CONSISTENCY OF RENDERED IMAGES AND THEIR TEXTUAL LABELS

Bernhard Preim, **Alf Ritter**, **Thomas Strothotte**
Department of Simulation and Graphics, Otto-von-Guericke University of Magdeburg
Universitätsplatz 2, D-39106 Magdeburg, GERMANY

e-mail: {bernhard, alf, tstr}@isg.cs.uni-magdeburg.de

Tilo Pohle
Department of Surgery, Otto-von-Guericke University of Magdeburg
Leipziger Str. 44, D-39118 Magdeburg, GERMANY

Lyn Bartram[1]
Center for Systems Science
Simon Fraser University,
Burnaby, BC, V5A 1S6, CANADA
lyn@cs.sfu.ca

David R. Forsey[1]
Department of Computer Science
University of British Columbia
Vancouver, BC, V6T 1Z2, CANADA
drforsey@cs.ubc.ca

ABSTRACT

We present algorithms and their implementation in the ZOOM ILLUSTRATOR which enable textual labels to “float” around rendered images to which they refer. The labeling remains consistent, even when the image is rotated or scaled. Furthermore, the level of detail of the labeling adjusts to the screen space available. This is accomplished using fisheye techniques on the labels. We describe a prototypical implementation using the Open Inventor™ and C++ on Silicon Graphics Workstations.

Keywords

Interactive illustrations, Relation between images and textual labels, fisheye techniques

1 INTRODUCTION

Illustrations play an important role in teaching materials. They often include both images and textual information like labels. The relationship between these two basic components determines to a large degree whether a student can extract the depicted content. Illustrations with a careful combination of images and texts are extremely valuable in illustrating the construction of complex things.

Learning with traditional static media, however, involves a lot of flipping through the pages, because illlustrations are often not at the same page as the describing text. This is especially true for anatomical illustrations. Using traditional textbooks (e.g. [Sobotta 88]) to study a certain part of the human body, numerous illustrations with plenty of labels and long explanations have to be compared with one another. This is due to the static character of a book, which makes several viewing directions necessary, to give an idea of the 3D-situation and due to different aspects (blood vessels, nerves, muscles), which cannot be covered in one image. This is just the situation in which interactive learning can help: According to [van Dam 91] we refer to interactive illustrations “when the student is able to look at the object with the level of detail he or she wants, completely customized. This means that ... these illustrations are derived from stored models in real-time”.

At present the emphasis of the research literature is placed, on the one hand on the construction and visualization of complex models, or on the other hand on structuring text in hypertext sy-

[1]This work was carried out in part while the authors were visiting scientists at the Univerity of Magdeburg

stems in connection with scanned images. Our goal is to develop a flexible system to handle interactive illustrations with rendered images and the corresponding textual labels and explanations. With this system the user is able to interact with both the image and the text, while the system is responsible for the *consistency* between these media in the presentation.

To achieve this consistency, several rules are established which refer to how textual information should relate to an image. For ease of viewing, our system aims at continuous transitions between complex views. It turns out that there is a trade-off between this consistency on the one hand and the continuous transitions on the other. The principles are tested for anatomical illustrations, an interesting area of application for our system, because complex issues are involved which can by no means be explained without images.

The paper is organized as follows: After a short review of related work the design of the ZOOM ILLUSTRATOR is described in Chapter 3. Emphasis is put on a close relation between images and text. The components of the system and their interplay are described in Chapter 4. Interaction techniques are explained by means of several examples in Chapter 5. Because the tool described here aims at interactivity on the one hand but involves complex transformations on the other hand, compromises between the quality of illustrations and the rendering times are necessary. These compromises and their influence on performance are described in Chapter 6. The concluding Chapter argues that the techniques presented can be easily applied to other applications which illustrate complex 3D-phenomena.

2 RELATED WORK

A look at previous work on interactive anatomy teaching reveals systems based on scanned images and those using 3D-models. The latter are more advanced because they allow more flexible transformations of the subject being studied.

One of the leading examples in this field is the VoxelMan™ (see [Pommert *et al.* 94]). This system employs volume models, constructed from image-producing approaches (e.g. CT-images) of patient-specific data. The use of volume models offers free positioning and, moreover, the facility to interactively cut off, take away parts i.e., literally to operate virtually. Although the images constructed this way are labelled and optionally explained, the coherence of pictorial and textual information, the correspondence between these basic components, however, is handled with fixed data-structures and not treated in its own right.

While anatomy teaching systems are related to our work in terms of the application area, systems generating illustrated documents are important in terms of the underlying principles and the relation between rendered images and textual information. Several systems exist which consider communicative intents to generate an integrated illustration with images and text to express them. These systems use knowledge-based techniques, like planning strategies, to present a subject from the „right“ position and combine images and texts. Advanced techniques for the automated design of such illustrations were developed by [Seligmann-Feiner 91], [Rist-André 92] and [Wahlster *et al.* 93]. The IBIS-system, described in [Seligmann-Feiner 91], also includes an interactive component which maintains the communicative intents which guided the generation of the original illustration. In contrast to these systems, the focus of the ZOOM ILLUSTRATOR lies on the flexibility provided to create a specific view. While the strategies to generate an initial layout are relatively straightforward, our goal is to develop more sophisticated techniques to support the placement of textual items after user-interaction.

3 DESIGN OF THE ZOOM ILLUSTRATOR

Our design is guided by rules for consistency between images and textual information. We derived these rules by an analysis of traditional teaching materials (e.g. [Sobotta 88]) and by following explicit hints of a book which describes how to make good medical illustrations ([Briscoe 90]). These rules, which are given on the next page, guide the creation of an initial illustration as well as the placement of information after user interaction.

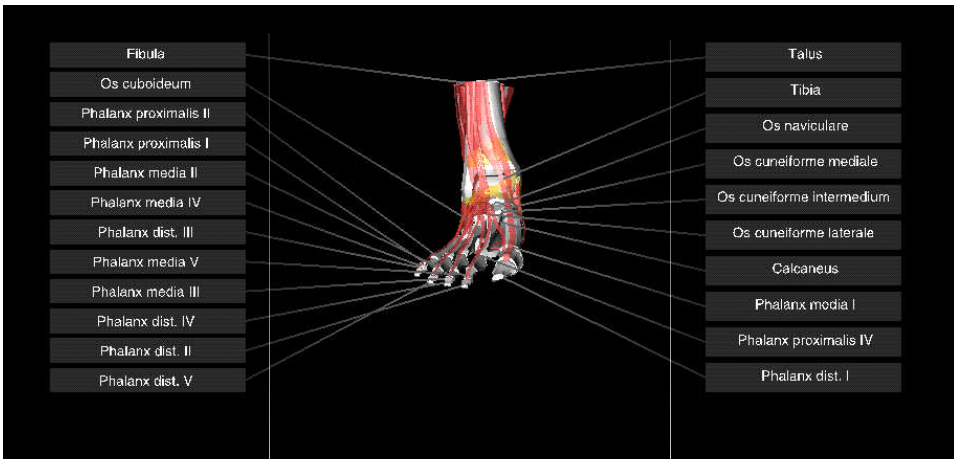

Figure 1: Example output of a foot with additional lines to show the separation of regions for the placement of textual information. Emphasis is put on the bones (white).

Rule 1. Labels are connected by reference lines to the objects in an image to which they refer.

Rule 2. Objects which are currently explained are coloured to be recognized easily.

Rule 3. Labels appear only for visible objects.

Rule 4. The placement of labels should be coherent with the position of reference points in an image so as to avoid long lines crossing each other.

Rule 5. The size of labels should vary according to the depth of the referred object in the image

While these rules sound simple enough, strictly applying them actually results in radical (and therefore irritating) changes of the presentation. We will explain this phenomen later on in the paper and show how to solve the problems which arise.

Textual information is embedded in rectangular areas. The placement of these rectangles is based on fisheye techniques, as introduced in [Furnas 86]. By contrast to other systems which automatically place entities (see e.g. [Forman 92]), fisheye techniques are useful to zoom interactively in and out. The placement and the size of information is based on the *degree of interest (DOI)*, an application-specific value influenced by user interactions. Continuously enlarging entities at the expense of others, the size of which is reduced accordingly, is accomplished by a variant of the continuous-zoom algorithm as presented in [Dill *et al.* 94], (see also [Bartram *et al.* 94]). Whenever a rectangle gets large enough to accommodate more text, an additional explanation is displayed which is subsequently hidden or made smaller automatically if other rectangles are enlarged. In accordance to the terminology of [Dill *et al.* 94], we refer to the rectangles and the content they represent as *nodes*. Nodes can be scaled continuosly. Depending on their size they are opened (presented) and closed (hidden).

To ensure that the image is not occluded by textual information and vice versa, the window is subdivided in a central part for the image (with 50%), as well as a left and a right part for textual descriptions, which occupy 25% each of the screen space available (see Figure 1).

The continuous zoom works independently in the left and right area to prevent irritating changes in one part due to interaction in the other. To emphasize the importance of this zoom-technique, we refer to our system as the ZOOM ILLUSTRATOR.

4 ARCHITECTURE

The generation of an illustration is based on three input-sources (see the top 3 vertical boxes in Figure 2):

The first is a *scene description*, containing a polygonal 3D-model which is structured into objects. To ensure high-quality images, we rely on commercially available models.

Secondly, we employ a file with related *textual descriptions* referring to the objects in the scene description. To allow flexibility in the presentation, the information space of textual descriptions is segmented into labels, short explanations and extended explanations. Each node can be zoomed up to accomodate a short explanation. This explanation contains links to extended explanations which will be displayed if the user clicks on them. Furthermore, each object has a reference to the organ system it belongs to. This allows to create illustrations with the focus on certain organ systems.

Finally, we use a file with *precalculated visibility information*. This file contains information on which objects are visible for discrete viewing directions. As we will see, this information is very useful for the interaction because it is time-consuming to derive it later.

After the sources are loaded, an internal representation is generated using the correspondence between the *object-id* in the 3D-model, in the textual description and the visibility information. This step is followed by the choice of which organ-system(s) is (are) important. Such a selection is very useful because the whole amount of information for one part of the human body is overwhelming and can not be presented in its entirety in one view.

Based on this information, an initial illustration is created which includes labels, reference lines and the rendered image. This task is accomplished by the *Layout Manager*. The decision which objects to label is a two-step process.

The first step serves to find out whether a conflict between the number of relevant objects (relevant in terms of organ systems selected and visibility) and the screen space available to label them exists. To do this, it is counted how many objects are relevant. Depending on this number the font size for the labels is determined. If the number of relevant objects is over a threshold which is oriented on the number of labels which can be displayed with legible font, a conflict has arisen. In this case a second step is necessary to reduce the number of objects to label to solve this conflict.

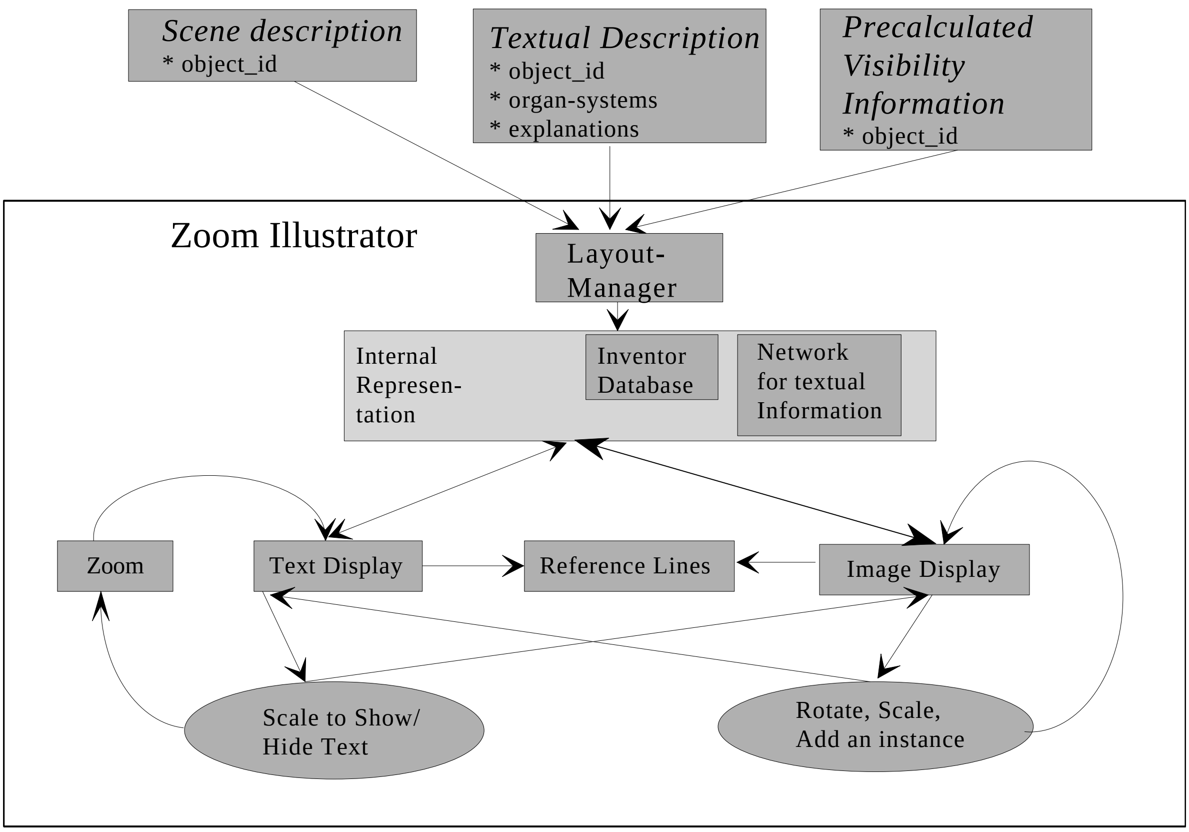

Figure 2: Architecture of the ZOOM ILLUSTRATOR

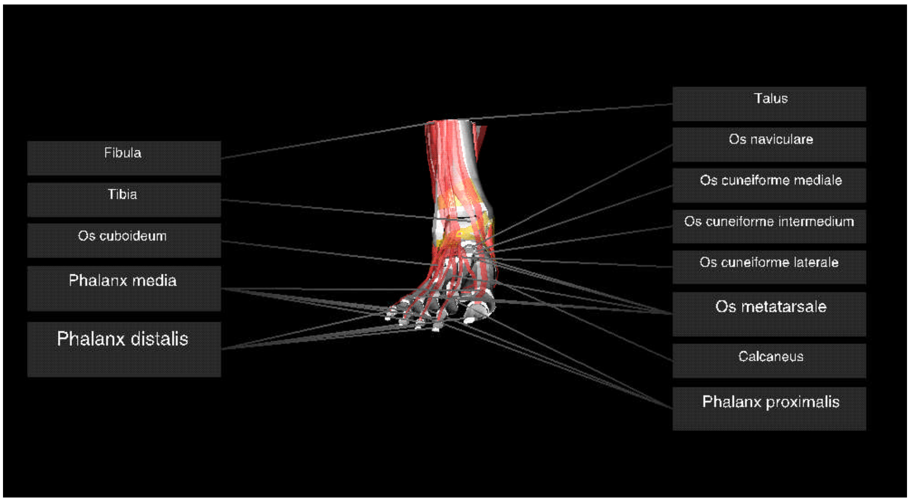

Figure 3: Bones of the right foot. Figure 3 is very similar to Figure 1 expect that the threshold for the number of labels was decreased, so that parent nodes are labelled with several reference lines. These parent nodes appear larger than others (see the bottom labels on the left and on the right side).

Figure 4: Explanation of a muscle which is automatically arranged within the columnspace available. The rectangle is continuously enlarged to accomodate an explanation. The very last words in the explanation, which differ in their colour from the others, symbolize links to additional explanations

The second step involves the investigation of hierarchical relations among the nodes in the textual description. These relations are noted in the *parent attribute* of each node. Hierarchical relations (which are put manually into the model) exist, if one label can summarize others. If this is the case only the parent nodes are considered. This can reduce the number of nodes considerably and solve the conflict. If there are still too many nodes, all relevant nodes are sorted according to their size and distance and the first nodes are labelled.

If parent nodes are considered instead of their children, they collect the *object_id*, which represents the connection to the scene description, of all their children. This has the effect that the label of the parent node is connected via reference lines with *all* reference objects of their children (see Figure 3 for such a layout). This strategy again is very common in reference books. However, the placement of the lines within reference books is much more subtle than in the described implementation.

To support spatial understanding, depth-cueing on the reference lines is implemented. They get brighter from the start point (at the label) to the reference object in the image. The brightness of the endpoint depends on its depth-value. The colour between start- and endpoint is interpolated automatically.

The arrangement of labels in the initial illustration is according to their position in an image. Labels are equispaced arranged on the left and on the right. Objects not belonging to the organ-systems selected are made transparent to provide contextual information without distracting from the parts of interest.

5 INTERACTION FACILITIES

The interaction facilities offered (recall Figure 2, which shows the relations) include navigation through textual information and transformation of an image. The reference lines which connect textual information with the image cannot be altered directly, but are updated automatically according to the position of the objects and textual information (according to Rule 1).

Interacting with textual information is accomplished by scaling a rectangle which accomodates textual information. The zoom component is invoked to redistribute the place for all rectangles of this network. The continuous-zoom algorithm is adapted to take into account the legibility of textual information (as minimum size) and the space needed to display explanations (as maximum size). Figure 4 presents an example of a node which is continuously zoomed up to accomodate a short explanation.

Interaction with the textual information causes that objects which are currently explained are automatically coloured to be recognizable (corresponding Rule 2). The colour of the explained object depends on its original colour which will become more saturated when it is explained. The number of explanations displayed at a certain point of time is usually one or two, simply because the space available allows no more explanations to be displayed. Therefore explained objects can be coloured without confusing the whole presentation.

The zoom algorithm influences the size of the rectangles. While this is the main feature of the zoom it is questionable whether additional modifications of the presentation are useful to emphasize the nodes which belong to the rectangles which are zoomed-up. Tests were carried out to discuss whether it is helpful to adapt the size of the font, the colour of the rectangle and the thickness of reference lines in dependence of the the size of the rectangles.

The idea behind a colour change is to increase the contrast between the rectangles colour to the background colour if the object is zoomed up.

Reference lines, which are originally very thin, become a little bit thicker if the corresponding label is zoomed up to make the corrspondence between the textual information and the reference object clear. This technique, however, is not applied when a label is zoomed up which has several reference lines because this would hopelessly overcrowd the image.

Because the value of these adaptions turns out to be controversial, these features can be turned on and off interactively. If they are turned on they influence the presentation after zooming a label and after rotating the model (in the case of a rotation the rectangles size is adapted to the depth values of the reference objects). Figure 5 shows an example with all features turned on.

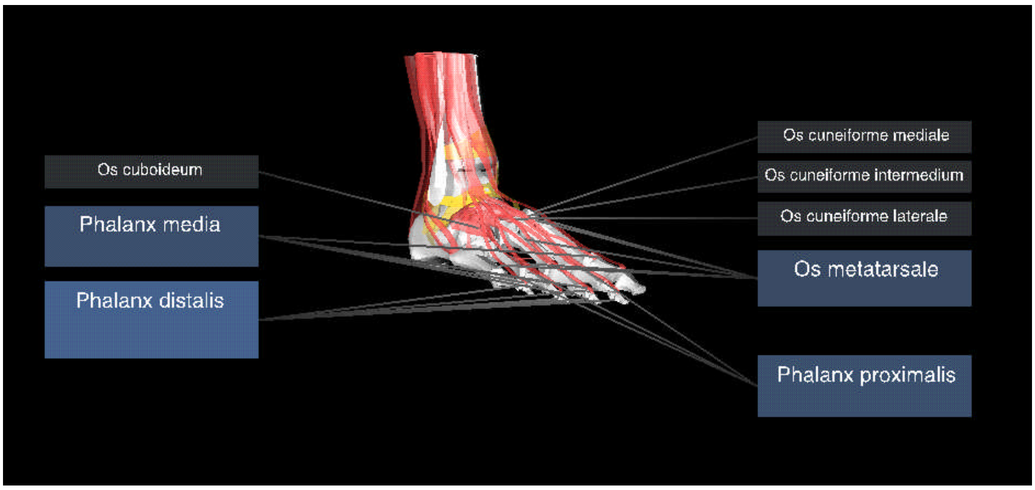

Figure 5: Colour, fontsize and line width are adapted to the size of the rectangles due to differing depth values after a rotation

The correspondence between geometric transformations of the image and textual information with respect to the above-mentioned consistency rules is complex. While scaling leaves the relation between labels and image intact insofar as the same objects are visible and the relation of their sizes remains the same, rotation changes the visibility of objects. Experiments have shown that strictly applying Rule 3 is neither easy to achieve in real-time nor useful for the student, because small rotations could lead to dramatic changes of the labels. Instead, precalculated visibility is employed to update the labels displayed only after major changes.

Another problem concerning rotation is that reference lines, which were originally well-aligned, may cross each other after a rotation (violation of Rule 4). Furthermore, very long lines arise due to rotation if objects move from left to right while their labels remain on the same position. These effects are illustrated in Figure 6. Crossing lines are annoying when carefully studying the subject. On the other hand, rearranging labels in real-time would result in a permanent movement of both labels and reference lines, which is highly irritating. To provide both smooth transitions on the fly and well-balanced displays for carefully studying, rearranging labels to prevent crossing lines is initiated by a user only.

Whether the maintainance of Rule 5 (the adaption of the size of labels to the depth-value of the referred objects) is helpful or not depends to a large extent on the number of labels. While it allows an easy comparison of the positions of a few objects, it is irritating for several dozens of labels, the position of which cannot be recognized at a glance. To cope with this a threshold-value for the number of labels is taken into account.

Figure 6a: Reference lines which were well-aligned (on the left) cross each other after a rotation by 180° (on the right)

Figure 6b: While the reference are near the reference points in the left image very long lines arise after rotation by 180°

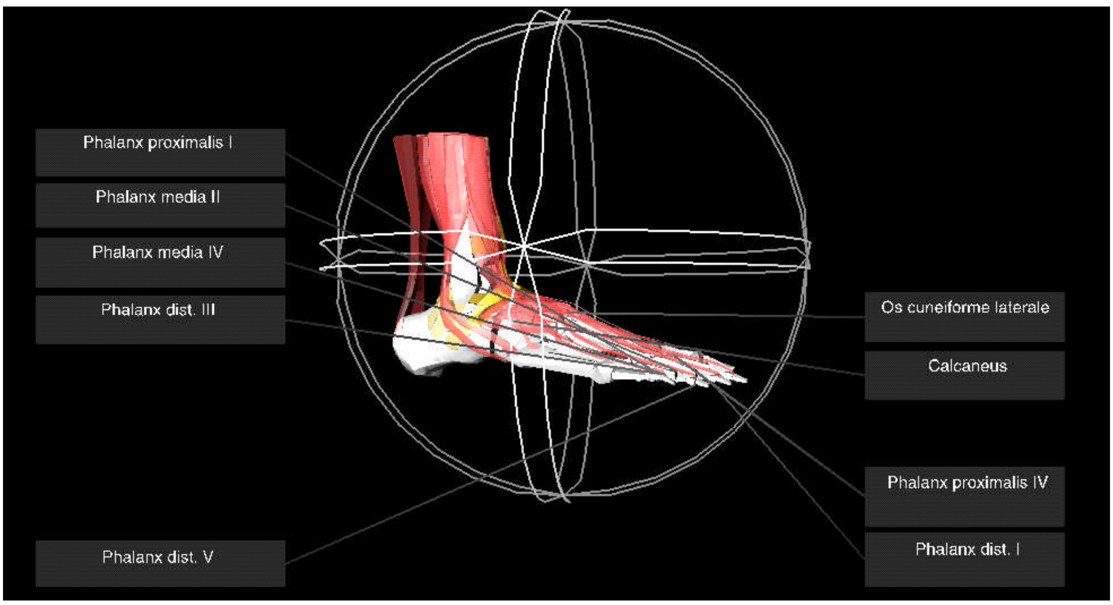

Figure 7: Bones of the foot after rotation with the virtual trackball. Due to the rotation very long reference lines from the labels on the right side to reference object in the left part of the image arise.

Besides allowing a user to manipulate one single image, a second image of the same subject can be added and transformed independently of the first. To present a subject from two different views is a common technique in anatomical atlases and very useful to develop a mental model of the spatial relations. In the ZOOM ILLUSTRATOR the images are placed beside each other with labels on the left side of the left image and on the right side of the right image. Those organ-systems which are not in the focus are drawn even more transparent (65%) than in the case of one image, because the total amount of information has increased. This way a concentration on the parts of interest is supported. Figure 8 shows an illustration created this way.

In the case of two views of the same object, presented at the same time the rearrangement of labels after rotation involves the determination which objects are visible in both images. The labels referring to these objects are placed between the images with reference lines to both to support the student in mentally integrating the two model views. See Figure 9 for an example of this technique. When a second instance of the geometric model is created and one object is visible in both instances it must be decided which of the both depth-values is judged to modify the font size of the label according to Rule 5. In the current implementation, the reference object which is nearest to the observer determines the size of the labels. With this decision an object which is visible in both images is labelled large, and thereby emphasized, if it is near to the observer in at least one of the instances of the geometric model. This seems to be reasonable because those objects which are visible in both images should have a higher priority than those parts which are only visible in one image.

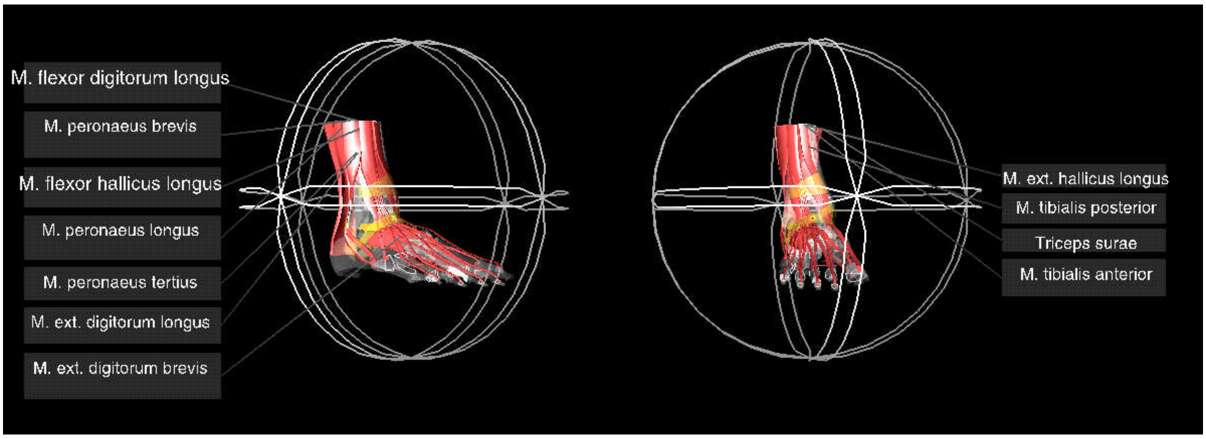

Figure 8: The muscles (red) of the bone from two different views. Bones (white) and sinews (yellow) are strongly transparent.

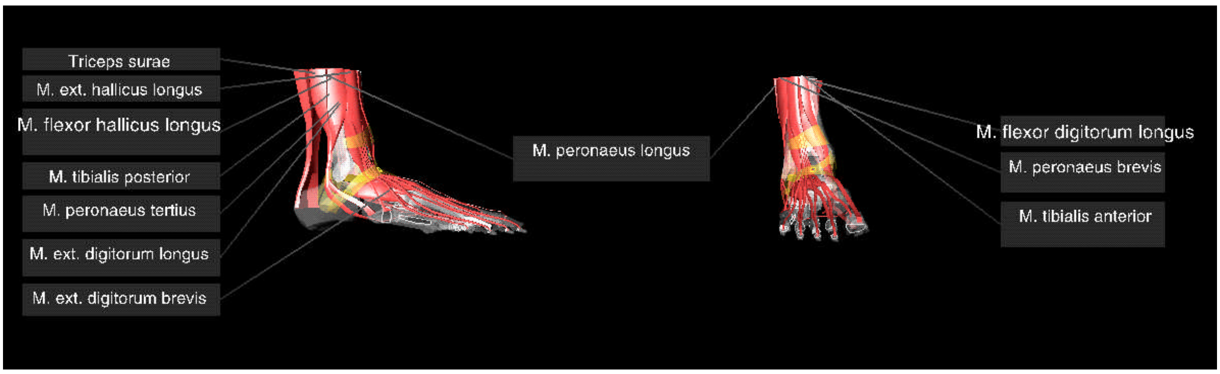

Figure 9: After the rotation from Figure 8 rearrangement of labels is activated. One muscle which is visible in both images and near to the observer is labelled between the images

6 IMPLEMENTATION

The ZOOM ILLUSTRATOR was implemented with Open Inventor™ on Silicon Graphics Indigo² Workstations with Extreme Graphics. This platform allows transforming 3D-objects of medium complexity in near real-time. Open Inventor™ provides powerful interaction facilities and an elegant way to manipulate the scene-graph, its internal scene-description. However, these interaction facilities are not for free, but time-expensive. Because interactivity is important for the system, compromises between high-quality rendering and rendering times are necessary.

To get reliable results on the performance of the system and the consequences of render-options on the performance, a series of measurements was made with *ivPerf*, a programm to monitor the performance of inventor-based applications. The model of a foot (with 6000 polygons) is rendered with a 6.5 frames per second using the low quality mode for rendering transparent objects and without antialising. The frame-rate is crucial when rotating the 3D-model via a trackball. In this case the model is rendered with a low-cost mode for rendering transparency and with antialising switched off. The frame rate drops considerably to update the presentation after one step of rotation. This is due to intensive manipulation of the scene-graph. Depending on the display options the frame-rate is between 3.5 and 2.8 (measurements are taken for a network with 14 labels), which is not convincing.

While the frame-rate is highly important in the case of rotation it is regarded as less important when the subject is not in the flow. In this case the best-quality mode to render transparent objects is used. This involves sorting all objects based on their distance to the observer and renders opaque objects before transparent objects from the very distant ones to the closest. To recognize what is behind a transparent object, a high quality transparency mode is crucial.

Even after sorting the scene-graph according to the material-definitions this transparency-mode decreases the frame-rate by a factor between 2 and 3 (depending on how many objects are transparent). Although the visual improvement of antialised images over not antialised is considerable antialising is turned off, because it drops the frame-rate by factor 5. This is due to the fact that rendering antialised images with the accumulation buffer is not supported from the graphics hardware employed.

Text presentation is expensive, because 3D-text is employed. This is necessary to have high-quality output after scaling, which can not be attained with 2D-text. The time to render text is reduced in rendering the front-side only and using a font without serifes. In spite of this it takes nearly one second to display an explanation with 30 words. An improvement of this can only be reached by rendering text with lower complexity, resulting in less convincing output, which interferes with the need for legibility.

Once the model and its description have been loaded, nearly all the interaction is mouse-based. To validate the decisions on the realization of a correspondence between the rendered image and its textual description, we tested our system with two medical students and some colleagues.

7 CONCLUSIONS

We presented the ZOOM ILLUSTRATOR, a system to illustrate complex models and investigated the relation between the rendered image and textual information referring to it. While it was relatively easy to derive rules for a consistent static presentation, it turns out that conflicts between these rules and the temporal coherence of succesive images arise. The investigations showed that these conflicts must be tackled carefully with dedicated strategies which differ in the amount of user-involvement.

While the straightforward mechanisms which guide the generation of an initial illustration are sufficient, more subtle presentation techniques are necessary to combine powerful interaction with an automatic placement of entities. To accomplish this, fisheye techniques are employed. They allow to display everything in one single window. This is by contrast to a lot of systems allowing to navigate in a large information space for the price of organizing lots of windows on the screen.

Although tested exclusively for anatomical examples, the strategies presented should be easily generalizable for illustrations of complex 3D-phenomena, e.g. in biology or even in engineering. To transform the system to another area of application, a careful segmentation of the information space is necessary.

The navigation through textual information is accomplished using a slightly modified variant of the continuous zoom algorithm (recall [Dill *et al.* 94]). This technique turns out to be very helpful to increase the temporal coherence between successive illustrations.

Future work will concentrate on techniques to improve the clarity of the illustrations generated, especially on providing cues to improve spatial perception of the rendered images. The adaption of the rendered image according to the objects textually explained should be improved by increasing the contrast between the objects within the focus and its environment.

Last but not least a usability study which extends the informal tests with some colleagues and two students is necessary. This study should help to find out whether the visualization techniques are adequate and flexible enough for the purpose at hand. A lot of assumptions have been made e.g. about the appropriateness of the effect of transparency, which must be validated thouroughly.

Acknowledgements

The authors want to thank all persons who have contributed to the presented paper. Our colleagues Steffi Fritz, Ralf Helbing, Axel Hoppe, Andreas Raab and Michael Rüger as well as the medical students Kristina Botero and Uta Hahn gave valuable hints for the design of the system. Uta Hahn helped us to label the anatomic model and to assign meaningful explanations. We are endepted to Ute Lau and Kathrin Weese, who proof-read the manuscript.

REFERENCES

Bartram, L., Ovans, R., Dill, J., Dyck, M., Ho, A., Havens, W. S. (1994)
Contextual Assistance in User Interfaces to Complex, Time-Critical Systems: The Intelligent Zoom, *Proc. Graphics Interface,* pp. 216-224

Briscoe, M. H. (1990)
A Research' Guide to Scientific and Medical Illustrations, Springer Verlag New York

Dill, J., Bartram, L., Ho, A., Henigmann, F. (1994)
A Continuously Variable Zoom for Navigating Large Hierarchical Networks, *Proc. IEEE Conference on Systems, Man and Cybernetics,* pp. 386-390

Forman, M. (1992)
Algorithms for Geometric Packing and Scaling Problems, PhD Thesis, Faculty of Mathematics, Free University of Berlin

Furnas, G. W. (1986)
Generalized Fisheye Views, *Proc. ACM* SIGCHI'86 Conference on Human Factors in *Computing Systems,* pp. 16-23

Pommert, A., Pflesser, B., Riemer, M., Schiemann, Th., Schubert, R., Tiede, U., Höhne, K. H. (1994)
Advances in Medical Volume Visualization, *Eurographics' 94, State of the Art Reports, Eurographics Technical Report ISSN 1017-4656,* pp. 111-139

Rist, T., André, E. (1992)
Incorporating Graphics Design and Realization into the Multimodal Presentation System WIP, *Proc. Advanced Visual Interfaces' 92*

Seligmann, D., Feiner, S. K. (1991)
Automated generation of intent-based 3D-illustrations, *Computer Graphics 25(4),* pp. 123-132

Sobotta, J. (1988)
Atlas der Anatomie des Menschen, 19. Edition, J. Staubesand (Ed.), Urban & Schwarzenberg, Munich-Vienna-Baltimore

van Dam, A. (1991)
Electronic Books and Interactive Illustrations – Transcript of a Talk, *S. Cunningham, R. Hubbold (Eds), Interactive Learning Through Visualization*, Springer Verlag, Berlin-Heidelberg-New York, pp. 10-24

Wahlster, W., André, E., Finkler, W., Profitlich, H.-J., Rist, Th. (1993)
Plan-Based Integration of Natural Language and Graphics Generation, *Research Report RR-93-02*, German Research Center for Artificial Intelligence