MIST: Multiscale Information and Summaries of Texts

Paulo Pagliosa*, Rafael Messias Martins†, Douglas Cedrim†, Afonso Paiva†
Rosane Minghim† and Luis Gustavo Nonato†

*FACOM – UFMS, Campo Grande
†ICMC – USP, São Carlos

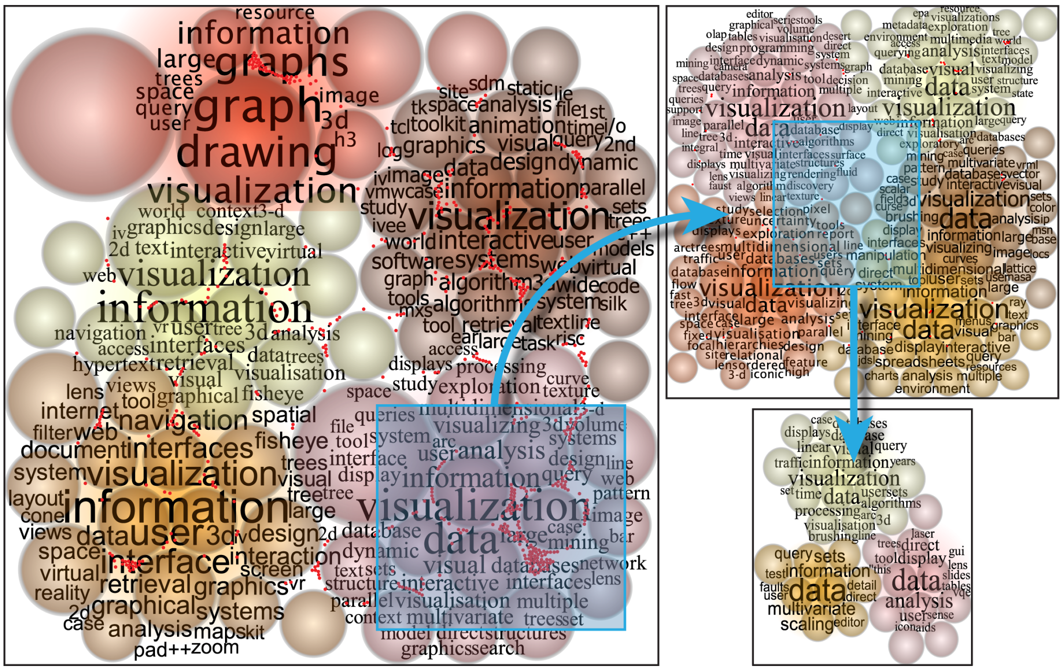


Fig. 1. MIST allows for the simultaneous visualization of individual documents as well as a summary of the document collection content, enabling multiscale exploration of subset of documents by content.

***Abstract*—Combining distinct visual metaphors has been the mechanism adopted by several systems to enable the simultaneous visualization of multiple levels of information in a single layout. However, providing a meaningful layout while avoiding visual clutter is still a challenge. In this work we combine word clouds and a rigid-body simulation engine into an intuitive visualization tool that allows a user to visualize and interact with the content of document collections using a single overlap-free layout. The proposed force scheme ensures that neighboring documents are kept close to each other during and after layout change. Each group of neighboring documents formed on the layout generates a word cloud. A multi-seeded procedure guarantees a harmonious arrangement of distinct word clouds in visual space. The visual metaphor employs disks to represent document instances where the size of each disk defines the importance of the document in the collection. To keep the visualization clean and intuitive, only the most relevant documents are depicted as disks while the remaining ones are either displayed as smaller glyphs to help convey density information or simply removed from the layout. Hidden instances are moved together with its neighbors during rigid-body simulation, should they become visible later, but are not processed individually. This shadow movement avoids excess calculations by the force-based scheme, thus ensuring scalability and interactivity.**

***Keywords*-information visualization; multidimensional projection; rigid-body simulation; word clouds; text visualization**

I. Introduction

Exploring document collections with the goal of identifying and extracting information of interest has become a central task in the current scenario of unprecedented growth of textual data, affecting a very large number of applications.

The usefulness of document collection visualization methods depends largely on how efficiently the underlying visual metaphor synthesizes and conveys the information one wants to grasp from the visualization. For instance, word clouds are quite effective in applications whose target is to provide a summary visualization of the content of documents while force-based methods are more appropriate for applications demanding the identification and manipulation of particular documents or groups of related documents.

In order to integrate a richer set of information in a single visualization, several methodologies propose the combination of multiple metaphors in a unified layout. Although multiple metaphors favor the simultaneous presentation of information with distinct natures, few existing approaches have in fact been successful in designing composite layouts that provide meaningful visualizations while avoiding distracting representations and preventing visual clutter. In particular, combining dynamical point layouts of textual data with content-based summarizations such as word clouds is a problem that has not been properly tackled yet. In fact, the few methods that propose the integration of these two visualization metaphors are still deficient in terms of the quality of the resulting layout, limited as to interactive resources, and not directly scalable.

This work proposes a new method to tackle the problem of providing word cloud summarization while allowing for interactive manipulation and visualization of individuals as well as groups of similar documents mapped on a visual plane. The methodology associates a word cloud to each group of similar documents displayed via a rigid-body simulation layout. Those word clouds are harmoniously merged on the visual space using a multi-seed scheme. A rigid-body simulation arranges the disks representing documents so as to ensure that similar instances are placed close to each other, as shown in Fig. 1. The formulation also takes into account the relevance of each document in the collection by changing the size of the corresponding disk according to the document relevance. To keep the layout clean and intuitive, only the most relevant documents are depicted as disks. Documents not inside the relevance threshold are either handled as tiny glyphs to maintain the perception of region density or simply removed from the visualization. Hidden instances are not taken into account during the rigid-body simulation to speed up the process, but they are moved with their neighbors so that they can be recovered when necessary. Such a passive transportation of hidden instances renders the rigid-body simulation scalable and enables fair navigation throughout the data space.

The approach is highly interactive. Disks representing documents can be dragged around the visual space to generate new layouts. The user can also focus the visualization on specific regions of the visual space in order to further explore subsets of documents. During navigation, the layout is dynamically updated so as to uncover hidden structures and content, rendering our approach quite intuitive and flexible.

Contributions The main contributions of this work are:

- A novel methodology for combining rigid-body simulation and word clouds which enables to visualize the similarity among documents as well as their content in an integrated manner on the same visual display.
- The use of a rigid-body simulation engine to arrange documents in the visual space preserving their neighborhood structure while avoiding overlapping documents. This is accompanied by an importance-driven approach to speed up the process.
- A new approach to build word clouds from multiple seeds positioned underneath clusters of corresponding documents.

As shown in the results presented in this paper, the harmonic combination of metaphors together with interactive and graphical resources provided endow the proposed system with a set of traits unusual in content-based document visualization approaches. Managing to code these features in a single presentation improves the state of the art in document analytics.

II. Related Work

Visualization of document collections is a quite prolific research area and the many existing methods vary greatly in their mathematical and computational foundation. In order to contextualize the proposed methodology we organize existing document collection visualization methods according to their visual metaphor, discussing overall properties and limitations, avoiding, though, going into fine details. A more thorough and detailed description can be found in the recent survey [1].

Word Cloud Kuo et al. [2] were one of the pioneers in using keywords as a visualization resource, proposing a line-by-line arrangement of words scaled according to their relevance. The large amount of white space reduced by Kuo's approach has been tackled by Kaser and Lemire [3] and Wordle [4], producing more pleasant and visually appealing layouts. The lack of semantic relation among words in the cloud has been handled by ManiWordle [5] through an interactive user assisted mechanism and by Cui et al. [6] through a force directed scheme that also allows for visualizing the temporal structure of documents. More recently, Wu et al. [7] presented a methodology that first computes semantic relationships and then uses multidimensional scaling to place the keywords on visual space. A Fiedler vector ranking is used to accomplish proper semantic word ordering in [8] with very effective results. Other methods such as SparkClouds [9] and Parallel Tag Clouds [10], augment word clouds with extra visual resources such as spark lines and parallel coordinates in order to better convey the summary content of the documents. Although very effective to reveal the essential information contained in a document collection, the word cloud paradigm alone does not identify the association of words to a particular document or group of documents or allows the examination of similarity between documents.

River Metaphor River metaphors have been acknowledged as an effective mechanism to visualize temporal thematic changes in document collections. Introduced in the ThemeRiver system [11], river metaphors have been further improved with sophisticated mechanisms to derive time-sensing topics [12] and layered visualizations able to depict birth, death and splitting of topic events [13]. EventRiver [14] makes use of a clustering scheme to group news that are similar in content and close in time, using a bubble metaphor whose thickness represents the number of documents and the length the duration of an event. History Flow [15] can also be seen as river-based metaphor designed to visualize editions of a document (or a document collection such as Wikipedia) made by different authors, emphasizing which parts survive along time. River metaphors provide a pleasant and intuitive visualization as to the temporal behavior of a document collection, but, similarly to word clouds, the technique does not allow immediate identification of specific documents, their relevance within the collection or their contribution to a topic. Moreover, interacting with the river layout to accomplish change in user perspective of the data set is not clearly feasible.

Linguistic-Based Methods that build visualizations based on semantically-defined linguistic structures have also been proposed in the literature. Word Tree [16], for example, uses a tree layout to visualize the occurrence of terms along with its following phrases, which are arranged in the descending branches of the tree. Phrase Nets [17] employs a graph-based layout where nodes correspond to a subset of words and edges correspond to semantic or lexical relation between words. Font size and edge thickness are used to visually map attributes such as the number of occurrences of a set of words and their relationship. A more sophisticated linguistic analysis is employed by DocuBurst [18], which makes use of an electronic lexical database and a radial space-filling tree layout to visualize document content in a lexical manner. Keim and Oelke [19] developed a method that employs semantic rules to segment a document into blocks and function words to map those blocks into feature vectors. The principal component of each feature vector is used to color the blocks, resulting in a fingerprint image of the document. In contrast to the other linguistic-based methods described above, Keim and Oelke's method allows for identifying and comparing specific documents in the data set, but compromising the legibility of their content.

Hierarchical Techniques that rely on hierarchical structures to enable level of detail content exploration and navigation are also present in the literature. Topic Island [20], for example, builds an hierarchy by applying wavelet analysis on a signal extracted from document words. The hierarchy allows for visualizing thematic changes and important parts of the document collection. InfoSky [21] visualizes hierarchically organized documents by subdividing the visual space using a recursive Voronoi diagram. Navigation throughout the hierarchy is enabled by a telescope-like zoom mechanism. HiPP [22] makes use of a cluster tree to hierarchically organize documents according to their similarity, performing the visualization of the hierarchy by projecting the nodes of the tree. Mao et al. [23] visualize documents using curves constructed from a generalization of n-grams and local averages, building the hierarchy by changing the support of the kernels used in the average computation. Although effective to build visual summarizations as well as to identify structures in the document topics, hierarchical techniques are not effective to associate content and documents when the hierarchy is done on the topics. Moreover, visualizing the hierarchical structure and the importance of each document simultaneously is not a straightforward task.

Force Directed Force directed techniques build visualizations by minimizing a functional defined from pairwise text similarity. FacetAtlas [24] employs a force direct graph layout for laying down clusters of nodes, enriching the visualization with density estimators as well as multifaceted data representation. TopicNets [25] computes the dissimilarity between topics extracted from a document collection and employs multidimensional scaling to place those topics in the visual space. A conventional force directed mechanism is then applied to place documents around corresponding topics. Streamit [26] uses the similarity among documents to define forces and Newton's law to update the position of those nodes in the visual space. Documents can dynamically be added into the system, enabling the visualization of streaming document data. Streaming data is also handled by TwitterScope [27], which converts a similarity graph into a map layout, employing a force scheme to remove overlap between nodes. The overlap removal problem has indeed been addressed by several methods [28], [29], [30], [31], [32] but preserving neighborhood structures during force simulation still remains an issue. Force directed methods allow for identifying a particular document and its neighborhood relation. However, they are not effective to provide a summary visualization of the content of the documents. The wordification technique proposed by Paulovich et al. [8] tackles this problem by combining a force directed algorithm, a multidimensional projection, and word clouds to enable visualizations in which similar documents are placed next to each other in the visual space, depicting their summary content through word clouds built from document clusters. Interacting with the layout so as to generate new arrangements while still incorporating information such as the importance of each document is an issue not handled by Paulovich's approach. Neither is the depiction of document relevance.

Other Metaphors Chuang et al. [33] proposed the Stanford Dissertation Browser, which enables a set of visualization resources to investigate the impact of interdisciplinary research between academic departments at Stanford University. PCA-based projections and radial layouts are used to visually investigate shared ideas and interdisciplinary collaboration. Document Cards [34] presents a quick overview of a collection or single documents by adopting the rationale of top trumps game cards, which use expressive images and facts to highlight important key terms and representative images extracted from a document. Chuang's and the Document Cards solutions are suitable to provide a compact visualization of a large document collection. Nonetheless, they fail to show inter-document relationships.

Existing methodologies are not designed to associate dynamic and interactive mechanisms to build a visual summary of a document collection while still enabling the visualization of the similarity between documents, their importance into the collection, and their relationships such as links and citations. No available technique integrates all these operations. As a consequence, they force users with tasks that require details to employ more than one visualization, that might not be linked together. The technique presented in this paper encompasses a set of traits that allow for those simultaneous functionality, making possible to have an overall idea of document contents while establishing a correspondence between words and documents, on a clutter free layout. Additionally, the user can dynamically rearrange the layout and further examine a subset of documents and their neighborhood, adding exploration flexibility to the approach.

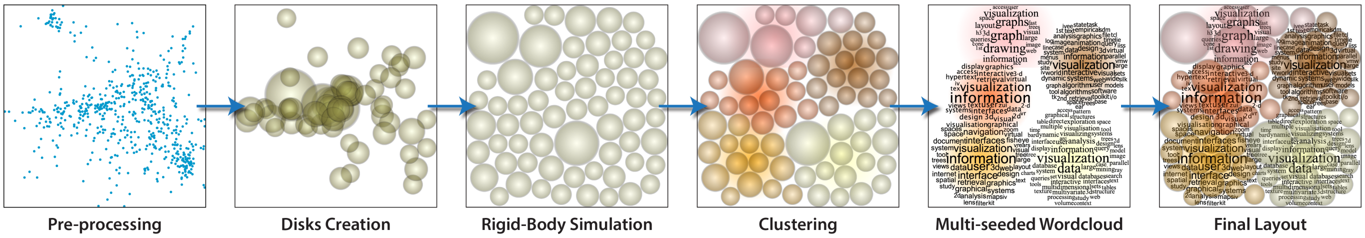


Fig. 2. Main steps of the MIST pipeline.

III. The MIST Technique

The MIST technique comprises three main stages: pre-processing, disks creation and rigid-body simulation, and word cloud generation, as illustrated in Fig. 2. Three tasks are performed during pre-processing. The first task is a process for keyword extraction, to generate the word cloud in the third step of the pipeline. Keywords are also used to compute the similarity between documents. This similarity is used in the second step, as input to a multidimensional projection method that maps documents to a 2D visual space. The importance of each document in the collection is also computed as a pre-processing task and is based on the connectivity between the individual documents, given by a user or application defined graph. In the second stage of MIST, a rigid-body simulation engine arranges a set of disks representing documents, with their size determined by the importance of the document, so as to avoid overlap while still preserving the neighborhood structures provided by the initial multidimensional projection. In the third and last stage of the pipeline, documents are clustered according to their neighborhood and word clouds are generated and harmoniously merged to produce the final layout. Technical details of each task are provided below.

A. Pre-processing

Keywords Extraction Documents are processed to extract keywords used in the word cloud construction as well as to generate the vector space representation needed for multidimensional projection. A list of keywords is associated to each document in the collection by simply removing stopwords (non-informative words such as articles and prepositions) and reducing relevant words to their radicals. The number of occurrences of each keyword is also stored to be used in the word cloud and to build the multidimensional vector space [35]. Terms that occur too sparsely or too often in all documents are also removed as they have little differential capability. The vector representation of documents is used by a function that determines the dissimilarity between them in the subsequent step.

Multidimensional Projection (MP) The rigid-body simulation assumes as initial condition a set of disks in the visual space. The center of those disks are provided by mapping attribute vectors from the feature space to the visual space using an MP, which ensures that the original neighborhood structures are preserved as much as possible in the visual space. In our system we are using the LSP (Least Square Projection) [36] with Euclidean distances as dissimilarity tool, since LSP's effectiveness in projecting textual data with high precision has already been established. In our experiments, we perform LSP with 10 nearest neighbors for each cluster and 10 control points. Any neighborhood preserving point placement can be applied in its place.

Ranking Changing the size of visual entities according to the relevance of the document they represent is a useful resource to better analyze and explore document collections. The relevance of a document in the collection can be calculated in many different ways, but the computation of the dominant eigenpair of a stochastic matrix is one of the most common approaches [37]. In the proposed system, when documents are linked according to some relation, as for example citations or hyperlinks, the relevance of each document is given by the solution the eigenvector problem $\mathbf{M}\mathbf{x} = \mathbf{x}$ where each column of $\mathbf{M}$ corresponds to a document and an entry M_{ij} is non-zero when the document i is link to the document j, in that case, $M_{ij} = 1/\text{outdeg}(i)$. When the document collection has no link information, the relevance is computed from the k-nearest neighbors (KNN) graph.

B. Disks Creation and Rigid-Body Simulation

Each point resulting from the MP gives rise to a rigid-body, which in our case is a disk centered in the projected point with radius defined by the relevance of the document associated to the projected point. At this stage the disks will overlap considerably, requiring high degree of interaction to access groups and individuals. If there is enough visual space to accommodate all individuals, it should be possible to avoid clutter and still allow the same interpretation of neighborhood provided by the projection.

Simply spreading the disks towards removing overlaps is not effective, as neighborhood structures can be damaged. Therefore, we propose a force scheme that pushes overlapping disks apart while still avoiding to disturb the neighborhood given by the initial placement. In such a scheme, we employ a rigid-body dynamics engine called *Box2D* [38] to update the position of the center of each disk along the simulation from impulses and forces acting on the bodies. Impulses are automatically applied by the physics engine to spread apart intersecting bodies [39]. In addition, we introduce attractive forces in order to prevent neighbor disks to distance from each other during the simulation. The attractive force $\mathbf{F}_i$ associated to a disk i is computed as:

$$\mathbf{F}_i = \sum_{j \in N_i^k} m_i m_j d_{ij} \frac{\mathbf{x}_i - \mathbf{x}_j}{\|\mathbf{x}_i - \mathbf{x}_j\|_2}, \qquad (1)$$

where $m_i = \pi r_i^2$ is the mass and r_i is the radius of the disk, $d_{ij} = \max\{0, \|\mathbf{x}_i - \mathbf{x}_j\|_2 - (r_i + r_j)\}$, and N_i^k corresponds to the KNN set of the disk. In our experiments, the value of k is chosen $\sim 2\%$ of the total number of documents.

The edges of the main window can also be considered as rigid bodies, thus enforcing the disks to be confined within the viewport. This feature can be set on/off in our system.

C. Multi-seeded Word Cloud

Once disks have been properly placed by the rigid body simulation mechanism, a summary of the corresponding documents content is depicted using word clouds. In order to provide as much information as possible we build a set of word clouds and merge them in an harmonious manner. More specifically we employ k-means++ [40] to group the disks, where the number k of clusters is set by the user. Users can also change the font and color used in each word cloud. Keywords associated to each document, computed in the preprocessing step, are filtered out according to their relevance, given by the number of occurrences of each keyword present in the documents making up the cluster. As illustrated in Sec. IV, the system also admits the relevance of keywords as input, thus enabling the use of more sophisticated semantic preserving techniques (eg. [8]) defining the relevance.

The center of the bounding box of each cluster c_i is used to seed the construction of its corresponding word cloud. The set of words W_i associated to the cluster c_i are sorted in descending order according to their relevance and the most relevant keyword is placed in the center of the bounding box b_i of the cluster c_i. The size of the font is defined as a function of the width of the bounding box b_i, that is, starting from a maximum font size the font size of the most relevant keyword is decreased until it reaches 70% of the width of b_i. The font size of the remaining keywords are linearly interpolated between the maximum font size and the minimum font size and those keywords are horizontally placed inside b_i following the same procedure proposed in RWordle [41]. The minimum font size is pre-fixed. The procedure stops when no more room is left in b_i to place a keyword.

Since the bounding box of distinct clusters may overlap, the bounding rectangle of a keyword to be placed inside b_i can intersect the bounding rectangle of keywords in other clusters. In order to accelerate the check of intersections between keywords in distinct clusters we use a dynamic tree to store the rectangles containing each keyword, being the data structure adopted a variation of the Presson's dynamic bounding volume tree employed in the Bullet Physics SDK [42].

D. Computational Aspects

The size of the disks representing documents and the physical units used in the rigid-body simulation are derived from a viewport defining the visualization window. The most relevant document is represented by the larger disk with radius $r_{\max}$ while the disk representing the less relevant document has the radius $r_{\min}$. The radii $r_{\max}$ and $r_{\min}$ are set so that the area of the larger and smaller disk correspond to 5% and 0.5% of the viewport area, respectively. The radius of the remaining disks are linearly interpolated between $r_{\max}$ and $r_{\min}$ according the their relevance value. The number of disks to be visualized is also dictated by the viewport area, that is, the area of the disks are summed up in descending order until reaching 75% of the viewport area. The first t of the remaining disks, sorted according to their relevance, are set with a small area (4 pixels) and included in the physical simulation while the least relevant instances are not visible and they move together with their closest visible disk. The value of t can be adjusted and it is by default set as two times the number of visible disks.

When the user selects a rectangular region on the viewport for further exploration (navigation zoom), the selected region is mapped to a new viewport and the whole process described above is executed in this new viewport but constrained to the subset of instances contained in the region selected by the user, as illustrated in Fig. 1.

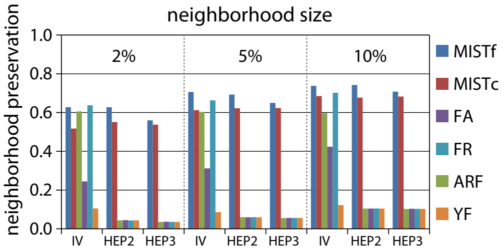
Another aspect that deserves to be discussed is how to speed up the physical simulation by changing the length unit. The Box2D rigid-body simulation engine is optimized to bodies with sizes ranging from ~ 0.1 and ~ 10 meters [38]. Therefore, the coordinates of the center and the radius of each disk are scaled to range in that interval.

IV. RESULTS AND COMPARISON

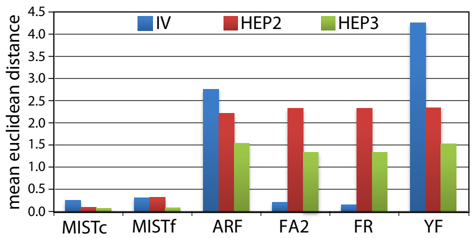
In order to show the performance of our approach we accomplish tests and comparison employing 3 data sets. One data set includes all papers (616 documents) published in the IEEE Information Visualization Conference (IV) from 1995 to 2002 [43], along with metadata such as title, keywords, abstract and references. We also used the High Energy Physics data set (`kdl.cs.umass.edu/data/hepth/hepth-info.html`), which contains information derived from the abstract and citation of papers in theoretical high-energy physics. More specifically, we split the original data set in two subsets with 2000 (HEP2) and 3000 (HEP3) randomly obtained instances.

In order to assess the effectiveness of the proposed approach regarding layout organization, neighborhood preserving, and compactness we have run a set of comparisons against other 4 force-directed techniques described in the literature. Fruchterman-Reingold method (FR) [29] uses a spring model to place the nodes of a graph in the visual space, using a reduced number of parameters. Yifan Hu's (YH) method [32] makes use of force-directed approach that employs a multilevel scheme to avoid local minima. The Attractive and Repulsive Forces (ARF) method [30] extends the spring-based methodology aiming at reducing empty spaces on the layout. These three techniques allow for representing nodes as disks but they only avoid overlap between disks if the radius is constant, that is, all disks have the same size. Overlap between disks with varying radii can be handled by tuning certain parameters. In contrast to FR, YH and ARF, the ForceAtlas2 (FA2) [31] technique manages to treat overlap with disks of varying sizes. It makes use of linear attraction and repulsion forces to arrange disks in the visual space while avoiding overlap.

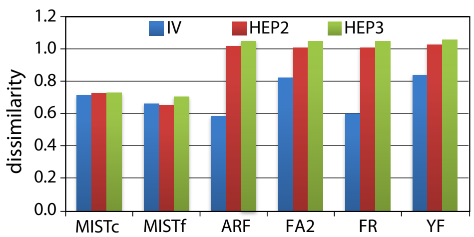
Four distinct measurements have been used to compare MIST against these techniques, namely neighborhood preser-



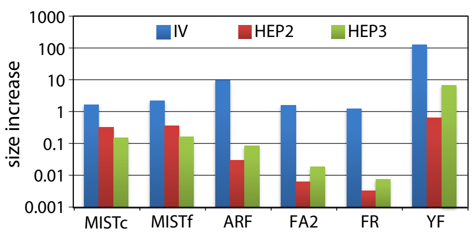
(a) KNN



(b) Distances



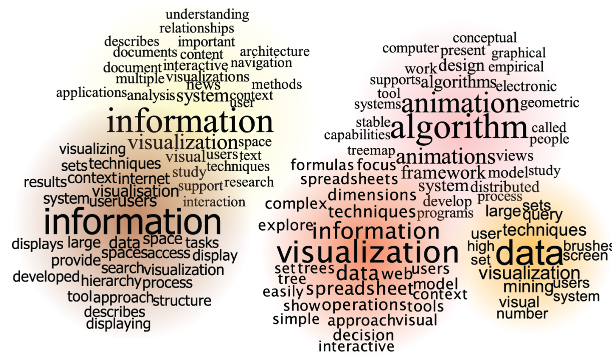
(c) Layout Dissimilarity



(d) Size Increase

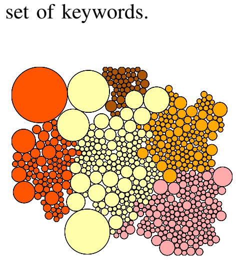
Fig. 3. Quantitative evaluation of neighborhood preservation and layout compactness for MIST, FR, ARF, FA2, and YH techniques.

(a) Wordification

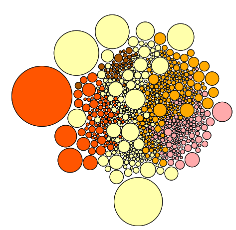


(b) MIST

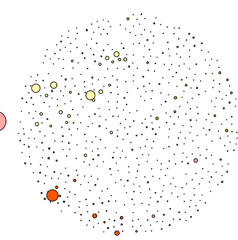
Fig. 4. Comparing the word clouds produced by (a) Paulovich's Wordification method [8] and (b) MIST using the same number of clusters (5) and the same set of keywords.



(a) MIST



(b) FA2



(c) FR

Fig. 5. Layouts produced by MIST, FA2, and FR. MIST presents much better cluster preservation.

vation (KNN), Euclidean distance average preservation, layout dissimilarity, and size increase.

Neighborhood preservation computes the average percentage of the KNN of the initial layout that is preserved after force simulation. The *Euclidean distance preservation* is given by $E = \frac{1}{n} \sum_i \|\mathbf{x}_i^o - \mathbf{x}_i\|_2$, where $\mathbf{x}_i^o$ and $\mathbf{x}_i$ are the initial and final position of the center of the disks, n is the number of disks and d is the Euclidean distance. This value measures how much disks move during the force simulation since the least they move the better the preservation of initial configuration. *Layout dissimilarity* quantifies the degree to which neighborhood structures are affected by the force simulation. The idea is to measure how much the length of edges in the initial KNN graph change after simulation. In mathematical terms, letting l_{ij}^o and l_{ij} denote the lengths of the edges before and after simulation, the layout dissimilarity is given by

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

where $r_{ij} = l_{ij}/l_{ij}^o$ and m is the number of edges in the KNN graph. Small values of dissimilarity correspond to good layout preservation. Finally, given the convex hulls C^o and C of the original and modified layouts, the *size increase* is measured as $S = \text{area}(C)/\text{area}(C^o)$ and it determines the relative changes in size as well as the compactness of the modified layout. The closer the size increase is to 1 the better, meaning that the technique does not require too much extension or shrinking of the original designed area.

Fig. 3 shows the result of the chosen measurements applied to the layouts produced by MIST and the methods used for comparison when visualizing the data sets IV, HEP2, and HEP3, assuming the same initial configuration, that is, the same MP before applying any of the methods. Relevance information has not been taken into account, thus the radii of the disks are constant, which is a necessary condition for FR, YH, and ARF to handle the overlap between disks adequately. Since MIST allows to confine the layout within a box we ran tests with two versions of the algorithm. Constrained MIST (MISTc) is the layout with box confinement and MISTf runs with the confinement turned off. Notice that MIST performs significantly better than other methods in all measurements when dealing with the larger data sets HEP2 and HEP3, and performs quite well also with the IV data set.

The quantitative results shown in Fig. 3 attest that MIST produces more compact results while better preserving the initial neighborhood structure provided by the MP. Fig. 5 shows the layout resulting from MIST, FA2, and FR when visualizing the IV data set taking relevance information into account. Colors show how neighborhoods are preserved after simulation. More specifically, we first apply k-means to cluster, in the visual space, the output of the MP, running then the physical/force simulation to observe how well the clusters (neighborhood structures) are preserved. The results clearly show that MIST preserves better the clusters while still keeping a compact layout. In contrast to MIST, that takes only the number of clusters as needed parameter, the user has to set various parameters for the FR method to avoid overlap. As it can be seen from Fig. 5c, tuning those parameters to obtain a good compromise between overlap removal and compactness is not an easy task.

Fig. 4 compares the word clouds generated by MIST and the word clouds generated by the wordification mechanism proposed by Paulovich [8]. While Paulovich's method discriminates groups of word clouds by using white space, MIST employs background colors, which allows for identifying the distinct word clouds while maintaining a more uniform and harmonious layout. In this comparison we employed the same number of clusters and keywords for both methods. The relevance of each keyword was also obtained by [8].

We conclude this section discussing two important functionalities of MIST, namely, linked documents visualization and exploratory navigation. As illustrated in Fig. 6, MIST allows for highlighting link relation between documents. In the specific example of Fig. 6, which represents a co-citation network, when the user selects a document (red ring) all documents that cite the selected one are highlighted. This functionality is quite useful when analyzing scientific paper reference and citation networks, as well as web pages. Fig. 6 shows that the user can also drag documents around to modify the layout and generate alternative arrangements.

Another important functionality of MIST is the exploratory zoom illustrated in Fig. 1. The tiny glyphs moved together with the disk help convey the density information of each cluster. The density information is useful to indicate which regions in the visualization enclose hidden data and should be further explored. To visualize the information hidden because of low relevance, the user simply draws a rectangle defining the region to be explored and MIST re-starts the whole visualization process from the data enclosed by the user defined region. As one can see from Fig. 1 the navigation can be run recursively, enabling a detailed investigation of the document collection. Notice that context is not lost during navigation, as a previously defined exploratory windows are kept in the visualizations until the corresponding windows are closed.

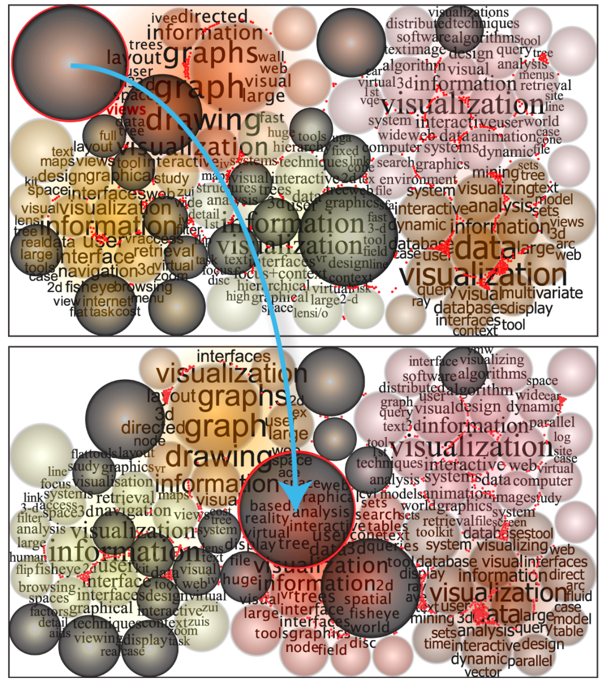


Fig. 6. Highlighting linked documents.

V. Discussion and Limitations

In our experiments we noticed that FR and FA2 do not converge for certain initial configurations. Additionally, YH tends to produce elongated layouts when parameters are changed to avoid overlap for size varying disks. We have not faced any of these issues in any of the experiments performed with MIST.

A fair comparison of the methods in terms of computational times is not a simple task, as FR, ARF, FA2, and YH are implemented in Java and our system is completely coded in C++. However, to give a rough idea of processing times, MIST provides visualizations at interactive rates, as it processes only the data that will be presented in the visualization window. The other methods, however, take dozen of seconds to process HEP2 and HEP3 data sets, impairing interactivity.

The mechanism we use to make MIST scalable, that is, moving unseen documents together with their nearest neighbor, may impact negatively in neighborhood preservation within the navigation window. Although the attraction forces tends to bring neighboring instances close to one another, some visible disk can be placed between two hidden neighbors, preventing the force simulation to approximate those disk when they become visible. We are investigating alternatives to overcome this limitation.

VI. Conclusions

In this paper we have presented MIST, a novel methodology to visualize the collection of textual documents that enables the simultaneous visualization of multiple levels of information in a single layout. The method builds upon a physical rigid-body simulation engine to arrange disks in the visual space avoiding overlap while preserving neighborhood structures. Individuals that do not reach a degree of relevance, calculated over an input relationship graph, can be explored in further levels of exploration. The provided comparisons have shown MIST to produce more compact layouts and to preserve neighborhood relationships more efficiently than other force-direct methods. Moreover, considering only visible data in the physical simulation renders MIST scalable to a reasonable degree, enabling a flexible exploration of document collection. By extending the initial layout over the available space and considering relevance, clutter is avoided. By integrating the layout with word clouds, summarization of the data set content helps guide the user towards interesting groups of documents to be further explored. Additionally, the layout are pleasant and harmonious. We are currently investigating how to incorporate out-of-core data handling into MIST, what should allow the navigation throughout large document collections feasible.

ACKNOWLEDGMENT

We would like to thank the anonymous reviewers for their useful comments to improve the quality of this paper. The authors are supported by CAPES, CNPq and FAPESP.

REFERENCES

[1] A. Alencar, M. de Oliveira, and F. Paulovich, "Seeing beyond reading: a survey on visual text analytics," *WIREs Data Mining Knowl. Discov.*, vol. 2, pp. 476–492, 2012.

[2] B. Kuo, T. Hentrich, B. Good, and M. Wilkinson, "Tag clouds for summarizing web search results," in *WWW'07*, 2007, pp. 1203–1204.

[3] O. Kaser and D. Lemire, "Tag-cloud drawing: Algorithms for cloud visualization," in *WWW'07*, 2007.

[4] F. B. Viegas, M. Wattenberg, and J. Feinberg, "Participatory visualization with wordle," *IEEE TVCG*, vol. 15, no. 6, pp. 1137 –1144, 2009.

[5] K. Koh, B. Lee, B. Kim, and J. Seo, "Maniwordle: Providing flexible control over wordle," *IEEE TVCG*, vol. 16, no. 6, pp. 1190 –1197, 2010.

[6] W. Cui, Y. Wu, S. Liu, F. Wei, M. Zhou, and H. Qu, "Context-preserving, dynamic word cloud visualization," *IEEE Comp. Graph. Appl.*, vol. 30, pp. 42–53, 2010.

[7] Y. Wu, T. Provan, F. Wei, S. Liu, and K.-L. Ma, "Semantic-preserving word clouds by seam carving," *CGF*, vol. 30, no. 3, pp. 741–750, 2011.

[8] F. V. Paulovich, F. M. B. Toledo, G. P. Telles, R. Minghim, and L. G. Nonato, "Semantic wordification of document collections," *CGF*, vol. 31, no. 3, pp. 1145–1153, 2012.

[9] B. Lee, N. H. Riche, A. K. Karlson, and S. Carpendale, "Sparkclouds: Visualizing trends in tag clouds," *IEEE TVCG*, vol. 16, no. 6, pp. 1182 –1189, 2010.

[10] C. Collins, F. B. Viegas, and M. Wattenberg, "Parallel tag clouds to explore and analyze faceted text corpora," in *VAST*, 2009, pp. 91 –98.

[11] S. Havre, I. C. Society, E. Hetzler, P. Whitney, and L. Nowell, "ThemeRiver: Visualizing thematic changes in large document collections," *IEEE TVCG*, vol. 8, pp. 9–20, 2002.

[12] S. Liu, M. X. Zhou, S. Pan, Y. Song, W. Qian, W. Cai, and X. Lian, "TIARA: Interactive, topic-based visual text summarization and analysis," *ACM Trans. Intell. Sys. Tech.*, vol. 3, no. 2, pp. 25:1–25:28, 2012.

[13] W. Cui, S. Liu, L. Tan, C. Shi, Y. Song, Z. Gao, H. Qu, and X. Tong, "Textflow: Towards better understanding of evolving topics in text," *IEEE TVCG*, vol. 17, no. 12, pp. 2412–2421, 2011.

[14] D. Luo, J. Yang, M. Krstajic, W. Ribarsky, and D. Keim, "EventRiver: Visually exploring text collections with temporal references," *IEEE TVCG*, vol. 18, no. 1, pp. 93 –105, 2012.

[15] F. B. Viégas, M. Wattenberg, and K. Dave, "Studying cooperation and conflict between authors with history flow visualizations," in *SIGCHI*, 2004, pp. 575–582.

[16] M. Wattenberg and F. B. Viégas, "The word tree, an interactive visual concordance," *IEEE TVCG*, vol. 14, no. 6, pp. 1221–1228, 2008.

[17] F. van Ham, M. Wattenberg, and F. B. Viegas, "Mapping text with phrase nets," *IEEE TVCG*, vol. 15, no. 6, pp. 1169–1176, 2009.

[18] C. Collins, S. Carpendale, and G. Penn, "Docuburst: Visualizing document content using language structure," *CGF*, vol. 28, pp. 1039–1046, 2009.

[19] D. A. Keim and D. Oelke, "Literature fingerprinting: A new method for visual literary analysis," in *VAST*, 2007, pp. 115–122.

[20] N. E. Miller, P. Chung Wong, M. Brewster, and H. Foote, "Topic islands: a wavelet-based text visualization system," in *VIS*, 1998, pp. 189–196.

[21] K. Andrews, W. Kienreich, V. Sabol, J. Becker, G. Droschl, F. Kappe, M. Granitzer, P. Auer, and K. Tochtermann, "The infosky visual explorer: exploiting hierarchical structure and document similarities," *Information Visualization*, vol. 1, no. 3/4, pp. 166–181, 2002.

[22] F. V. Paulovich and R. Minghim, "Hipp: A novel hierarchical point placement strategy and its application to the exploration of document collections," *IEEE TVCG*, vol. 14, no. 6, pp. 1229–1236, 2008.

[23] Y. Mao, J. V. Dillon, and G. Lebanon, "Sequential document visualization." *IEEE TVCG*, vol. 13, no. 6, pp. 1208–1215, 2007.

[24] N. Cao, J. Sun, Y.-R. Lin, D. Gotz, S. Liu, and H. Qu, "Facetatlas: Multifaceted visualization for rich text corpora." *IEEE TVCG*, vol. 16, no. 6, pp. 1172–1181, 2010.

[25] B. Gretarsson, J. O'Donovan, S. Bostandjiev, T. Hllerer, A. U. Asuncion, D. Newman, and P. Smyth, "TopicNets: Visual analysis of large text corpora with topic modeling." *ACM TIST*, vol. 3, 2012.

[26] J. Alsakran, Y. Chen, D. Luo, Y. Zhao, J. Yang, W. Dou, and S. Liu, "Real-time visualization of streaming text with a force-based dynamic system," *IEEE Comput. Graph. Appl.*, vol. 32, no. 1, pp. 34–45, 2012.

[27] E. R. Gansner, Y. Hu, and S. C. North, "Visualizing streaming text data with dynamic maps," *http://arxiv.org/abs/1206.3980*, 2012.

[28] A. S. Spritzer and C. M. Dal Sasso Freitas, "Design and evaluation of magnetviza graph visualization tool," *IEEE TVCG*, vol. 18, no. 5, pp. 822–835, 2012.

[29] T. M. J. Fruchterman and E. M. Reingold, "Graph drawing by force-directed placement," *Software: Pract. and Exp.*, vol. 21, no. 11, 1991.

[30] M. M. Geipel, "Self-organization applied to dynamic network layout," *Int. J. Mod. Phys. C*, vol. 18, no. 10, pp. 1537–1549, 2007.

[31] H. Gibson, J. Faith, and P. Vickers, "A survey of two-dimensional graph layout techniques for information visualisation," *Information visualization*, 2012.

[32] Y. F. Hu, "Efficient and high quality force-directed graph drawing," *The Mathematica Journal*, vol. 10, pp. 37–71, 2005.

[33] J. Chuang, D. Ramage, C. Manning, and J. Heer, "Interpretation and trust: designing model-driven visualizations for text analysis," in *SIGCHI*, 2012, pp. 443–452.

[34] H. Strobelt, D. Oelke, C. Rohrdantz, A. Stoffel, D. A. Keim, and O. Deussen, "Document cards: A top trumps visualization for documents," *IEEE TVCG*, vol. 15, no. 6, pp. 1145–1152, 2009.

[35] G. Salton, "Developments in automatic text retrieval," *Science*, vol. 253, pp. 974–980, 1991.

[36] F. V. Paulovich, L. G. Nonato, R. Minghim, and H. Levkowitz, "Least square projection: A fast high-precision multidimensional projection technique and its application to document mapping," *IEEE TVCG*, vol. 14, no. 2, pp. 564–575, 2008.

[37] A. Langville and C. Meyer, *Google's PageRank and beyond: The science of search engine rankings*. Princeton University Press, 2009.

[38] E. Catto, *Box2D - a 2D physics engine for games*, 2011. [Online]. Available: http://box2d.org

[39] ——, "Iterative dynamics with temporal coherence," *Game Dev. Conf.*, pp. 1–24, 2005.

[40] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding," in *SODA*, 2007, pp. 1027–1035.

[41] H. Strobelt, M. Spicker, A. Stoffel, D. Keim, and O. Deussen, "Rolled-out Wordles: A Heuristic Method for Overlap Removal of 2D Data Representatives," *CGF*, vol. 31, pp. 1135–1144, 2012.

[42] E. Coumans, *Bullet 2.80 Physics SDK Manual*, 2012. [Online]. Available: www.bulletphysics.com/ftp/pub/test/physics/Bullet_User_Manual.pdf

[43] J.-D. Fekete, G. Grinstein, and C. Plaisant, "IEEE InfoVis 2004 contest – The history of InfoVis," 2004. [Online]. Available: www.cs.umd.edu/hcil/iv04contest