as to prevent rectangles from overlapping. However, this minimization process is likely to spoil the neighborhood structure established by the multidimensional projection, placing similar snippets far apart in the final visualization.

The energy term E_N is introduced to balance the effect of the overlapping energy during optimization. In practice, the energy E_N is defined from a k-nearest-neighbor graph G constructed from the projected "snippet-vectors" (our implementation uses $k = 10$). To ensure G is connected,

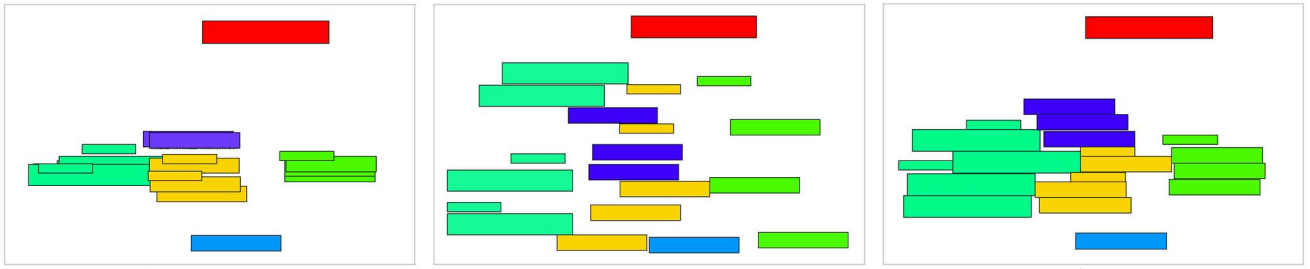


Fig. 2. (a) Layout without optimization; (b) overlapping energy E_O only; (c) both energies E_O and E_N combined ($\alpha = 0.5$).

any disconnected components resulting from constructing the k-nearest-neighbor graph are connected by adding to G the shortest edge between them.

Let L be the $n \times n$ matrix with entries l_{ij} given by

$$l_{ij} = \begin{cases} -1/|i| & \text{if } j \neq i \text{ and } \overline{ij} \text{ is an edge of G,} \\ 1 & \text{if } j = i, \\ 0 & \text{otherwise,} \end{cases} \quad (3)$$

where $|i|$ is the valence of node i.

Denoting by $\vec{x}^{\,0}$ and $\vec{y}^{\,0}$, the x and y coordinate vectors of the nodes of G (recall that $\vec{x}^{\,0}$ and $\vec{y}^{\,0}$ result from the multidimensional projection step) we define the differential vectors $\vec{\delta_x}$ and $\vec{\delta_y}$ as

$$\vec{\delta_x} = L\vec{x}^{\,0}, \qquad \vec{\delta_y} = L\vec{y}^{\,0}. \quad (4)$$

Notice that the components of $\vec{\delta_x}$ and $\vec{\delta_y}$ are given, respectively, by

$$x_i^0 - \frac{1}{|N_i|}\sum_{j \in N_i} x_j^0, \qquad y_i^0 - \frac{1}{|N_i|}\sum_{j \in N_i} y_j^0. \quad (5)$$

In less mathematical terms, $\vec{\delta_x}$ and $\vec{\delta_y}$ measure how much each node deviates from the average of its neighbors. Therefore, we define the neighborhood energy as

$$E_N = \frac{n^2}{2\left(\left\|\vec{\delta_x}\right\|^2 + \left\|\vec{\delta_y}\right\|^2\right)}\left(\left\|L\vec{x} - w\vec{\delta_x}\right\|^2 + \left\|L\vec{y} - w\vec{\delta_y}\right\|^2\right). \quad (6)$$

It is not difficult to realize that E_N will be minimal when $\vec{x}$ and $\vec{y}$ are such that their differentials $L\vec{x}$ and $L\vec{y}$ are proportional to the initial differentials $\vec{\delta_x}$ and $\vec{\delta_y}$. In other words, the energy term E_N is minimized when neighborhood relations are preserved during optimization. The unknown w is added to the optimization to ensure that any scale of the points is a minimum of the neighborhood energy (w is optimized together with $\vec{\delta_x}$ and $\vec{\delta_y}$).

The normalization factor $\frac{n^2}{2}\left(\left\|\vec{\delta_x}\right\|_2^2 + \left\|\vec{\delta_y}\right\|_2^2\right)^{-1}$ ensures that the range of E_N is in the same order of magnitude as E_O, so that both terms play similar roles (controlled by the parameter α) in the total energy E.

Fig. 2 illustrates the result of optimizing the layout shown in Fig. 2a. Fig. 2b shows the layout produced by optimizing the overlapping energy only, whereas Fig. 2c shows the outcome of the optimization procedure with both energy terms equally balanced.

4.2 Computational Aspects and Implementation

Bounds on the size of the visualization windows are imposed as constraints for the minimization of the energy (1). This is necessary because for a sufficiently large positive number K, the coordinate vectors $\vec{x} = K\vec{x}^{\,0}$ and $\vec{y} = K\vec{y}^{\,0}$, $(w = K)$ correspond to a global minimizer of E, as no overlap should happen and differentials are preserved by properly scaling the layout. However, the minimal solution given by scaling is prone to spread the snippets far apart, resulting in unpleasant and useless visualizations.

Therefore, denoting the horizontal and vertical bounds of the visualization window by x_{min}, x_{max} and y_{min}, y_{max}, the minimization problem becomes

$$\begin{aligned} \min \quad & (1-\alpha)E_O + \alpha E_N \\ \text{such that :} \quad & x_{min} \leq x_i \leq x_{max} - h_i, \quad i = 1, \ldots, n, \\ & y_{min} \leq y_i \leq y_{max} - v_i, \quad i = 1, \ldots, n, \end{aligned} \quad (7)$$

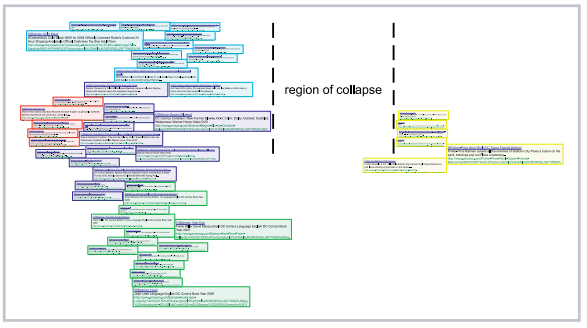
(recalling that variables x_i and y_i are encapsulated into E_O and E_N) which ensures that all rectangles lie within the visualization window, therefore preventing an exaggerated scaling effect.

The minimization is accomplished by a globally convergent local optimization method, namely the Method of Moving Asymptotes [30], available from the NLopt library at http://ab-initio.mit.edu/wiki/index.php/NLopt.

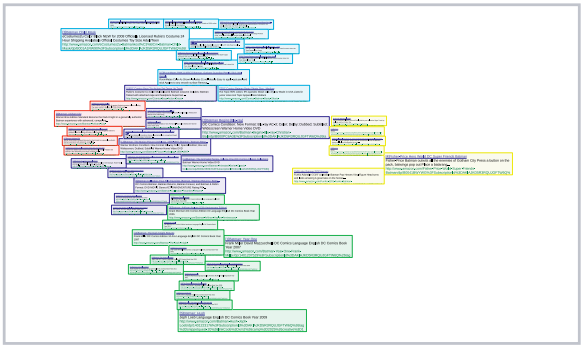
Reducing white space. To reduce white space in the final layout, we implemented a simplified version of the *seam carving strategy* [31]. The idea is to partition the "white regions" (snippet-free regions) of the visualization window into a rectilinear grid, as illustrated in Fig. 3a. Seams are then created by collapsing rectangular grid cells from left to right and then from top to down. A cell is collapsed if and only if all the snippets in the clusters affected by the collapse can be moved horizontally or vertically. If only a part of a cluster can be moved, no collapsing is performed. Such a simple carving mechanism runs quickly and it obviously preserves the clusters. Albeit more sophisticated carving strategies exist capable of further removing white space, they are computationally expensive and tend to spoil the neighborhood structures. The simple strategy described above is computationally efficient, produces pleasant layouts, and preserves clusters altogether (see Fig. 3).

5 Results, Comparisons, and Evaluation

In the following, we present examples illustrating the ProjSnippet visualization and its capability to globally convey the results of a web query while emphasizing related hits in a meaningful way. All examples have been generated in a Intel Core i7 CPU 920 2.66 GHz with 8 Gb



(a) Rectilinear grid to perform carving.



(b) With carving strategy.

Fig. 3. Reducing white space with a carving mechanism.

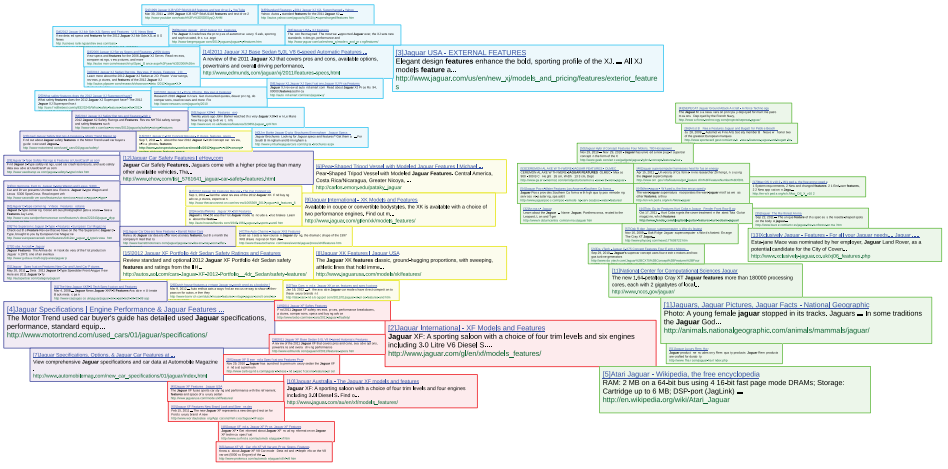
of RAM. A k-means++ clustering has been applied just to color the rectangles to visually highlight groups of similar snippets and provide some visual segregation to facilitate user inspection. The optimization procedure has been performed with $\alpha = 0.3$ (the default value in our current implementation).

The first example illustrates a visualization displaying the results of a query on the terms "jaguar features" submitted to Google's search engine. The view in Fig. 4a shows the 10 best ranked snippets shown in the first page. Fig. 4b displays a ProjSnippet view with the 64 best ranked snippets. Inspection reveals that the snippets on the left (cyan, red, blue, yellow) all refer to different models of Jaguar cars, whereas the green ones on the right refer to a surprising variety of topics. Those include multiple references to the wild animal (three snippets) and also to supercomputer models named Jaguar (two instances). There are also unique references to an earlier MacOs operating system named Jaguar, to a video game, a swimming pool brand, a hair product brand, an aircraft model, and a few other varied stuff. Looking at the left region, one identifies that most snippets in the blue cluster contain general references to the car brand, whereas the each of the three other clusters refer mostly to a specific Jaguar model, namely most yellow snippets refer to the XK model, cyan snippets refer to XJ, and red to XF models. There are some noticeable exceptions, for example, a yellow snippet refers to the XF model and a blue one refers to the XJ model. Still, overall the final layout depicts a representative overview of the search hits, as far grouping/separating similar/dissimilar results is concerned. Notice that it is pretty difficult to handle such a variety topics and subtopics in Google's list-based view, which indeed brings only results on cars, animals, and the game in the first page.

Fig. 5 shows the result of a search on Amazon's search engine, illustrating the potential of ProjSnippet in scenarios of searching for products at online stores. In these examples, the fields Title, Author, Brand, Color, Edition, Feature, Published Language, Manufacturer, Product Group, Size, Warranty, Year of Publication we used to generate the vector space model.

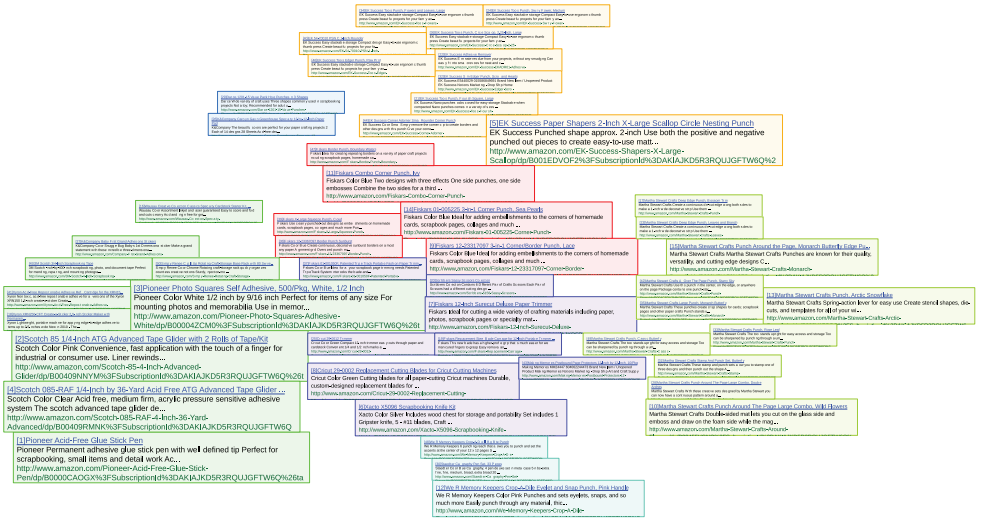
In Fig. 5a, we issued a query with terms "scrapbooking supplies" in the category "Office Products," from which 50 products were returned and visualized. Overall, the layout organization reflects a global arrangement of the products by brand and functionality. Most of the snippets in the top orange group refer to punch models from the same brand, EK Success. There are also snippets that refer to an adhesive

(a) First page of Google

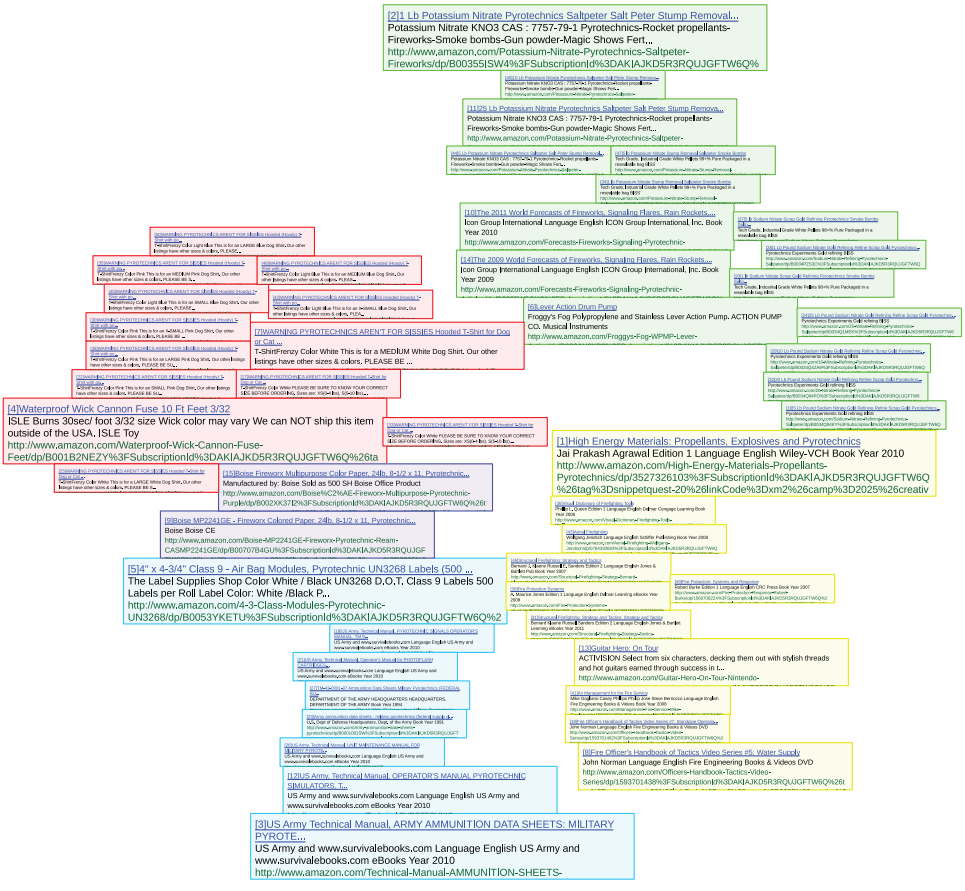


(b) ProjSnippet showing 64 snippets

Fig. 4. Google view and ProjSnippet view of the results of a query with terms "jaguar features."



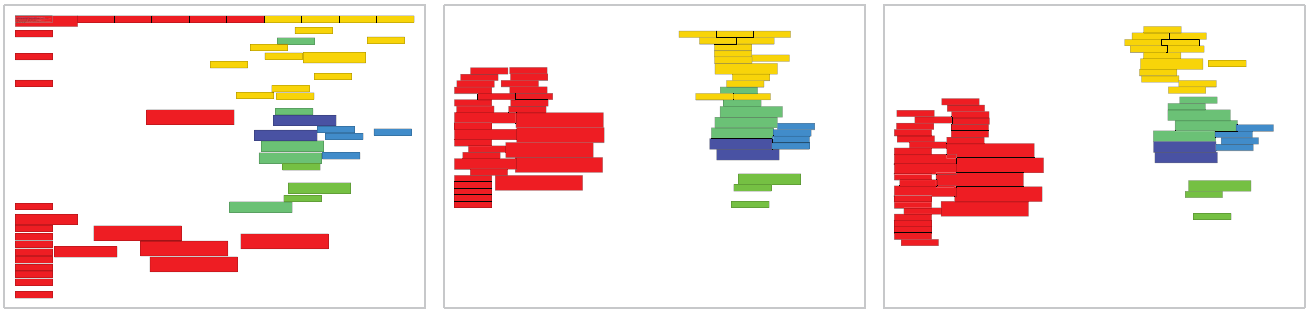
(a) 'scrapbooking supplies' (50 snippets)



(b) 'pyrotechnics supplies' (50 snippets)

Fig. 5. Searching for "scrapbooking supplies" (a) and "pyrotechnics supplies" (b) on Amazon.

remover and a rounder, both products from the same brand as the punches. The red group includes only products from Fiskars, also comprising punches and corner and border punches. The light yellowish green group on the right displays products from a particular brand (Martha Stewart Crafts), again including mostly models of punches. The green group of snippets on the left is more varied in content, including different products from various brands. Still, the majority refers to various types of adhesives and related products: tape, tape gliders, and tape refills; glue stick and varied occurrences of stickers, such as baby stickers and a sticker maker. The green group also includes a reference to cardstock and a reference to a craft storage rack. The central red group includes mostly references to utensils ranging from knife to cutting blades, from varied brands—including Fiskars, that has also utensils in the red group. The remaining



(a) $\alpha = 0.0$ (b) $\alpha = 0.5$ (c) $\alpha = 0.8$

Fig. 6. Effect of varying parameter α.

two groups identified by the clustering are both very small. The two blue snippets on the top, to the left of the yellow group, refer to a hole puncher and a paper pad. The three ones in cyan at the bottom refer to punch models from a single brand and a pen set.

Fig. 5b shows the visual result for a query on terms "pyrotechnic supplies" on Amazon. The search, conducted in category All, returned 50 hits shown in the visualization. The red group on the left contains snippets referring to dog T-shirts designs with a printed phrase that includes the term "pyrotechnics." The exception is the bigger snippet that refers to a waterproof fuse. Just under is a blue cluster of just two snippets that refer to colored paper from a brand named Fireworx. Further down are the cyan snippets, most referring to US Army technical manuals on military pyrotechnics. Again, there is one exception, a reference to label supplies. The green cluster on the top right region refers mostly to chemical supplies, but it also includes two references to a book and a reference to a toy. Finally, the yellow group contains instruction material, mainly books but also video, on safety, addressing topics as protection, strategy, tactics, and firefighting. Again, an exception is a reference to a video game. In this visualization, the groupings are very uniform in content and clearly separated by topic, except for the few unusual items, such as the labels or the video game.

Users can interact with the visualizations to navigate directly from the snippets, for example, to further inspect page contents, as afforded by the conventional list-based metaphor. Moreover, the examples shown clearly illustrate that the ProjSnippet visualizations are capable of depicting many snippets simultaneously in a clear and organized manner.

The effect of varying the parameter α in (1) is illustrated in Fig. 6a, which shows optimized layouts (without applying the carving mechanism) of the results from a query on terms "wave applications" posed to Bing's search engine. In Fig. 6a, only the overlapping energy has been considered ($\alpha = 0$). There is no overlapping, but neighborhoods are clearly not preserved and snippets are far too spread. Inspecting Figs. 6b and 6c, one observes how similar snippets get more tightly connect as α values increase. Notice that even with large values of α ($\alpha = 0.8$ in Fig. 6c) the snippets do not overlap unduly, showing the robustness of ProjSnippet as to the choice of α. No displacement of snippets occurs if $\alpha = 1$, since the result of the projection is clearly a minimizer of E_N.

Table 1 shows the energy values after optimization, as well as computational times (in seconds) for the examples presented in the paper (the search "Batman" is depicted in Fig. 3). The minimization strategy does a pretty good job in quite acceptable times, supporting interactive visualization.

5.1 Comparing with Overlap Removal Heuristics

Several heuristics have been proposed to arrange rectangular boxes in a viewport so as to avoid overlapping while still preserving the semantic relations among boxes as much as possible. To assess the effectiveness of the overlap removal mechanism built into ProjSnippet, we have compared it with four well-known heuristics, namely, VPSC [32], PRISM [33], Voronoi based [34], and RWordle-C [35], regarding the following metrics:

Euclidean distance. Denoting the original and final position of the bottom-left corner of each box by x_i^o and x_i, the euclidean distance metric is defined as

$$E = \frac{1}{n}\sum_i d(x_i^o, x_i), \tag{8}$$

where n is the number of boxes and d is the euclidean distance. This metric measures how much the boxes move during the overlap removal process. Less movement is preferred, since the original configuration is better preserved.

Layout similarity. This metric attempts to quantify how much neighborhood structures are affected by the overlap removal mechanism and it is derived from the Frobenius metric. The idea is to measure how much the length of Delaunay edges, computed from the original layout, changes after overlap removal. In mathematical terms, letting l_{ij}^o and l_{ij} denote the lengths of the Delaunay edges before and after overlap removal, the layout similarity is given by

TABLE 1
Optimization Results

Search	k	E_O	E_N	E	Time (s)
Pyrotechnics supplies	5	4.54^{-7}	1.97^{-4}	5.93^{-5}	0.24
Scrapbooking supplies	7	1.71^{-7}	5.09^{-4}	5.11^{-5}	0.23
Jaguar features	5	3.97^{-7}	1.91^{-4}	5.76^{-4}	0.27
Wave applications	6	5.38^{-6}	8.53^{-4}	3.44^{-4}	1.12
Batman	5	2.79^{-8}	4.86^{-5}	2.19^{-5}	0.26

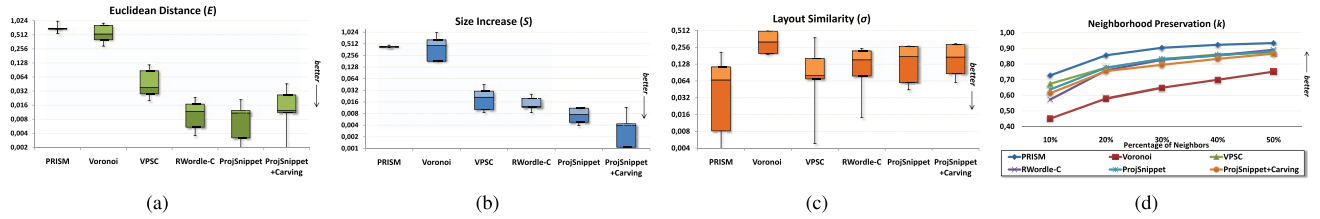


Fig. 7. Comparing ProjSnippet, VPSC, PRISM, Voronoi-based, and RWordle-C considering metrics E (a), σ (b), S (c), and k-nearest neighborhoods (d). Better performance is indicated by lower values for E and S, values of σ closer to 1, and higher k-nearest neighborhood curves.

$$\sigma = \frac{\sqrt{\left(\sum (r_{ij} - \overline{r})^2\right)/m}}{\overline{r}}, \quad \overline{r} = \frac{\sum r_{ij}}{m}, \tag{9}$$

where $r_{ij} = l_{ij}/l^o_{ij}$ and m is the number of edges in the Delaunay triangulation.

Size increase. Given the convex hulls C^o and C of the original and modified layouts, the size increase is measured as

$$S = \frac{area(C)}{area(C^o)}, \tag{10}$$

determining the relative changes in size as well as the compactness of the representation.

Neighborhood preservation. This metric accounts for neighborhood preservation and it computes the average percentage of preservation of the k-nearest neighbors of each box in the final layout.

Fig. 9 shows the layouts produced by the algorithms when applied to the data sets in the first column of Table 1. We run ProjSnippet with (ProjSnippet+Carving Strategy) and without (ProjSnippet) the seam-carving mechanism. One notices that ProjSnippet outputs a more organized layout, as compared with existing heuristics. On a first glance, its layouts resemble those obtained by RWordle-C, but ProjSnippet is more effective in preserving the grouping of similar elements (notice, for example, the red and the light blue groups on "Jaguar Features" and "Scrapbooking Supplies," respectively). Fig. 7 summarizes the quantitative results by the above metrics: ProjSnippet performed quite well, resulting in better values than the other methods, in most cases.

Fig. 8 shows a plot that considers all metrics simultaneously. Each overlap removal technique has been represented as a four-dimensional vector $(E_a, \sigma_a, S_a, k_a)$, where E_a, σ_a, S_a, k_a are the average values of the metrics E, σ, S, and k-nearest neighbors computed for each technique, over all data sets. The points labeled "best," "average," and "worst" in Fig. 8 were created artificially as four-dimensional vectors describing the best, average, and worst results computed considering all methods over all data sets. More precisely, the coordinates of the point labeled "best" are given by the best value of each metric obtained in the experiments (over all data sets). The same for the "worst" point, now considering the worst values, whereas the "average" point is obtained by averaging the values of each metric computed from all methods over all data sets. The four-dimensional vectors were projected with the LAMP multidimensional projection [26]. The techniques closer to "best," namely ProjSnippet and RWordle-C, are the ones with the best global performance, relative to all metrics.

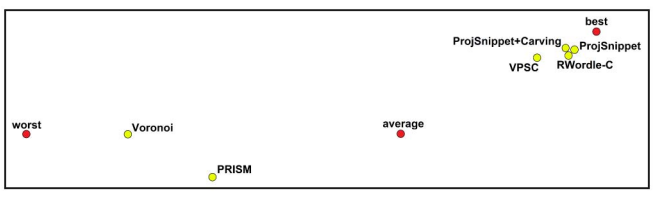


Fig. 8. Global comparison of overlap removal methods regarding the four metrics simultaneously.

5.2 User Evaluation

We conducted two controlled user evaluations: one comparing ProjSnippet with a standard list-based interface and another comparing it against other layout techniques, namely PRISM, VPSC, and RWordle-C. The first study was aimed at assessing whether the ProjSnippet layout allows users to find information faster than a list-based interface in tasks that require identifying groups of related sites, without significantly affecting precision. The second study was aimed at comparing ProjSnippet with other layout techniques, regarding the correctness of such tasks.

We formulated specific questions, detailed in Table 2, relative to the two queries already introduced, on "pyrotechnics supplies" (DT1) and on "jaguar features" (DT2). Each snippet in the interfaces shows its rank as returned by the search engine, so that the rank could be taken as a site identifier by subjects answering the questions, when required.

Both evaluations followed the same overall procedure comprised of four steps:

1. *Introduction*. Participants were given a brief explanation on the purposes of the study.
2. *Tool exposure*. Participants were shown basic functionalities and interaction functions of the prototypes interfaces (ProjSnippet and list based).
3. *User familiarization*. Participants interacted with their relevant interfaces for around 10 minutes, exploring a collection other than DT1/DT2.
4. *Evaluation*. Participants were invited to answer the questions in Table 2 on their assigned interface/collection.

For the first study, we invited 14 persons, all undergraduate or graduate students to execute the tasks using ProjSnippet and the standard list-based interface. Subjects were split into two groups of seven, so that a group used the list-based interface to answer questions on the "pyrotechnics supplies" hits and the ProjSnippet interface to

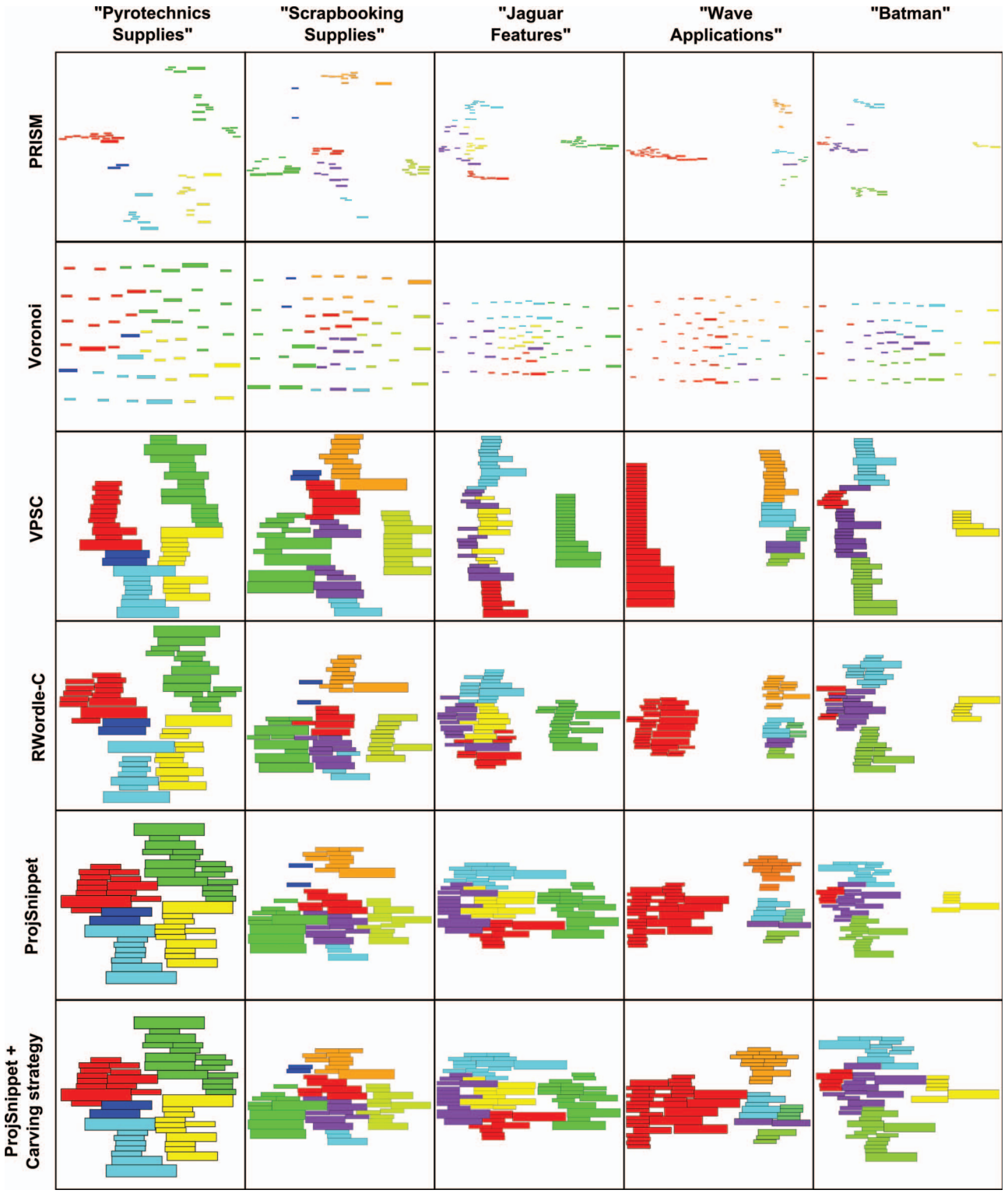


Fig. 9. Layouts produced by ProjSnippet, VPSC, PRISM, Voronoi-based, and RWordle-C for five distinct data sets.

answer questions on the "jaguar features" hits, whereas the other group used the interfaces in the reverse order. This study verified the following hypothesis:

- H: Users of ProjSnippet will spend less time to answer questions that require a global analysis of the query results (T1, T2, T3, and T4), with no significant loss in correctness.

We measured correctness of the answers (success rates) and the elapsed times taken to answer questions T1 to T5. To assess only the effectiveness of the proposed two-dimensional layout, we turned off the clustering mechanism, that is, clusters of similar snippets have not been colored or highlighted. For the sake of fairness, we also disabled the carving mechanism available in ProjSnippet. Results are shown in Fig. 10, for each task and each data set.

TABLE 2
Task Questions in User Tests

	Task Target	Question
T1	Identify groups of related sites	**DT1:** How many websites report on chemical supplies for pyrotechnics?
		DT2: How many sites depict content on the "Jaguar XJ" car model?
T2	Identify groups of related sites	**DT1:** Which sites present books, guides or papers about pyrotechnics?
		DT2: Which sites depict content on the "Jaguar XK" car model?
T3	Identify groups of related sites	**DT1|DT2:** How many different topics you can identify in the returned results?
T4	Find different sites addressing similar content	**DT1:** Find three websites announcing "T-Shirt for Dogs".
		DT2: Find two websites that refer to the animal "Jaguar".
T5	Find a particular site	**DT1:** Find a website that addresses the topic "wick cannon fuse".
		DT2: Find a website that includes the expression "Jaguar Features".

Hundred percent success rates were achieved on both collections on tasks T4 and T5. We applied a T-test with a 5 percent level ($\alpha = 0,05$) to check for statistical significance of the differences found.

One observes in Fig. 10a that subjects answering task T1 ("how many websites...") achieved better correctness on the ProjSnippet interface, on both collections. The difference, however, is not statistically significant. Participants answering task T2 ("which websites...") performed better on the list-based interface on DT1 ("pyrotechnics supplies"), and better with ProjSnippet on DT2 ("jaguar features"). Again, differences have not been found to be statistically significant. Finally, on task T3, which required identifying the multiple topics addressed, performance of ProjSnippet users was equivalent to those of the list-based on DT2, and better on DT1—the only difference found to be of statistical significance. Therefore, we conclude that in general users could identify the relevant sites with both interfaces. In fact, in most cases users of ProjSnippet performed better, albeit it is not possible to conclude that it favors an improvement in the success rates.

Fig. 10d confirms that ProjSnippet users took less time to answer all questions on both collections, with one single exception (task T5 on DT1). Differences have been found to be statistically significant for tasks T1, T2, and T3 on DT1 ("pyrotechnics supplies") and for all five tasks on DT2 ("jaguar features"). Table 3 shows the p-values computed

TABLE 3
ProjSnippet x List-Based T-Test p-Values

Task	"pyrotechnics supplies" (DT1)	"jaguar features" (DT2)
T1	0.002629	0.001074
T2	0.002387	0.002976
T3	0.036768	0.039171
T4	0.652960	0.005878
T5	0.780640	0.036399

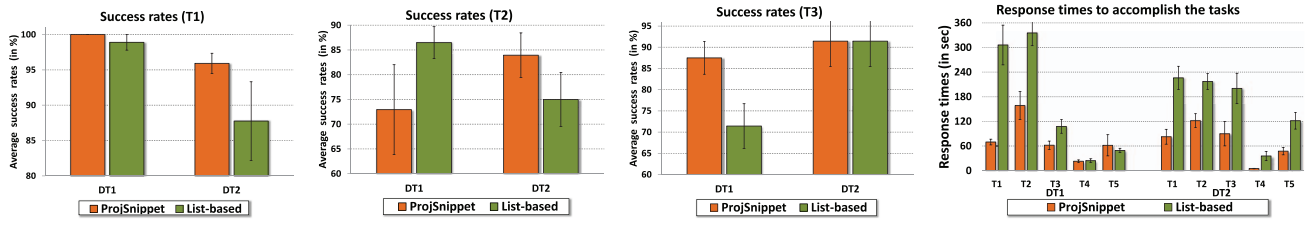
for the time differences in tasks, on both collections. Even for task T5 (identifying a particular website), ProjSnippet users performed better on DT2, whereas we expected scanning through the list view would be faster. These results confirm our initial hypothesis.

For the second user study, we invited 24 persons, again undergraduate or graduate students in computer science and none involved in the previous study. They were asked to answer the same questions, with evaluation taking place in two stages: first, each subject worked on DT1 displayed by a particular layout technique, and then on DT2 and a different layout technique. Subjects were initially randomly assigned to four groups, and each group of six assessed one layout technique. In the second stage, subjects were reassembled into four groups ensuring they would work on a layout technique different from the previous one. The working hypothesis can be stated as:

- H: Users of ProjSnippet will achieve better success rates than users of other layouts when answering questions that require a global analysis of the query results (T1, T2, and T3).

Subjects spent roughly 20 to 30 min to complete each stage. We measured correctness of the answers to questions T1 to T5. Again, 100 percent success rates were achieved on tasks T4 and T5 on both collections, and results for tasks T1, T2, and T3 are shown in Fig. 11. We applied one-way ANOVA at 5 percent level to check for statistical significance of the performance differences, corresponding values are shown in Table 4.

Analysis of Fig. 11 reveals that ProjSnippet users did better than the others on task T2, on both data sets. However, only in DT2, the performance difference was found to be significant. They also did better on Task T1 with DT2, whereas with DT1 the ProjSnippet layout came as second best. Again, differences have not been found to be statistically significant. On task T3 ProjSnippet came second



(a) (b) (c) (d)

Fig. 10. Correctness (average success rate) of subject answers in Tasks T1, T2, and T3 when using the ProjSnippet and the List-based interfaces, and the response times, in seconds, to all tasks.

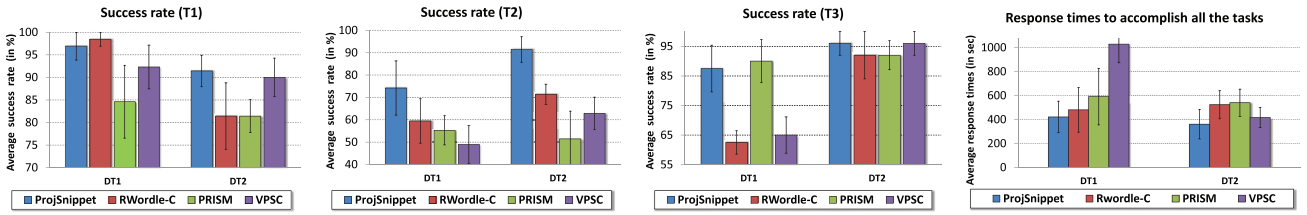


Fig. 11. Correctness (average success rate) of subject answers in Tasks T1, T2, and T3 for the different layout techniques, and overall response times (in seconds).

to PRISM on DT1, and first (but equivalent to VPSC) on DT2, with statistical significance found on DT1 only.

Although these results do not allow us to confirm the original hypothesis on the superiority of ProjSnippet, we observe it displayed good performance and a more stable behavior across tasks than the other techniques considered. Its users achieved higher or equivalent average success rates, as compared to users of other layouts, in four out of the six scenarios, and they also did well in the remaining two. They also spent less time executing their tasks, as observed in Fig. 11d.

Notice that Fig. 8 indicates the inferior quality of PRISM compared to ProjSnippet, RWordle-C, and VPSC regarding the quantitative measures, which is consistent with the observations from our user evaluation, where PRISM users did worse than users of the other three techniques in all except one task (T3/DT1). Moreover, RWordle-C users performed better in one evaluation (T1/DT1), being second in other two (T2/DT1 and T2/DT2), while VPSC tied with ProjSnippet in the first position once (T3/DT2), being second in the T1/DT2 test. Based on these evidences, one could claim that RWordle-C has a better performance than VPSC, again in agreement with Fig. 8. We point out, though, that additional studies should be conducted to further investigate the relationship between quantitative measures provided by the metrics and the qualitative results resulting from our user evaluation.

6 Discussion and Limitations

The ProjSnippet views of a collection of returned hits highlight their global relationships, as opposed to organizing them by their inferred relevance to the query. Still, the visualizations retain the simplicity of the snippet-based interaction, which from our perspective is a significant advantage. The underlying visualization paradigm is modified gently, requiring no substantial additional effort from users familiar with the standard list-based views. Even the aspect ratio of the rectangles reflects the nature of textual snippets, which are wider than higher. Users can still navigate the snippets and click to see a webpage preview (see Fig. 12) and to inspect the contents of particular documents.

Both the display size and the overall number of snippets exhibited affect visualization readability. The illustrative examples shown were handled on medium to large-sized monitors and were readily interpreted and easily read. Obviously, readability will be hampered on small monitors, in which case it is better to display less snippets. Finding an optimal number of snippets to display is not straightforward, since a decision involves many variables, such as the screen resolution and the nature of the search. Moreover, if the user-defined number of clusters is not set properly, nonsimilar snippets may end up in the same cluster and mislead user interpretation.

Our examples also indicate that creating the visualizations only from the summarized snippet texts is quite satisfactory. Similar entities are nicely clustered, although some apparent "outliers" may occur. Cluster quality might be further improved by inputting additional text from the documents into the clustering algorithm. Nonetheless, this would incur in higher computational cost and not necessarily produce better results, as text clustering is intrinsically fuzzy: in many situations, one could easily justify assigning a document to multiple clusters.

ProjSnippet requires a very simple preprocessing step, but some tricky issues remain. For example, setting appropriate values for Luhn's lower and upper cuts in scenarios where little information is available, as it is the case here, is not straightforward and deserves further investigation. In our examples, we typically employed a

TABLE 4
ANOVA p-Values Relative to Comparison of the Four Layout Techniques

Task	"pyrotechnics supplies" (DT1)	"jaguar features" (DT2)
T1	0.2427	0.5827842
T2	0.426798	0.020015
T3	0.012762	0.908402

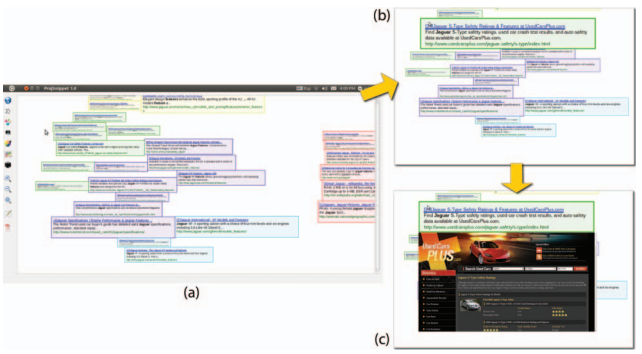


Fig. 12. Interactive exploration with ProjSnippet: (a) Main window, (b) on mouse over a snippet is highlighted and enlarged, (c) after a few seconds a preview of the page content is displayed.

lower cut of three and no upper cut and removed the query terms from the vector representation.

Finally, clustering in visual space will produce good results as long as the projection technique does a good job of preserving the relevant neighborhoods. Our choice of the LSP method is justified by its being known to perform quite well in terms of neighborhood preservation, which is confirmed by the results shown here. Nonetheless, if this is not the case, the visualization of groups may be misleading. Users may investigate alternatives playing with the number of clusters while observing the visualization.

7 Conclusions

We introduced ProjSnippet, a novel technique to visualize the collection of textual snippets returned from a web query. The method builds intuitive and meaningful layouts that optimize the placement of snippets by employing an innovative energy functional that considers both overlapping removal and preservation of neighborhood structures.

We showed results illustrating how the ProjSnippet layouts convey a global view of the results from a query while allowing for identifying similar content through a clustering mechanism. Since ProjSnippet relies only on information extracted from the textual snippets, it can be plugged into search engines in a straightforward manner, with a modest impact on the computational times. The unique combination of simplicity, low computational cost, and flexibility renders ProjSnippet an attractive alternative for visualizing web queries results. We are currently investigating interactive mechanisms to enable a free navigation in the snippet-based layout as well as on how to modify the energy functional to improve the layout so that it better highlights density information and similarity between neighboring snippets.

Acknowledgments

The authors acknowledge the financial support from FAPESP (the State of São Paulo Research Funding Agency), CNPq (the Brazilian Federal Research Funding Agency), and CAPES (Coordination for the Improvement of Higher Education).

References

[1] J. Teevan, E. Cutrell, D. Fisher, S.M. Drucker, G. Ramos, P. André, and C. Hu, "Visual Snippets: Summarizing Web Pages for Search and Revisitation," *Proc. ACM SIGCHI Conf. Human Factors in Computing Systems (CHI '09),* pp. 2023-2032, 2009.

[2] J.T. Yao, O. Hoeber, and X.D. Yang, *Supporting Web Search with Visualization,* pp. 183-214. Springer, 2010.

[3] G. Marchionini, "Exploratory Search: From Finding to Understanding," *Comm. the ACM,* vol. 49, no. 4, pp. 41-46, 2006.

[4] M. Hearst, *Search User Interfaces.* Cambridge Univ. Press, 2009.

[5] M.A. Hearst, "TileBars: Visualization of Term Distribution Information in Full Text Information Access," *Proc. ACM SIGCHI Conf. Human Factors in Computing Systems,* pp. 59-66, 1995.

[6] T. Heimonen and N. Jhaveri, "Visualizing Query Occurrence in Search Result Lists," *Proc. Ninth Int'l Conf. Information Visualisation,* pp. 877-882, 2005.

[7] O. Hoeber and X.D. Yang, "The Visual Exploration of Web Search Results Using Hotmap," *Proc. 10th Int'l Conf. Information Visualization,* pp. 157-165, 2006.

[8] B.Y.-L. Kuo, T. Hentrich, B.M. Good, and M.D. Wilkinson, "Tag Clouds for Summarizing Web Search Results," *Proc. 16th Int'l Conf. World Wide Web (WWW),* pp. 1203-1204, 2007.

[9] O. Hoeber and X. Yang, "Interactive Web Information Retrieval Using Wordbars," *Proc. ACM Conf. Web Intelligence,* pp. 875-882, 2006.

[10] E. Clarkson, K. Desai, and J. Foley, "Resultmaps: Visualization for Search Interfaces," *IEEE Trans. Visualization and Computer Graphics,* vol. 15, no. 6, pp. 1057-1064, Nov. 2009.

[11] H. Lam and P. Baudisch, "Summary Thumbnails: Readable Overviews for Small Screen Web Browsers," *Proc. ACM SIGCHI Conf. Human Factors in Computing Systems,* pp. 681-690, 2005.

[12] A. Woodruff, A. Faulring, R. Rosenholtz, J. Morrsion, and P. Pirolli, "Using Thumbnails to Search the Web," *Proc. ACM SIGCHI Conf. Human Factors in Computing Systems,* pp. 198-205, 2001.

[13] Z. Li, S. Shi, and L. Zhang, "Improving Relevance Judgment of Web Search Results with Image Excerpts," *Proc. 17th Int'l Conf. World Wide Web (WWW),* pp. 21-30, 2008.

[14] B. Jiao, L. Yang, J. Xu, and F. Wu, "Visual Summarization of Web Pages," *Proc. 33rd Int'l ACM SIGIR Conf. Research and Development in Information Retrieval,* pp. 499-506, 2010.

[15] T. Nguyen and J. Zhang, "A Novel Visualization Model for Web Search Results," *IEEE Trans. Visualization and Computer Graphics,* vol. 12, no. 5, pp. 981-988, Sept./Oct. 2006.

[16] A. Spoerri, "RankSpiral: Toward Enhancing Search Results Visualization," *Proc. IEEE Symp. Information Visualization,* pp. 208-214, 2004.

[17] M. Nizamee and M. Shojib, "Visualizing the Web Search Results with Web Search Visualization Using Scatter Plot," *Proc. IEEE Second Symp. Web Soc.,* pp. 5-10, 2010.

[18] L. Nowell, R. France, D. Hix, L. Heath, and E. Fox, "Visualizing Search Results: Some Alternatives to Query-Document Similarity," *Proc. 19th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval,* 1996.

[19] M. Dork, S. Carpendale, C. Collins, and C. Williamson, "VisGets: Coordinated Visualizations of Web-Based Information Exploration and Discovery," *IEEE Trans. Visualization and Computer Graphics,* vol. 14, no. 6, pp. 1205-1212, Nov./Dec. 2008.

[20] M. Dörk, S. Carpendale, and C. Williamson, "Fluid Views: A Zoomable Search Environment," *Proc. Int'l Working Conf. Advanced Visual Interfaces,* pp. 233-240, 2012.

[21] F.V. Paulovich, R. Pinho, C.P. Botha, A. Heijs, and R. Minghim, "Pex-Web: Content-Based Visualization of Web Search Results," *Proc. 12th Int'l Conf. Information Visualization,* pp. 208-214, 2008.

[22] A. Sallaberry, F. Zaidi, C. Pich, and G. Melançon, "Interactive Visualization and Navigation of Web Search Results Revealing Community Structures and Bridges," *Proc. Graphics Interface,* pp. 105-112, 2010.

[23] G. Salton, "Developments in Automatic Text Retrieval," *Science,* vol. 253, pp. 974-980, 1991.

[24] H.P. Luhn, "The Automatic Creation of Literature Abstracts," *IBM J. Research and Development,* vol. 2, no. 2, pp. 159-165, 1968.

[25] F. Paulovich, L. Nonato, R. Minghim, and H. Levkowitz, "Least Square Projection: A Fast High-Precision Multidimensional Projection Technique and Its Application to Document Mapping," *IEEE Trans. Visualization and Computer Graphics,* vol. 14, no. 3, pp. 564-575, May/June 2008.

[26] P. Joia, D. Coimbra, J. Cuminato, F. Paulovich, and L. Nonato, "Local Affine Multidimensional Projection," *IEEE Trans. Visualization and Computer Graphics,* vol. 17, no. 12, pp. 2563-2571, Dec. 2011.

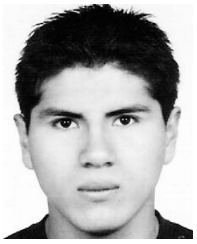
[27] D. Arthur and S. Vassilvitskii, "k-Means++: The Advantages of Careful Seeding," *Proc. 18th Ann. ACM-SIAM Symp. Discrete algorithms (SODA),* pp. 1027-1035, 2007.

[28] D.R. Cutting, D.R. Karger, J.O. Pedersen, and J.W. Tukey, "Scatter/Gather: A Cluster-Based Approach to Browsing Large Document Collections," *Proc. 15th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval,* pp. 318-329, 1992.

[29] P. Pirolli, P. Schank, M. Hearst, and C. Diehl, "Scatter/Gather Browsing Communicates the Topic Structure of a Very Large Text Collection," *Proc. ACM SIGCHI Conf. Human Factors in Computing Systems,* pp. 213-220, 1996.

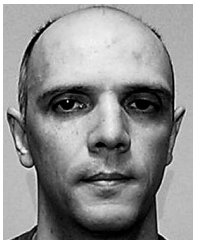
[30] K. Svanberg, "A Class of Globally Convergent Optimization Methods Based on Conservative Convex Separable Approximations," *SIAM J. Optimization,* vol. 12, no. 2, pp. 555-573, 2002.

[31] Y. Wu, T. Provan, F. Wei, S. Liu, and K.-L. Ma, "Semantic-Preserving Word Clouds by Seam Carving," *Computer Graphics Forum*, vol. 30, no. 3, pp. 741-750, 2011.
[32] T. Dwyer, K. Marriott, and P.J. Stuckey, "Fast Node Overlap Removal: Correction," *Proc. 14th Int'l Conf. Graph Drawing*, pp. 446-447, 2007.
[33] E.R. Gansner and Y. Hu, "Efficient Node Overlap Removal Using a Proximity Stress Model," *Proc. 16th Int'l Symp. Graph Drawing*, pp. 206-217, 2009.
[34] Q. Du, V. Faber, and M. Gunzburger, "Centroidal Voronoi Tessellations: Applications and Algorithms," *SIAM Rev.*, vol. 41, no. 4, pp. 637-676, 1999.
[35] H. Strobelt, M. Spicker, A. Stoffel, D. Keim, and O. Deussen, "Rolled-Out Wordles: A Heuristic Method for Overlap Removal of 2D Data Representatives," *Computer Graphics Forum*, vol. 31, no. 3, pp. 1135-1144, 2012.



Erick Gomez-Nieto received the BSc degree in 2009 in informatics engineering from San Pablo Catholic University - Arequipa, Peru, the MSc degree in 2012 in computer science from University of São Paulo - SP, Brazil. Currently, he is working toward the PhD degree at the University of São Paulo. He was a lecturer/researcher at Computer Science School of San Pablo Catholic University, in 2012/2013. His research interests include interactive visualization and geometry processing.

Frizzi San Roman received the informatics engineering degree from the San Pablo Catholic University, Peru, in 2013, and the MSc degree in computer science from the University of São Paulo, Brazil, in 2012. She is currently an assistant professor in the Computer Science Department at the San Pablo Catholic University. Her research interests include visual text analytics, visual data mining, and information visualization.



Paulo Pagliosa received the PhD degree in structural engineering from the Escola de Engenharia of São Carlos at Universidade de São Paulo in 1998. He is currently an associate professor of computer science in the Faculdade de Computação at Universidade Federal de Mato Grosso do Sul. His research interests include geometric processing, visualization, physics-based animation, and general-purpose computation on graphics hardware.

Wallace Casaca received the BSc and MSc degrees in pure and applied mathematics from São Paulo State University in 2008 and 2010, respectively. He is currently working toward the PhD degree at the University of São Paulo, Brazil and as a visiting scholar at Brown University. His research interests include image restoration, image segmentation, visualization, and geometry processing.

Elias S. Helou received the PhD degree in applied mathematics in 2009 from the University of Campinas - SP, Brazil. Since 2010, he has been a faculty member in the Department of Statistics and Applied Mathematics, University of São Paulo, where he is a professor of nonlinear optimization. His current research interests include tomographic image reconstruction, parameter selection for regularization of inverse problems and non-smooth convex/nonconvex optimization.

Maria Cristina F. de Oliveira received the BSc degree in computer science from the University of São Paulo, Brazil, in 1985, and the PhD degree in electronic engineering from the University of Wales, Bangor, in 1990 (now Bangor University). She is currently a professor at the Computer Science Department of the Instituto de Cincias Matemáticas e de Computação, at the University of São Paulo, Brazil, and has been a visiting scholar at the Institute for Visualization and Perception Research at the University of Massachusetts, Lowell, in 2000/2001. Her research interests include visual analytics, visual data mining and information visualization. She is currently the head of the Computer Science Department at ICMC, and the chief editor of the *Journal of the Brazilian Computer Society*, published by Springer. She is a member of the ACM, IEEE, and the Brazilian Computer Society.

Luis Gustavo Nonato received the PhD degree in applied mathematics from the Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro - Brazil, in 1998. He is an associate professor at the Instituto de Cincias Matemáticas e de Computação (ICMC) - Universidade de São Paulo (USP) - Brazil. He spent a sabbatical leave in the Scientific Computing and Imaging Institute at the University of Utah from 2008 to 2010. Besides having served in several program committees, he is currently in the editorial board of Computer Graphics Forum, he is the president of the Special Committee on Computer Graphics and Image Processing of Brazilian Computer Society and leads the Visual and Geometry Processing Group at ICMC-USP. His research interests include visualization and geometry processing. He is a member of the IEEE.