Mutual Text-Image Queries

Timo Götzelmann,[1] Pere-Pau Vázquez,[3] Knut Hartmann,[1] Tobias Germer,[1] Andreas Nürnberger,[2] and Thomas Strothotte[1]

[1] Graphics and Interactive Systems Group
[2] Information Retrieval Group
University of Magdeburg / Germany
{timo,knut,germer,tstr}@isg.cs.uni-magdeburg.de
nuernb@iti.cs.uni-magdeburg.de

[3] Modeling, Visualization, and Graphics Interaction Group
Technical University of Catalonia / Spain
ppau@lsi.upc.edu

Abstract. This paper presents a novel approach to support students to learn a comprehensive domain-specific terminology and to understand textual descriptions of complex-shaped objects. We implemented an experimental system where learners can interactively explore textual descriptions and 3D visualizations. We propose a method for hierarchical content representations of text documents and views on 3D models. Based on these data structures, user interactions on texts and interactive 3D visualizations are transformed into queries to an information retrieval system. This enables us to coordinate the content of both media, to focus the attention of the user on the most salient graphical objects, and to suggest potential relevant text segments in large text documents and appropriate views on 3D models to illustrate the spatial relations between the relevant domain objects of the query. Finally, we demonstrated this concept in an interactive tutoring environment based on standard textbooks on human anatomy.

1 Introduction

Learning material often incorporates text and images in order to benefit from the complementary capabilities of both media. While complicated processes and contextual information are best described textually, illustrations are more suitable to provide information about visual and spatial attributes of complex-shaped objects. *Secondary elements* in illustrations, such as *textual annotations* or *labels*, establish links between both media. Annotations support two search tasks: readers may use them in order to determine the corresponding visual object for unknown terms in the text; but the content of textual annotations also provides links to more elaborative descriptions for visual objects in corresponding text passages (see Fig. 1). To focus the attention of the learner on salient objects, experienced illustrators also utilize many graphical abstraction techniques.

The creation of expressive illustrations that carefully reflect the subject matter of the corresponding text is very expensive. Moreover, the space required to include these illustrations into tutoring materials also increases the costs for printed versions. Hence, illustrators often aim to minimize the number of illustrations which results in illustrations with many labels. But the limited cognitive capacity makes both search tasks mentioned in the last paragraph more complicated. Furthermore, learners often have to understand complicated spatial configurations—but illustrations depict 2D projections of 3D objects.

Therefore, on-line versions of these tutoring materials offer new possibilities to support learners: Efficient retrieval mechanisms ease the access to sections containing information required for user-specific tasks in large documents while interactive 3D visualizations support the mental reconstruction of complex spatial configurations. Moreover, a dynamic layout of secondary elements can make illustrations more effective.

This paper proposes a novel technique to coordinate the content of visual and textual elements in on-line tutoring systems. As the majority of the application domains define a standard terminology, unified retrieval methods are applied on large text corpora and annotated 3D models. User interactions on both media—text and graphics—initiate queries to an information retrieval system; the retrieval results are then presented to the learner. Our system addresses both main search tasks of text↔image relations. On the one hand, users can select text passages to obtain corresponding renditions of 3D models. The relevance of terms and their corresponding visual objects is considered to determine good views on 3D models. As a single view might not sufficiently correspond to the content of text paragraphs, these 3D visualizations can be explored interactively. On the other hand, user selected points of 3D views are considered as queries; corresponding text passages of the underlying text corpus are automatically obtained and presented to the learner.

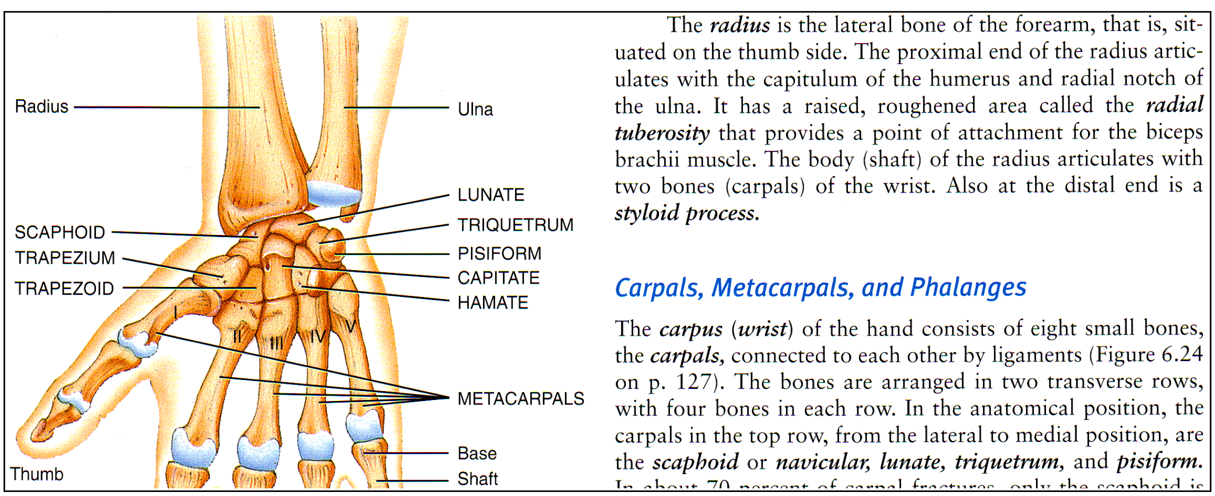
After this introduction, Sec. 2 presents the related work of those fields involved in this paper and introduces their fundamental terminology. Sec. 3 discusses the architecture of our framework, which comprises some preprocessing steps for information retrieval techniques (Sec. 4). Sec. 5 describes how both search tasks (see Fig. 1) are handled in our approach and how the search results are presented in both cases. To intuitively refine the search queries, Sec. 6 introduces techniques for both images and text. Finally, Sec. 7 discusses some limitations of our work and the future work to be done in this field.

2 Related Work

This section presents related work and fundamental terminology of research areas that influenced our approach: automatic text illustration, information retrieval, and the determination of good viewpoints for 3D models.

2.1 Text illustration

The automatic generation of technical documentation with coordinated content in several media (e.g., text, images, and animation) was pioneered by the COMET [7] and WIP system [28]. These research prototypes aim at adopting the content of on-line documents to user specific tasks, preferences, or contextual requirements. Moreover, coherent multi-lingual variants should be generated from a single source. Therefore, a multitude of planning mechanisms is used in the automatic content selection and in media-specific content generators. But the enormous amount of required formal representations as well as the need to approve all document variants according to legal aspects prevented the application of this powerful technique.



The *radius* is the lateral bone of the forearm, that is, situated on the thumb side. The proximal end of the radius articulates with the capitulum of the humerus and radial notch of the ulna. It has a raised, roughened area called the ***radial tuberosity*** that provides a point of attachment for the biceps brachii muscle. The body (shaft) of the radius articulates with two bones (carpals) of the wrist. Also at the distal end is a ***styloid process.***

Carpals, Metacarpals, and Phalanges

The ***carpus*** (***wrist***) of the hand consists of eight small bones, the ***carpals,*** connected to each other by ligaments (Figure 6.24 on p. 127). The bones are arranged in two transverse rows, with four bones in each row. In the anatomical position, the carpals in the top row, from the lateral to medial position, are the ***scaphoid*** or ***navicular, lunate, triquetrum,*** and ***pisiform.***

Fig. 1: Media coordination in tutoring material: The most relevant objects for a learning task are emphasized in descriptive texts and illustrations (Source: [23, p.127])

Based on the observation that authors of scientific textbooks, technical documentations, or maintenance manuals reuse and improve large text databases with a standardized domain-specific terminology, several researchers developed (semi-)automatic systems to illustrate on-line texts. In order to select appropriate 3D models and to control render parameters for expressive visualizations, the TEXTILLUSTRATOR [20] employs a database of annotated 3D models. This system extracts exact matches between terms in the document and media descriptors and highlights those objects that are mentioned in the visible text portion in a 3D visualization. This illustrative browsing metaphor enables a quick access to large scientific texts but fails to detect morphological and syntactical variants of terms, synonyms, and implicit semantic associations between domain objects. Therefore, the Agile system [12] integrates a morphological analysis and a shallow syntactic text parsing. Moreover, the Agile system incorporates a formal domain representation with semantic networks and thesauri for media- and language-specific realizations of formal concepts. In contrast to multimedia generation systems, Agile only requires partial formal representations. Moreover, for some application domains like human anatomy, the number of domain concepts is rather limited.

Semantic associations between domain objects can be inferred without an in-depth analysis of the syntactic or semantic structure of the text. To implement this, inference mechanisms based on graph structures were applied on the same underlying semantic network: Schlechtweg and Strothotte [21] employed degree-of-interest functions [8] while Hartmann et al. [12] used spreading activation [3]. The open mind common sense project [15] proved the wealth of these inference mechanisms on large semantic networks that haver been automatically extracted from corpora.

2.2 Information retrieval

In contrast to query languages that are designed for large structured databases (e.g., SQL), information retrieval (IR) has to cope with partial matches between terms in the query and those used within the documents in a text database. Therefore, text retrieval techniques are based on similarity measures for documents and employ different strategies to rank the retrieved results. For an efficient representation of a large corpus of documents, we employ the standard *vector space model* [19], where both the query q and the documents $d \in D$ are transformed into a vector representation d and q, respectively (see Fig. 2). Hence, IR systems can also be used to determine similarities between given documents and documents in databases. While these systems efficiently handle terms in natural language, the search for semantic concepts is still very limited.

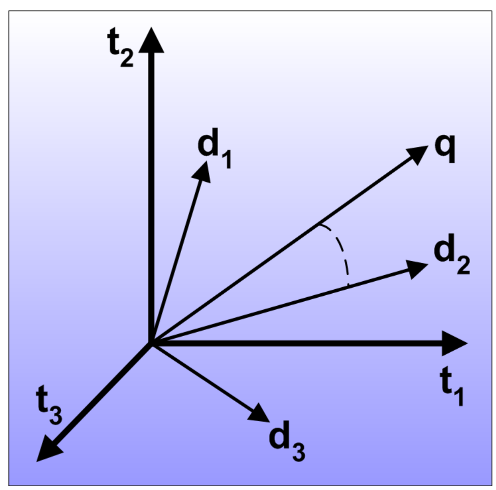


Fig. 2: According to the cosine measure the document d_2 is most similar to query q.

In order to compute vector descriptions, our system first extracts descriptors for text documents D as well as for 3D models M. In order to reduce the dimension of the vector space, we apply standard preprocessing methods: Very frequent words (*stop words*) are not considered and simple transformation rules aiming at normalizing morphological variants of words are applied to the remaining terms. Finally, by the sorted set of the remaining terms, we obtain an *index* (or *dictionary*) T_D defining a linear order over the set of terms used in a document set D. The same procedure is applied to object descriptors and textual annotations associated with geometric objects in the 3D model to construct the *index* for 3D models and their views T_M.

Then, based on the obtained dictionaries T_D and T_M we compute the document vectors for each document. The document vector d specifies weights w_t^d for all terms t in the dictionary for a given document d. In order to obtain descriptive weight vectors, we apply a standard measure [18] that considers both the frequency tf_t^d of the term t in the current document d as well as its frequency in the respective database C:

$$w_t^d = \mathrm{tf}_t^d \cdot \log(N/n_t), \tag{1}$$

where N denotes the size of the document collection C and n_t refers to the number of documents in C that contain term t. These document vectors are created for each document for the databases containing the text documents D as well as for 3D models M and, if necessary, for a given query vector q.

In contrast to the standard model, we additionally integrate a *boost function*, which enables us to increase or decrease the importance of a specific term during user interaction by modifying the document weight for a term t, where

$$\begin{array}{ll} \mathrm{boost}(t) < 1.0: & \text{de-emphasized term } t; \\ \mathrm{boost}(t) = 1.0: & \text{normal;} \\ \mathrm{boost}(t) > 1.0: & \text{emphasized term } t. \end{array} \tag{2}$$

Thus, the computation of weights as defined in Eq. 1 is changed to:

$$w_t^d = \mathrm{boost}(t) \cdot \mathrm{tf}_t^d \cdot \log(N/n_t). \tag{3}$$

The *boost function* allow us to considers the visual dominance of terms in the presentation (i.e., highlighted text or terms in the center of the visualization), as described in more detail in Sect. 4.

Based on the document vectors we are now able to search for documents that are similar to a given query q or similar to another document d', in which case $q := d'$. By comparing the query vector with each document in the collection $d \in D$ we can rank all documents according to their similarity with the query q. Here, we apply the commonly used cosine similarity [18], which is computed by the inner product of both normalized document vectors d and q:

$$\mathrm{sim}(d,q) = \frac{d \cdot q}{|d| \cdot |q|} \tag{4}$$

Similar approaches have been used in information retrieval techniques on multimedia databases, especially to retrieve bitmaps that correspond to a given query. Our approach, however, aims at analyzing a corpus of 3D models and automatically extracted views with respect to their relevance for a vector of weighted terms. Users can bidirectionally interact with explanatory texts and visualizations of 3D models. By proposing corresponding 3D views to interactively selected text or corresponding text segments to selected 3D views, the system coordinates the content of both media.

2.3 Determination of 'good' views

The determination of optimal camera parameters for a view on a 3D model given a specific task requires long-time experience and usually comprises an iterative process of trial and error. Automatic methods have been developed which use the geometrical representation of scenes in order to determine qualitative viewpoints.

Uninformed methods to determine 'good' views are purely based on topological properties of the geometric object or on evaluations of visible or occluded geometric features. Many researchers propose measures for 'good views' that aims to minimize the number of degenerated faces as seen under orthogonal projection [13] or maximize the amount of detail shown in a view [17]. Vázquez et al. [26] proposed the *viewpoint entropy*, a new measure based on Shannon's formalization of entropy [22]. The *viewpoint entropy* of a point p is defined as:

$$H_p(X) = -\sum_{i=0}^{N_f} \frac{A_i}{A_t} \log \frac{A_i}{A_t}, \tag{5}$$

where N_f refers to the number of faces of the scene, A_i denotes to the projected area of face i, A_t is the total area covered over the sphere, and A_0 represents the projected area of background in open scenes. In a closed scene, or if the point does not *see* the background, the whole sphere is covered by the projected areas and consequently

$A_0 = 0$. The maximum entropy is obtained when a certain point can *see* all the visible faces with the same relative projected area A_i/A_t.

The viewpoint entropy measure yields a good balance between the number of visible faces and the area covered by them, but can also be used if the elements' visibility we want to maximize are objects instead of polygonal faces ([27]), provided that we can identify each of the parts we are measuring.

3 Framework

We implemented an interactive application, where users can interact with elements of both media (text and graphics). These interactions are transformed into search tasks; information retrieval techniques improve the learner's facilities to explore the content of both media. Fig. 3 illustrates that several data structures are created in a pre-processing step. For a collection of descriptive texts D and of 3D models M, this comprises indices T_D, T_M and document vectors for *paragraphs* within the source document $d \in D$ as well as for *views* on 3D models (see Fig. **??** and Sect. 2.2 for further details). We use the open-source search engine LUCENE [4] to determine similarities between descriptions of views and text segments. Hence, appropriate views for the content of user-selected text segments and vice versa can be determined during runtime.

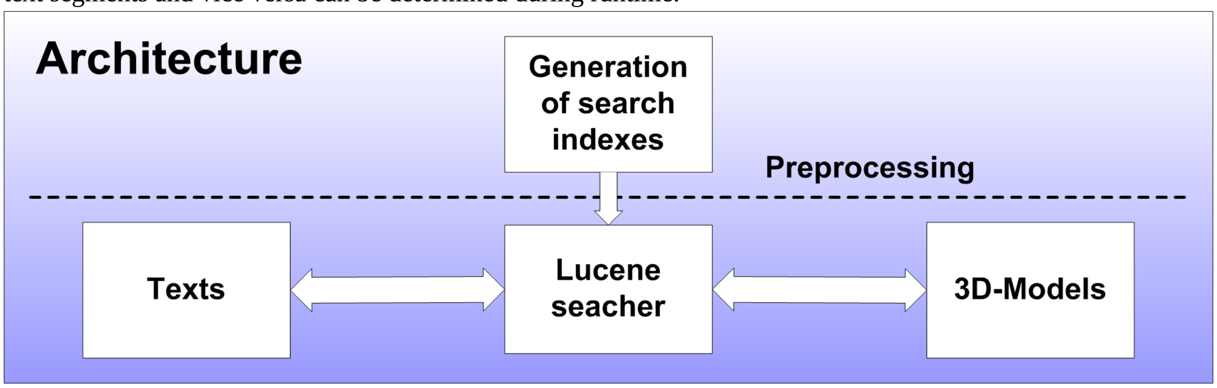


Fig. 3: The architecture of our approach.

We have chosen human anatomy as an initial application domain as (i) comprehensive corpora of tutoring material are available[4] and (ii) the comprehension of a domain-specific terminology and (iii) complicated spatial configurations are important learning tasks. In contrast to standard information retrieval systems, we employ hierarchical content descriptors (see Fig. **??**): The coarse level contains document vectors for large structures of text documents (e.g., books, chapters, and sections) or descriptors for 3D models. Moreover, the content of the coarse level is further processed to extract more fine-grained content descriptors.

4 Preprocessing of Text and 3D Models

In our approach document vectors represent the content of textual as well as visual elements. For sake of a hierarchical content description, the input data are subdivided to the desired level of granularity. Therefore, a simple parser segments text documents into paragraphs. On the graphical side, view descriptors provide more fine-grained content specifications for 3D models. Obviously, the denotations of geometric objects cannot be extracted purely from their shape. But often hierarchical geometric models (e.g., scene graphs) use technical terms to identify their constituting components. Moreover, our system allows to add or manipulate textual annotations for geometric objects directly within the 3D visualization of 3D models. This information is stored in an annotation table that links unique color codes for geometric components to technical terms. *View descriptors* specify those components, that are visible from the given point of view. In a final pre-processing step, 2 indices for the paragraph descriptors I_P

[4] Our corpus comprises an electronic version of Gray's popular textbook on human anatomy [10] and all anatomic models contained in the Viewpoint library of 3D models [6].

and the view descriptors I_V are created (see Fig. 5). Subsequently, the resulting paragraph i_p and view indices i_v can be used for search queries (see Sec. 5).

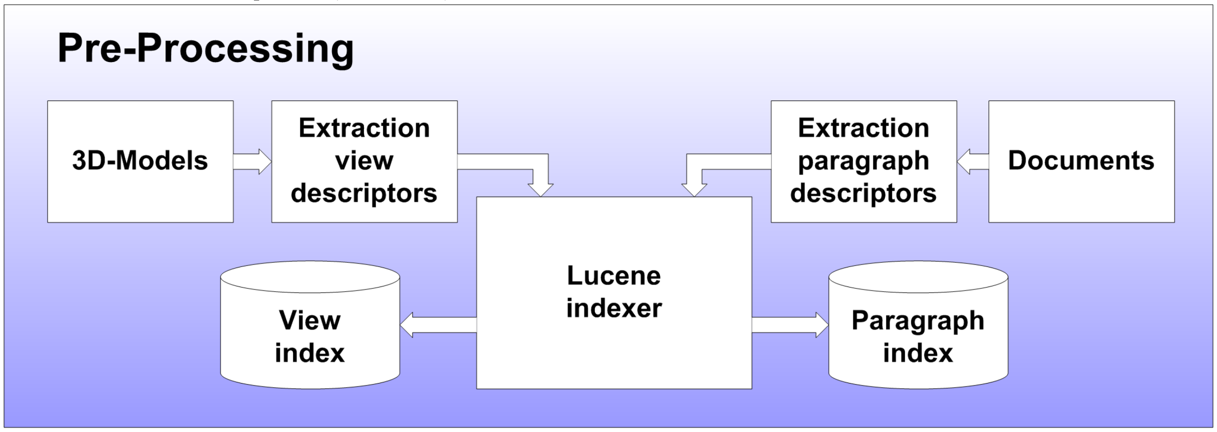


Fig. 5: Preprocessing steps.

In the following, the extraction of content descriptors for text documents and 3D models is described in dedicated sections (Sec. 4.1 and Sec. 4.2). To clarify these methods, we first propose *basic methods*, but our experimental application uses *optimized methods* with further enhancements to improve the retrieval results.

4.1 Paragraph descriptors

In well-structured documents, the layout reflects the rhetorical structure of the presentation. Especially paragraphs embrace related arguments that form a self-contained semantic context. Thus, our system uses paragraphs as basic contextual groups. Document vectors represent the content of individual paragraphs; all other levels of our hierarchical content representations can be constructed from these basic entities.

The **basic method** segments input documents $d \in D$ into paragraphs p. Subsequently, document vectors d_p for these paragraph are constructed. The **optimized method** considers additional layout information. Authors use several layout conventions to emphasize more relevant terms. Therefore, the boost function $boost(t)$ of Eq. 2 is used to assign a higher term weight to emphasized terms (see Eq. 3). Fig. 6 summarizes the process to create paragraph descriptors.

$$boost(t) = \begin{cases} 1: & e_t \equiv \text{normal}; \\ 3: & e_t \equiv \text{bold} \vee \text{italic} \vee \text{underlined}; \\ 5*\Delta: & e_t \equiv \text{changed font size (factor } \Delta); \\ 10: & e_t \equiv \text{heading}. \end{cases} \tag{6}$$

4.2 View descriptors

In our application domain, the majority of textual descriptions is dedicated to explain spatial relations between many complex-shaped objects. It is a challenging task for illustrators to select a single point of view, which presents *all* relevant objects. Moreover, characteristic object features should be visible. Hence, the visibility of the individual components from a given point of view determines their potential to complement textual descriptions.

The information of the annotation table (objectId $\mapsto$ technical terms) is used to assign unique colors to the individual components of a 3D model. Our system incorporates methods to determine visible objects by color-coded renditions into hidden buffers. We precompute descriptors for views, that are equally distributed on an orbit around the bounding sphere. Theoretically, an infinite number of views could be sampled. In order to limit the preprocessing time and the storage size, users can specify the number of samples in a range [10..10000].

In the **basic method** the weights of the terms (object descriptor) in a view descriptor reflect whether or not the corresponding geometric object is visible (see Fig. 7). In the **optimized method** the *relative size* of visible objects and their *centricity* on the projection are used as boost factors. Users can weight the influence of both factors. Thus, the boost value of each object-term is calculated by the weighted sum of both relative size and centricity. This means that objects which are (i) big and/or (ii) central are getting a higher boost value in the view's descriptor than (i) small and/or (ii) eccentric objects. Additionally, the *view entropy* can reveal how much contextual information an entire view possesses.

Relative Size and Centricity: Important objects should be optimally visible. For all geometric objects o of an 3D model we determine the maximal projection size $o_{size_{max}}$ in the set of sample view points. For a given view the *relative size* is determined as follows:

$$\text{importance}_{size} = \frac{o_{size_{view}}}{o_{size_{max}}} \tag{7}$$

Moreover, the most important objects are often presented in the center of an illustration. In order to determine the *centricity*, we first determine the objects bounding box. In this bounding box we inscribe 10 concentric ellipses which define the multiplicative factor for each pixel of the objects (see Fig. 8). Finally, we sum up the objects' score in each elliptical region.

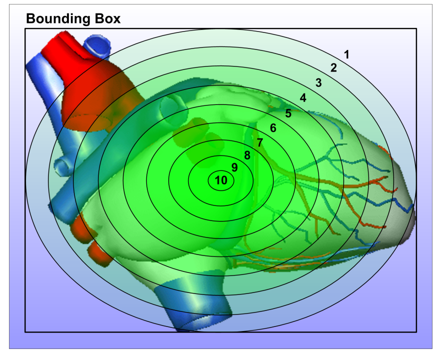


Fig. 8: Determination of the centricity of the projection of a visual object.

View's Entropy: This measure reflects the available contextual information for a given point of view. Once again, the impact of this measure on the comprehension of a views can be specified by the user. Therefore, a weight is assigned to these measures that can be adjusted with sliders during run-time. Our implementation uses a modified metric proposed by Vázquez et al. [27], where the probability distributions are the relative numbers of pixels of a certain object with respect to the size of the view, while Vázquez' original proposal [26] is based on the total solid angle subtended by the face of interest divided by the solid angle of the view.

5 Interactive Browser and Mutual Queries

Our system allows learners to explore comprehensive tutoring materials stored in multi-modal databases in an interactive browser. In order to coordinate the content presented in text and graphics, user interactions are transformed

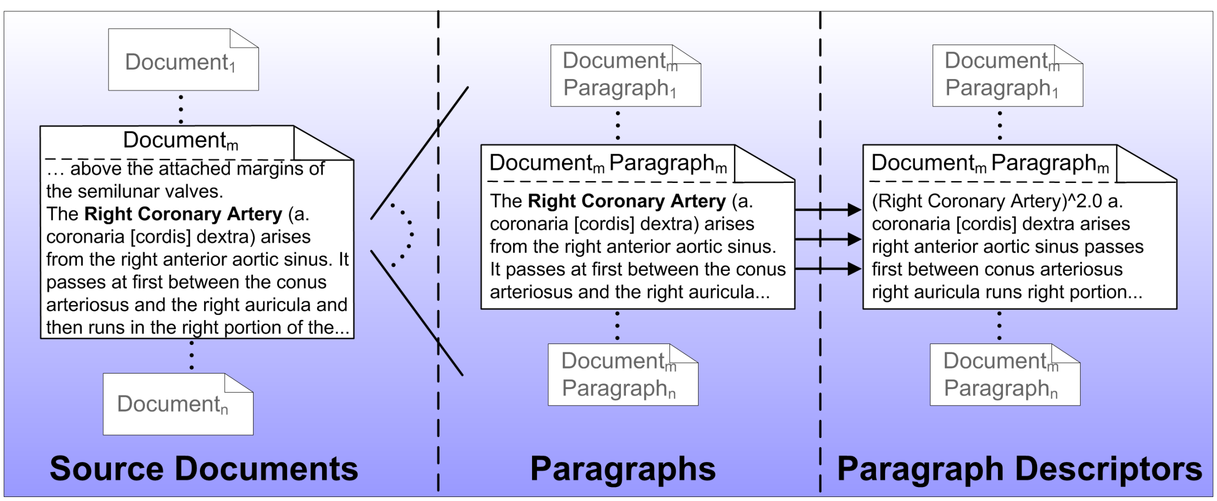


Fig. 6: The extraction of paragraph descriptors.

into queries to an information retrieval system. Two precomputed data structures enable us to find corresponding text descriptors $\leftrightarrow$ view descriptors in real-time. Subsequently, a text and a 3D browser are used to present the retrieved results (see Fig. 9).

This section highlights the need of adoptive and coordinated multi-modal presentations in our application domain. As in many scientific or technical areas, students of human anatomy have to learn a large number of technical terms. Moreover, anatomic textbooks focus on descriptions of geometric properties: chapters on *osteology*, for example, contain descriptions of characteristic features of complex-shaped bones, chapters on *myology* employ the features of these bones as landmarks to describe the course of muscles, the *syndesmology* explains the direction of movements in joints. These examples illustrate (i) the wealth of labeled illustrations to learn a domain-specific terminology, (ii) the relevance of basic learning tasks in anatomy for almost all scientific or technical domains, and (iii) the need to complement textual descriptions with expressive illustrations. The integration of a real-time label layout system and the automatic selection of appropriate 3D models and views from a database in our system aim at supporting all these learning tasks.

Text $\mapsto$ view queries: Medical students can select text segments in descriptive texts or textual labels in the 3D visualization where they need additional background information (see Fig. 12). These interactions raise the demand to search for corresponding textual descriptions and adopted illustrations.

The content of user-selected text fragments are transformed into query vectors as described in Sec. 4.1. The view index I_V allows to retrieve views on 3D models that correspond to the most relevant terms in the query. The best fitting 3D model in the best view is loaded and presented immediately; the remaining good views of other 3D models are presented in small overview windows on the side of the 3D screen; they are loaded if they are selected by the learner. To support the interactive exploration of q 3D model, the quality of views with respect to the current context defined by the query terms is presented on a colored sphere.

Moreover, the label layout within these computer-generated illustrations should reflect this new context. Hence, only relevant graphical objects are annotated. Therefore, the 3d visualization component of our system integrates a real-time layout for secondary elements that aims at minimizing the visual flow of during user interactions (see [11]). Finally, the renderer employs transparency to de-emphasize unimportant objects.

View $\mapsto$ paragraph queries: As many learners prefer visualizations to explore complex spatial configurations, our system also supports 'visual' queries. Therefore, information on visible objects, their relative size and the viewpoint entropy are extracted from the current view and used to construct a query vector as described in Sec. 4.2

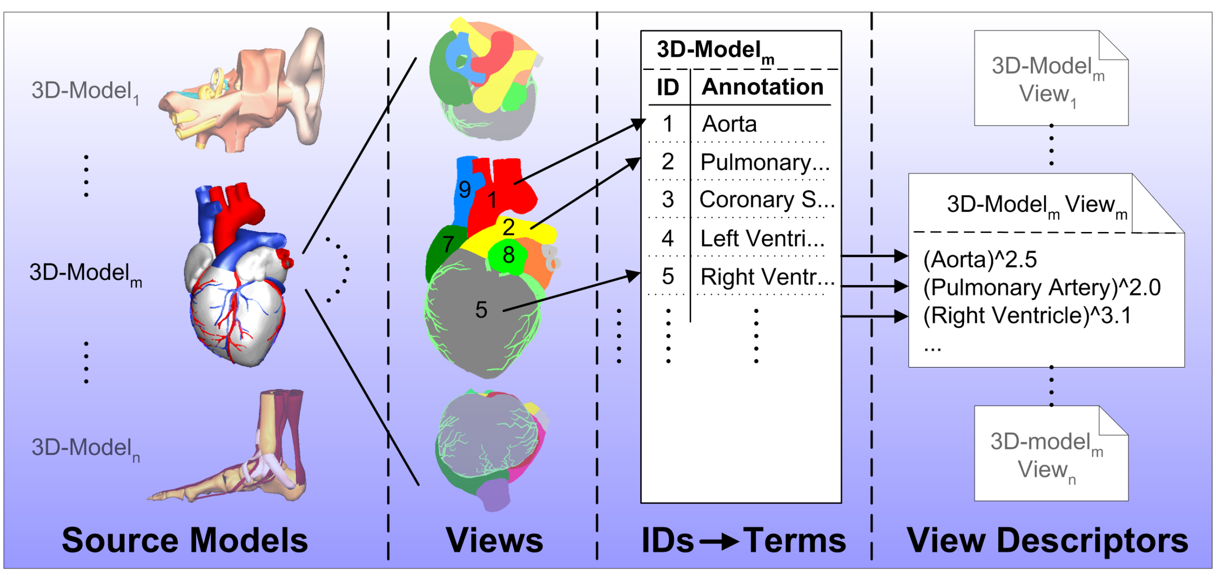


Fig. 7: The extraction of view descriptors.

(see Fig. 13). Using this view descriptor the paragraph index I_P is used to determine appropriate text segments for user defined views. The search result is sorted according to the relevancy and presented to the user. User-selected paragraphs are then presented in their original content of the chosen document.

6 Interactive Query Refinement

In this section interactive tools to refine search queries are presented. We introduce intuitive extensions for both search tasks using the techniques described in the previous sections.

Refining text $\mapsto$ view queries: As the layout is tightly coupled with the relevance of terms, we allow users to emphasize or de-emphasize text segments interactively. Text selections in combination with some control keys or with specific mouse buttons increase or decrease the font size. Moreover, additional terms can be added to the query. We believe that this interaction technique can be used intuitively. As the layout is already considered in the boosting function, this interaction directly influences the weights of the query terms and helps to improve the view retrieval results. Fig. 10 presents an example of a refinement to a text query (blue mark). Fig. 10-top contains the original paragraph as printed in the textbook, a learner used the font size to signal a major interest on some terms (i.e., conus arteriosus) and minor interest on another term (i.e., right coronary artery). The video shows how this interaction affects the search results (views).

Refining view $\mapsto$ text queries: Our system also employs (non-linear) magnifications as a visual counterpart to textual emphasis techniques. In contrast to global enlargements, this illustration technique is able to emphasize important objects while preserving contextual objects. Even though many approaches have been proposed for this task, the majority can only be applied on predefined focus objects (e.g., [2,29]). But our application requires to enlarge arbitrary regions that correspond to certain objects. General distortions [14] is a powerful technique but real-time algorithm are not available yet.

To enlarge user-selected objects we use image-based nonlinear magnifications. Our system employs the *SpringLens*-technique [9] which provides complex, interactive, nonlinear magnifications. The *SpringLens* algorithm treats images as flexible surfaces. The image is covered with a grid of particles connected with springs. By changing the rest lengths of the springs and applying a physical simulation, the grid is distorted. The particles self-organize, such that the focus regions are magnified. In this new application of the *SpringLens* algorithm we enlarge the rest length of springs that corresponds to regions in focus and distort renditions of the 3D scene. As the query construction

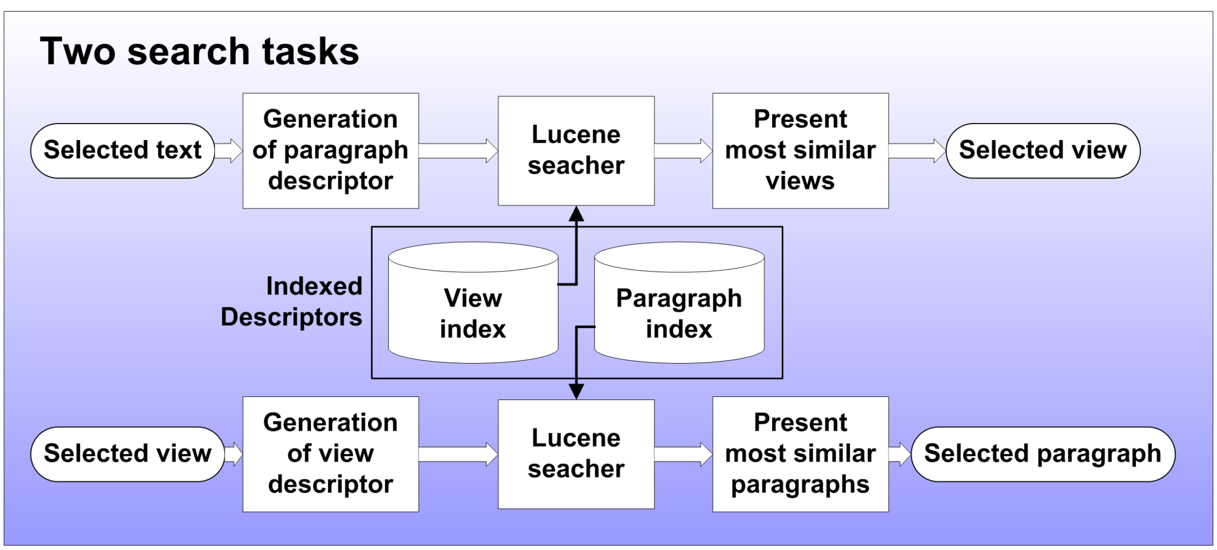


Fig. 9: Search tasks: Text↔Illustration.

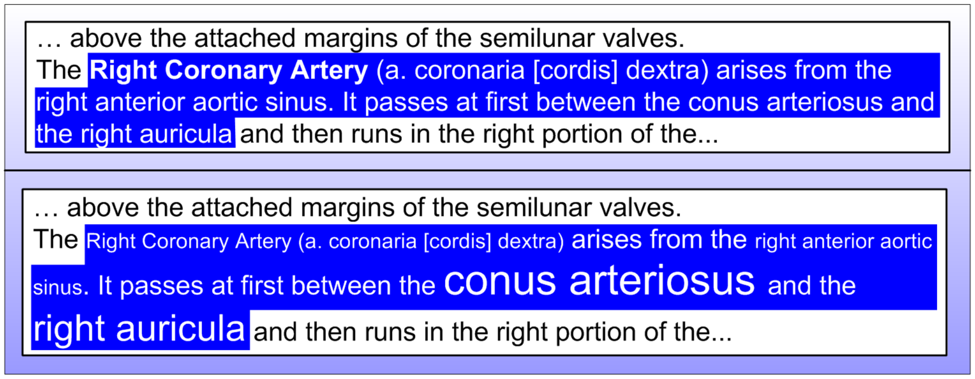


Fig. 10: Refinement of a text query.

considers the relative size of objects on the projection in the boost function (see Sec. 4.2), non-linear distortions directly alter the weights of the query terms.

Fig. 11 presents an exemplary refinement of a view query (blue mark). The left illustration shows the 3D object from a selected view with its original size. On the right side the user enlarged some objects (i.e., atrium dextrum) while others were shrinked (i.e., aorta). The search results (paragraphs) are affected by these refinements.

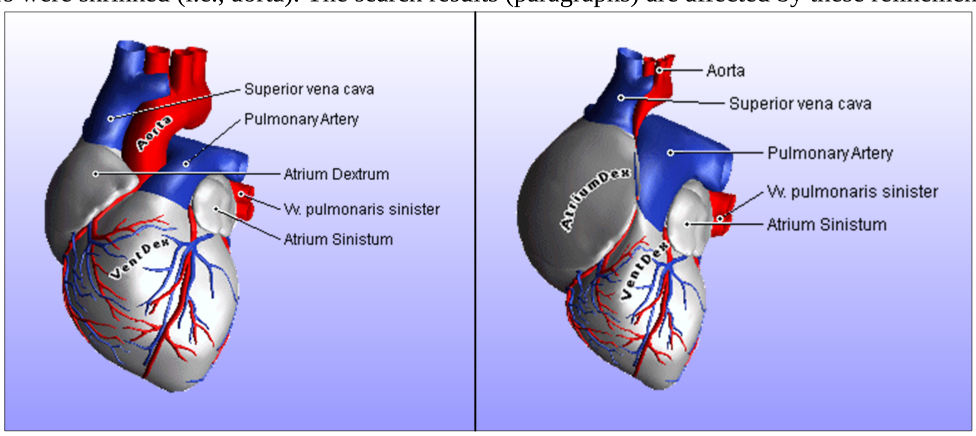


Fig. 11: Refinement of a view query.

7 Discussion and Future Work

The very efficient indexing mechanism of vector-based search engines permits the access to an almost unlimited number of documents. In our approach, we construct content descriptions for all paragraphs within voluminous tutoring documents and for a user-selected number of views onto a library of 3D models. User evaluations have to analyze whether a finer level of granularity (e.g., sentences) in the representation of the textual content or more view descriptors are required.

The rendering engine of experimental application employs semi-transparency in order to focus the attention of the user to the most salient visual objects in result of a text→image query. Moreover, a dynamic label layout is used to convey more detailed information for salient objects and to highlight their relevance. More elaborated illustration techniques such as correct transparency, cut-aways, ghost views, adjusted lighting, or NPR rendering styles were proposed for interactive 3D visualizations of surface and voxel models (e.g., [5,24]).

The presentation in this paper suggested to annotate the individual components of a 3D model with a single descriptor in natural language. While this approach is appropriate for application domains with a standardized terminology, our system already provides mechanisms to define language-specific object descriptors. This mechanism allows to integrate interactive 3D-visualizations into a multi-lingual tutoring environment based on translated documents or completely different documents for all languages. Moreover, several techniques were proposed in information retrieval to cope with the multitude of possible denotations for objects in natural language. Multi-word terms are another challenging problem for information retrieval approaches. Our current text parser segments documents into paragraphs and words. Hence, multi-word terms are not directly represented in the document descriptors, only their components. We consider two solutions: (i) to extract those multi-lingual terms from annotated 3D models prior to the text indexing or (ii) to employ text mining techniques to determine proper names and technical terms from arbitrary texts.

Our approach is highly scalable: users or domain experts can easily add new text documents or 3D models. Beside a dynamic index creation, the support of more file formats, drag&drop mechanisms, or a plug-in to import data from a web browser would improve the user interface. Medical tutoring systems would benefit from expressive renditions of volumetric datasets in addition to or in combination with renditions of polygonal models. Especially the visible human dataset [16] has been exhaustively annotated, hence this data set is very attractive for our tutoring application. As all required components have already been successfully applied to volumetric data (e.g., expressive visualizations [25]), real-time label layout [1], viewpoint entropy) the integration of them could be done nearly seamlessly.

Currently, we are performing a user study to evaluate (i) the influence of relative size and centricity to the users decision of a good view of important objects and to (ii) assess our algorithms for the selection of views to given text segments as well as the selection of text segments to given views.

References

1. S. Bruckner and E. Gröller. VolumeShop: An Interactive System for Direct Volume Illustrations. In *IEEE Visualization*, pages 671–678, 2005.

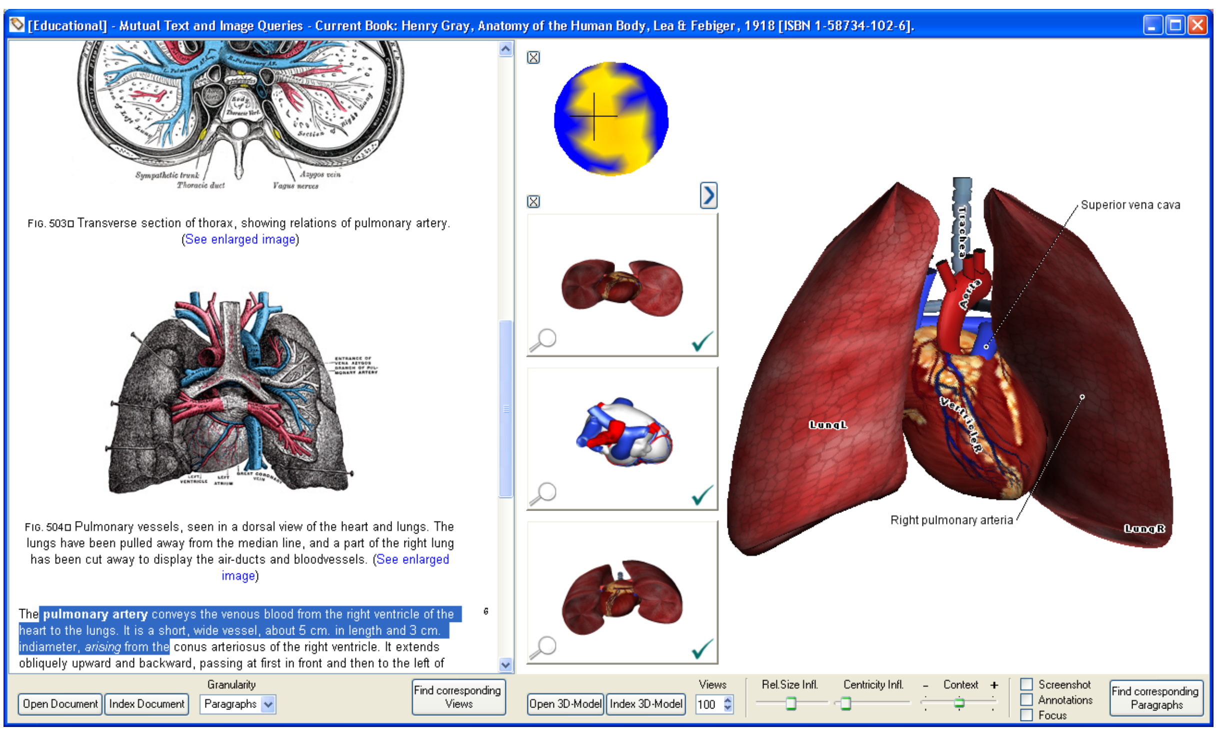


Fig. 12: A text ↦ view query (left) with it's results (right).

2. M. S. T. Carpendale and C. Montagnese. A Framework for Unifying Presentation Space. In *14th Ann. ACM Symp. on User Interface Software and Technology*, pages 61–70, 2001.
3. A. M. Collins and E. F. Loftus. A Spreading-Activation Theory of Semantic Processing. *Psychological Review*, 82(6):407–428, 1975.
4. Cutting, D. et al. Lucene text retrieval engine, 2007.
5. J. Diepstraten, D. Weiskopf, and T. Ertl. Transparency in Interactive Technical Illustrations. *Computer Graphics Forum*, 21(3):317–325, 2002.
6. Digimation Inc. Viewpoint 3D Library, 2007.
7. S. K. Feiner and K. R. McKeown. Automating the Generation of Coordinated Multimedia Explanations. In M. T. Maybury, editor, *Intelligent Multimedia Interfaces*, pages 117–138. AAAI Press, Menlo Park, 1993.
8. G. W. Furnas. Generalized Fisheye Views. In *Conf. on Human Factors in Computing Systems*, pages 16–23, 1986.
9. T. Germer, T. Götzelmann, M. Spindler, and T. Strothotte. SpringLens: Distributed Nonlinear Magnifications. In *Eurographics'06 Short Papers*, pages 123–126, 2006.
10. H. Gray. *Anatomy of the Human Body*. Lea & Febiger, Philadelphia, 20th edition, 1918.
11. T. Götzelmann, K. Hartmann, and T. Strothotte. Agents-Based Annotation of Interactive 3D Visualizations. In *6th Int. Symp. on Smart Graphics*, pages 24–35, 2006.
12. K. Hartmann and T. Strothotte. A Spreading Activation Approach to Text Illustration. In *2nd Int. Symp. on Smart Graphics*, pages 39–46, 2002.
13. T. Kamada and S. Kawai. A Simple Method for Computing General Position in Displaying Three-Dimensional Objects. *Computer Vision, Graphics, and Image Processing*, 41(1):43–56, 1988.
14. T. A. Keahey and E. L. Robertson. Nonlinear Magnification Fields. In *IEEE Symp. on Information Visualization*, page 51, 1997.
15. H. Lieberman, H. Liu, P. Singh, and B. Barry. Beating Common Sense Into Interactive Applications. *AI Magazine*, 25(4):63–76, Winter 2005.
16. National Library of Medicine. Visible Human Project, 2007.
17. D. Plemenos and M. Benayada. Intelligent Display in Scene Modeling. New Techniques to Automatically Compute Good Views. In *Int. Conf. on Computer Graphics & Vision*, 1996.
18. G. Salton, J. Allan, C. Buckley, and A. Singhal. Automatic Analysis, Theme Generation, and Summarization of Machine-Readable Texts. *Science*, 264:1421–1426, 1994.

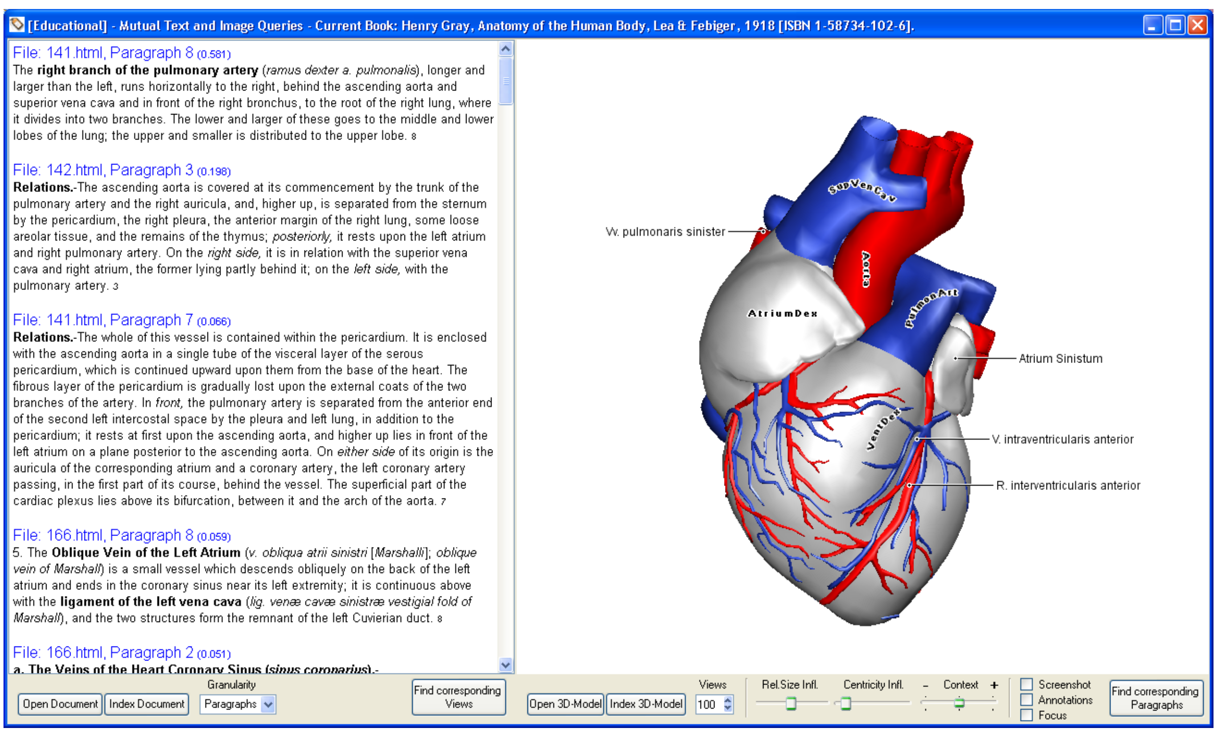


Fig. 13: A view ↦ text query (right) and it's results (left).

19. G. Salton, A. Wong, and C. S. Yang. A Vector Space Model for Automatic Indexing. *Communications of the ACM*, 18(11):613–620, 1975.
20. S. Schlechtweg and T. Strothotte. Illustrative Browsing: A New Method of Browsing in Long On-line Texts. In *Int. Conf. on Computer Human Interaction*, pages 466–473, 1999.
21. S. Schlechtweg and T. Strothotte. Generating Scientific Illustrations in Digital Books. In *AAAI Spring Symp. on Smart Graphics*, pages 8–15, 2000.
22. E. C. Shannon. A Mathematical Theory of Communication. *The Bell System Technical Journal*, 27:379–423,623–656, July-October 1948.
23. G. J. Tortora. *Introduction to the Human Body: The Essentials of Anatomy and Psychology*. Benjamin Cummings, Menlo Park, CA, 1997.
24. I. Viola, M. Feixas, M. Sbert, and M. E. Gröller. Importance-Driven Focus of Attention. *IEEE Transactions on Visualization and Computer Graphics*, 12(5), 2006.
25. I. Viola, A. Kanitsar, and M. E. Gröller. Importance-Driven Volume Rendering. In *IEEE Visualization 2004*, pages 139–145, 2004.
26. P.-P. Vázquez, M. Feixas, M. Sbert, and W. Heidrich. Viewpoint Selection using Viewpoint Entropy. In *Vision Modeling and Visualization Conference*, pages 273–280, 2001.
27. P. P. Vázquez, M. Feixas, M. Sbert, and A. Llobet. Realtime Automatic Selection of Good Molecular Views. *Computers & Graphics*, 30(1):98–110, 2006.
28. W. Wahlster, E. André, W. Finkler, H.-J. Profitlich, and T. Rist. Plan-Based Integration of Natural Language and Graphics Generation. *Artificial Intelligence*, 63:387–427, 1993.
29. Y. Yang, J. X. Chen, and M. Beheshti. Nonlinear Perspective Projections and Magic Lenses: 3D View Deformation. *Computer Graphics & Applications*, 25(1):76–84, 2005.