Illustrating Anatomic Models – A Semi-Interactive Approach

Bernhard Preim, Alf Ritter, Thomas Strothotte

Institut für Simulation und Graphik, Otto-von-Guericke-Universität Magdeburg
Universitätsplatz 2, D-39106 Magdeburg
e-mail: {bernhard, alf, tstr}@isg.cs.uni-magdeburg.de

Abstract: We present the ZOOM ILLUSTRATOR which illustrates complex 3D-models for teaching anatomy. Our system is a semi-interactive tool which combines an animation mode and an interactive mode. While the animation mode is suitable for beginners, the interactive mode is dedicated to experienced users. The animation is controlled by scripts specifying *what* should be explained in *which level of detail.*

The design of the animation and the interactive components is directed to generate illustrations according to the user's interest. This includes the presentation of text and the parameterization of the rendered image.

Changes on the textual part, interactively requested or generated in an animation, are propagated to the graphics part and vice versa. Thus the display of an explanation results in an adaptation of the corresponding graphical part.

Keywords: Script-Based Animation, Interactive Illustrations, Emphasis Techniques, Image-Text-Relation, Fisheye Zoom Techniques

1 Introduction

In many areas, learning involves the study of complex 3D-phenomena. This is especially true for anatomy, where highly complex structures are studied under different aspects and viewing directions. To get an insight into the spatial structure and to be able to name parts of objects, are important issues in the learning process.

Traditional teaching materials do not support all aspects of this learning process sufficiently. Textbooks (see e.g. Sobotta (1988)) contain valuable drawings but require a lot of flipping through the pages. Numerous illustrations and the text referring to it have to be integrated mentally. Videos do a better job in explaining spatial relations because continuous transitions of different camera positions are possible. The person who explains uses pointers to direct the student to certain parts of the image.

The main problem with these media is the separation of initiative. When studying a textbook the student decides himself what to read and to look at. On the other hand, when watching a video the student can only influence the speed of the presentation.

Based on these observations we developed the ZOOM ILLUSTRATOR to combine techniques from traditional media with the flexibility of a computer system. Our system generates illustrations which are adapted to the parts currently explained. Thus a detailed study of interesting parts is possible while maintaining the context.

This paper describes the scripting language which controls the animation, the interactive mode and mechanisms to switch between them. The interpretation of a script results in commands to present text and to accentuate parts of the underlying 3D-model. To enhance comprehension, smooth transitions for both the display of textual information and changes of the graphics are realized.

The interactive mode allows to ask for explanations which are displayed using Fisheye Zoom Techniques, and further result in an accentuation of the parts referred to in the image. Emphasis is put on the combination of the interactive mode and the animation mode. While the beginner tends to work with prepared film-sequences the advanced student will shift to the interactive mode. Interactive exploration within an animation as well as access to film-sequences during interactive working are provided.

2 Related Work

Previous work on interactive anatomy teaching is based on scanned images or on 3D-models. The latter are more advanced because they allow flexible transformations.

The leading example in this field is the VOXELMAN™ (see e.g. Pommert *et al.* (1994)). This system exploits volume-models and offers free positioning and the facility to interactively cut off, i.e. literally to operate virtually. Although the images constructed this way are labeled and optionally explained, the coherence of pictorial and textual information is not treated in its own right, meaning that no automatic adaptation takes place in the image due to interaction on the textual part.

While anatomy teaching systems are related in terms of the application area, systems generating illustrated animations are important in terms of the underlying principles.

The generation of animations based on high-level scripts has been described in Zeltzer (1990) as well as in Karp and Feiner (1993). Zeltzer (1990) refers to the level on which an animation is specified as the *task level*, indicating that tasks (communicative intents in the terminology of Zeltzer) are supported. A simple example for a task is “*show object*”. To bridge the gap between task level specifications and the low-level commands for an animator (camera movements and geometric transformations) *decomposition rules* are used. These rules are oriented at *filmmaking heuristics* which are generally accepted practices which have evolved during the 90-year-history of cinema. Besides traditional methods, specific techniques from the medium *computer animation* are applied. This includes exploded views and cutaways.

The ESPLANADE-System (Expert System for PLANning Animation, Design and Editing) of Karp and Feiner (1993) exploits a sophisticated planning scheme to fulfill communicative intents. The design of ESPLANADE is guided by the hierarchical structure of traditional films with sequences, scenes and individual shots.

These systems are based on an object-structure within the scene-description. Furthermore, information is available, as to where an object is positioned and what it occludes. This information is used to calculate the transformations necessary to study certain aspects in detail.

Previous work concentrated either exclusively on the graphical part or on the textual part in combination with relatively simple images. The ZOOM ILLUSTRATOR is dedicated to illustrate complex models with a large space of related textual information. This requires flexible mechanisms to emphasize/deaccentuate parts of the image to adapt to a more or less detailed textual description.

3 Design of the Animation Component

This chapter describes the animation techniques developed and the combination of animations with interactive exploration. Our design is guided by educational videos for anatomy. We observed that parts of the model are colored to ensure easy recognizability. It is striking with how much effort the extent of complicated objects is explained. The explanation involves the removal of occluding parts. Pointing devices are used to make clear where an object starts and ends.

3.1 Animation Techniques

From these observations we concluded that to explain an object, it should be clearly recognizable and an appropriate camera position has to be chosen. Although the object of interest must not be occluded, the context should be maintained. Because of the importance of proportions we avoid distortions and selective scaling as a means to accentuate certain parts. From these principles the following techniques are derived:

- *Automatic Transformation*: Positioning the camera so that the object to be explained is visible and appears large enough.
- *Direct Emphasis*: Modifications of an object to direct the viewer towards this object. Colors and textures are important parameters with respect to emphasis. Textures, however, cannot be arbitrarily modified because their use may lead to false impressions of the surface structure.
- *Indirect Emphasis*: Changes of objects to deaccentuate them and thereby to pronounce other objects. This includes the modification of occluding objects, which are removed, clipped or made (semi-)transparent. Furthermore, the colour, especially saturation and brightness, is modified to deaccentuate objects.

Besides these general, dedicated methods for complicated objects are provided:

- *Clarify Shape*: The shape is shown with pointing devices (short: pointers) which are moved from the starting point to endpoint(s) and follow the contour of the object to be explained. This technique is especially helpful for objects with branching structures, like muscles.
- *Separate*: An object is separated, presented in a second window and slowly rotated. The rotation is stopped in some standard viewing directions and in an optimal viewing direction (optimal in terms of visibility and projected size).

3.2 Design of the Data Structures

This section describes the structure of the information necessary to realize the animation techniques listed above. To clarify this description, two terms are explained. An *object* is a part of the geometric model. Each object corresponds to a part of the textual description (with labels and explanations), which is called *node*.

3.2.1 Additional Information about the Graphics

To apply the techniques explained in Section 3.1, more information about the 3D-model than the mere geometry is required. For automatic transformations and indirect emphasis, the following information about the 3D-model is necessary:

- Standardized viewing directions from textbooks, optimal viewing direction
- For a number of precalculated viewing directions: size of the projected object, visibility, position (image coordinates), occluding objects

This information is summarized in the *visibility information*. For each object it contains 64 viewing directions (azimut- and declination-angle are varied in steps of 45 degrees). This data structure is gained by an offline calculation in which the rendered image is analyzed once for each direction. The analysis is performed by casting rays into the scene and evaluating which objects are hit in which order.

The visibility of an object o is a relative value, which is the quotient of the number of rays which hit o first and the overall number of rays which hit o. The system registers not only which objects occlude o in a specific viewing direction, but also how often o_i was hit before object o and thereby to which extent o_i occludes o. This information is evaluated to decide how occluding objects should be handled if o is explained. If o_i occludes o only to a small amount, o_i can be clipped against the bounding box of o. If on the other hand o_i almost totally occludes o it is made semitransparent or removed.

3.2.2 Structural Information on the Textual Descriptions

While the visibility information contains additional information on the geometric model, structural information about the related textual information is needed as well.

This *structure information* assigns a category, for our domain an organ-system, to each node. Furthermore, subcategories can be assigned e.g. to summarize nodes of a region. Although appropriate explanation techniques depend on the specific object, an assignment of animation techniques to categories is useful as a default value. The shape of muscles is often complicated, whereas most bones have a simpler structure. This information is collected in the *structure information* and described below:

1. Membership of nodes to categories/subcategories
2. *Aspects*, under which nodes from a category can be textually explained
3. Hypertextlinks between nodes under a certain *aspect*
4. Default animation techniques for a certain category

The information on memberships of nodes (1) is used to change the properties of all objects of a category. Thus the saturation of all muscles can be reduced in order to recognize one of them better. The *aspects* (2) of a node depend on its organ-system. At least a label and a short explanation are assigned to each object. Additional aspects are possible, e.g. muscles have an aspect “Function”. Each aspect is characterized by its name and a value to indicate how detailed this information is. The registration of links (3) is used to evaluate which nodes are conceptually near to the one currently explained. The default techniques (4) specify which animation techniques should be employed. Furthermore it is stated which aspect of textual information will be presented, concerning the explained object or linked objects.

To apply the techniques presented in Section 3.1 at all, the *visibility information* is necessary. *Structural information* gives hints about when to use which technique. One problem is still left out: how to demonstrate the path an object takes. The application of this technique assumes information on starting-points and branches. To define these points from an analysis of the geometry, is complicated and not generally applicable. Therefore an interactive approach was chosen which will be described later on. So far we only assume that a path exists.

3.3 Design of the Scripting Language

Important questions when designing a formal language are *what* should be expressed and *who* should use it. We call the person who writes a script the *author*. The author is assumed to know the content to be illustrated well. The scripting language allows him to specify easily *what* should be explained in *how much detail*.

For this purpose the author does not need to know which data structures are used and relies on the information organized by a *Content Provider.* The role of the author is similar to that of a lecturer who uses a textbook someone else has written.

Our experience with interactive illustrations is that a the connection between rendered images and related text is crucial for comprehension (Preim *et al.* (1995)). From this experience we suppose that a coupling of these media is also essential for the efficiency of educational animations. Therefore we provide constructs to state that changes on the graphical part are accompanied by simultaneous changes of text presentations.

Smooth transitions are required for both, changes on the textual part and on the graphical part. For smooth changes of text presentations Fisheye Zoom Techniques (see Dill *et al.* (1994)) are exploited. They allow continuous scaling of rectangular areas for the presentation of text with one node growing at the expense of others which are automatically scaled down. The text presented is adapted to the space available. Rüger *et al.* (1996) describes the application of these techniques for illustrations.

Scripts are basically sequences of *explain*-statements specifying what should be explained. An optional parameter list specifies the timing and the techniques to apply. An *explain*-statement can be added by an *emphasize*-statement to specify which techniques should be used. A *while*-statement specifies what should happen on the textual part in the meantime. To stress the role of the language to generate commands for the ZOOM ILLUSTRATOR, we call it *Illustrator Control Language (IC-Language).*

Concerning the timing, we decided that all movements which are connected with a distance (linear or angular in the case of a rotation) are specified in terms of speed so that the timing is relative to this distance. The display of textual information can only be specified as a duration. Relative specifications are superior to absolute ones.

To explain a group of objects, summarizing text is required. For this purpose we designed a *table*-statement and a *title*-statement. The latter serves to list all objects of a (sub-)category under the name of this (sub-)category. Similar, but more flexible is the *table*-statement. It generates a table with descriptions of all objects involved, where aspects can be specified. To define the details of the layout, a *format*-specification is available. The *IC Language* is described in more detail in Preim *et al.* (1996).

3.4 Combination of the Animation Mode with the Interactive Mode

Our previous work concentrated on interactive learning. As already noted in the introduction of this paper, interaction alone is not sufficient, hence we added animation techniques. However, we regard the coupling of interactive and animation techniques as the key for a good learning help. This leads to several interesting questions:

- Which interaction – over and above the video-recorder functionality of spooling – is useful for the viewer of an animation? (interactive interrupts of an animation)
- Which kind of film-sequences can be included to enhance interactive working? (animation during interactive exploring)

The answers to these question will be given later in connection with sample outputs.

4 Architecture

The ZOOM ILLUSTRATOR reads a *Scene Description*, containing a polygonal model, and related *Textual Information* derived from Sobotta (1988). The structure of the *Scene Description* corresponds to that of the *Textual Information*. Furthermore, the *Visibility Information* and the *Structure Information* (recall Section 3.2) are read in. From these sources an *Internal representation* is constructed which guides the execution of the *Animation script*. Figure 1 shows the relation between these components.

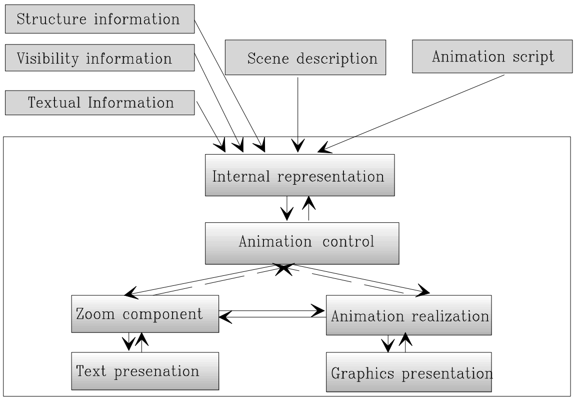


Figure 1: Architecture of the Zoom Illustrator

The ZOOM ILLUSTRATOR can be operated in interactive mode, in animation mode, or in a mixture of both. An illustration (see Figure 2, next page) with an image and textual labels is generated from the specification of the 3D-model and the categories of interest. The details of the layout-generation are described in Preim *et al.* (1995).

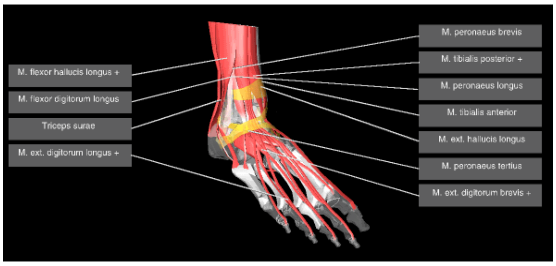


Figure 2: Illustration of a foot with focus on the muscles

The model can be transformed freely while the labels and the reference lines which connect them to the graphics are updated. On the textual part, explanations can be requested which result in zoom steps to accommodate the explanation.

The way the system works does not differ that much between animation mode and interactive working. The main difference is that in the animation mode the *Zoom component* and the *Animation realization* are triggered by the *Animation control* instead of getting “interactive requests” from the user.

5 Implementation

The ZOOM ILLUSTRATOR was developed on Silicon Graphics Indigo² Workstations using the graphics library Open Inventor™. The interaction mechanisms offered and the support for animation predestine Open Inventor for the purpose at hand.

We experimented with a commercially available model with 6000 polygons. It offers enough detail for stepwise exploration but is not too complex to be displayed at interactive rates. The assignment of textual information is based on Sobotta (1988).

For the realization of smooth transitions on the graphical part, the concept of the Inventor Engines is helpful. Engines support animation and simulating object behaviour. Special engines allow the interpolation between initial and final states. Several changes can be specified to happen at the same time. This mechanism is employed for geometric transformations as well as for changes of material properties.

In Section 3.2 we left out how the path to show an object’s shape is defined. We developed an interactive tool in which the object to be explained is handled separately. The author marks the starting point, branching points and endpoints resulting in a tree structure of the path to show (see Figure 3). The path contains 3D-points and is therefore independent of the viewing direction.

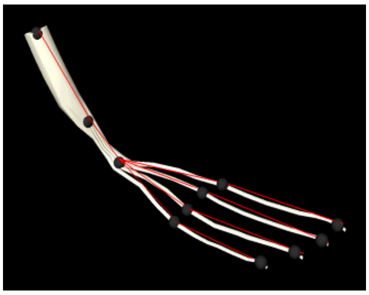


Figure 3: Interactive path definition of a muscle in a separate window

Additional points are derived from the coordinates of that object. These points are extracted near the linear connection of two control points. If the object is strongly curved no coordinates can be found near this connection. In this case the points derived strongly influence the path. With this procedure the surface structure can be well approximated. The author can test the path and modify control points.

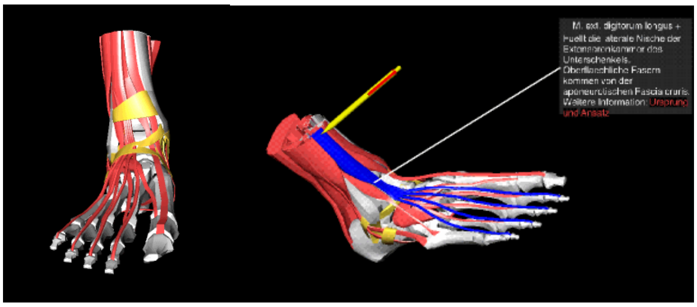
Path-specification is not the only interesting detail in the demonstration of an object’s shape.

Other important issues are: Which pointers are appropriate? How to move the pointer along the path?

According to our observations, we modelled objects used in videos for anatomy teaching to point at something. These include tweezers, scissors and ballpoints. In addition, we modelled some 3D-arrows. These pointers can be selected via a symbolic name. The movement of a pointer is guided by the current surface normal, that is, the pointer is always perpendicular to the path. The peak of the pointer touches the surface directly. This is in contrast to human pointing gestures, where a gap between the pointer and the object to explain remains. We believe that this gap is due to the fear not to damage the object and is therefore not necessary for our virtual objects.

6 Examples

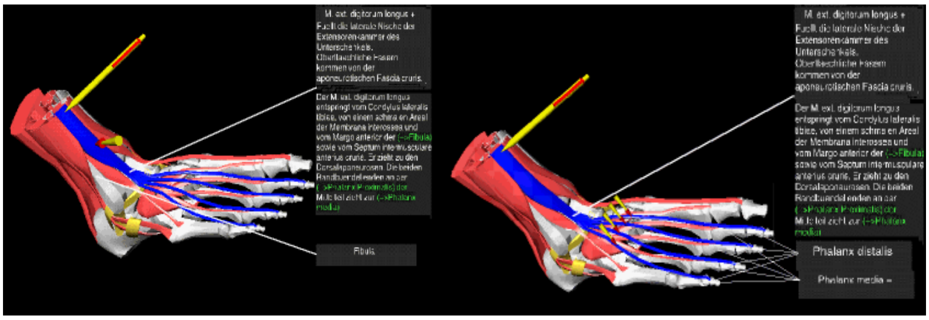
The left image of Figure 4 shows the start of an animation, while the right image shows modifications to explain a muscle.



M. ext. digitorum longus +
Fuellt die laterale Nische der Extensorenkammer des Unterschenkels. Oberflaechliche Fasern kommen von der aponeurotischen Fascia cruris. Weitere Information: Ursprung und Ansatz

Figure 4: Left image: Original position; Right Image: Modifications to prepare the explanation of the course. The camera position has been adjusted and a ballpoint is directed towards the origin of the emphasized muscle.

Figure 5 presents how the animation continued. The course is explained with arrows, while the ballpoint remains unchanged. An explanation is displayed, which describes the course. Structure information is used to display the labels of related nodes.



M. ext. digitorum longus +
Fuellt die laterale Nische der Extensorenkammer des Unterschenkels. Oberflaechliche Fasern kommen von der aponeurotischen Fascia cruris.

Der M. ext. digitorum longus entspringt vom Condylus lateralis tibiae, von einem schmalen Areal der Membrana interossea und vom Margo anterior der (->Fibula) sowie vom Septum intermusculare anterius cruris. Er zieht zu den Dorsalaponeurosen. Die beiden Randbuendel enden an der (->Phalanx Proximalis) der Mittelteil zieht zur (->Phalanx media)

M. ext. digitorum longus +
Fuellt die laterale Nische der Extensorenkammer des Unterschenkels. Oberflaechliche Fasern kommen von der aponeurotischen Fascia cruris.

Der M. ext. digitorum longus entspringt vom Condylus lateralis tibiae, von einem schmalen Areal der Membrana interossea und vom Margo anterior der (->Fibula) sowie vom Septum intermusculare anterius cruris. Er zieht zu den Dorsalaponeurosen. Die beiden Randbuendel enden an der (->Phalanx Proximalis) der Mittelteil zieht zur (->Phalanx media)

Figure 5: Left image a large arrow demonstrates the upper part. After reaching a branching point, the arrow splits (right image) into several small arrows. As the motion continues, the label of the bone where the muscle starts disappears while others become displayed.

Now let us come back to the questions from Section 3.4: How can an animation – as illustrated by the images of this chapter – be interrupted for interactive exploration? How can interactive exploration be enhanced by film-sequences?

The hypertext links displayed in an explanation can be activated. Such an activation leads to the display of an explanation to this object. Moreover, the viewer is asked whether this object should be explained by an animation, which may result in a sequence as demonstrated in Figure 4 and Figure 5. The rotation can be stopped and the viewer can rotate the explained model freely.

Figure 6 illustrates the answer to the second question. The left image of Figure 6 contains an example from the interactive mode. All muscles are labeled and the user requested an explanation (right part of left image). This causes the other rectangles on this part of the image to be scaled down; their labels are automatically hidden.

The user may rotate the model so that it is clear what is explained. This, however, is a time-consuming trial and error-process. To work more efficiently, the user can start the *explain*-dialog and specify the parameters for a short film (or use the defaults). From this specification an explain-statement of the *IC Language* is interpreted.

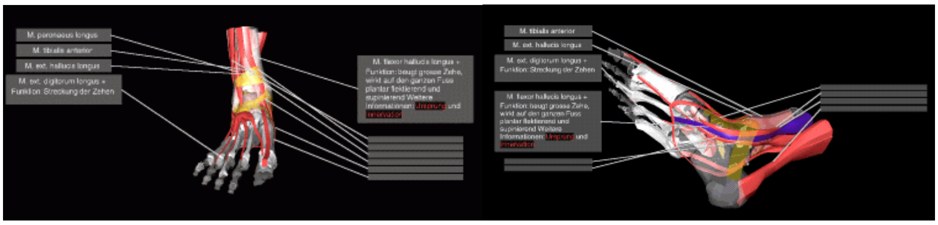

M. peronaeus longus
M. tibialis anterior
M. ext. hallucis longus
M. ext. digitorum longus + Funktion: Streckung der Zehen
M. flexor hallucis longus + Funktion: beugt grosse Zehe, wirkt auf den ganzen Fuss plantar flektierend und supinierend Weitere Informationen: Ursprung und Innervation
M. tibialis anterior
M. ext. hallucis longus
M. ext. digitorum longus + Funktion: Streckung der Zehen
M. flexor hallucis longus + Funktion: beugt grosse Zehe, wirkt auf den ganzen Fuss plantar flektierend und supinierend Weitere Informationen: Ursprung und Innervation

Figure 6: Combination of interactive exploration and animation: Left Image: The user requested an explanation on a muscle (upper right of the left image). After the display of an explanation the user „asks“ to transform the model to the optimal viewing direction, rendering a sinew transparent which occludes the muscle.

7 Conclusions and Future Work

This paper describes strategies for illustrating complex 3D-models using a combination of interactive exploration and animation. For the techniques developed geometry data, related visibility information, textual descriptions and information concerning structure are exploited. We developed the IC-Language for specifying an animation. To explain the shape of objects, we developed special pointing techniques.

The ZOOM ILLUSTRATOR focuses on a consistent presentation of rendered images and explaining text. In this paper we extend this approach by introducing an automated adaptation of text attributes to changing parameters of the model display. This allows to transform the model and display text information at the same time. Future work should include “before” - “after” specifications of changes on the graphical part and text presentations.

Our system is based on a huge amount of information concerning the subject to be studied, in this case anatomy. So far this information is collected and updated manually including complex consistency checks. A data base to collect this complex information is desirable as well as facilities to automate data acquisition.

The animation techniques incorporated into the ZOOM ILLUSTRATOR are designed especially for beginners seeking for an overview. Once being familiar with the system, the student composes his or her own combination of interactive and animation features to accomplish the learning task efficiently.

The usability has to be proofed in a field test which has to be conducted in the time to come. Questions to be answered are whether our system covers the main features a student expects and whether our scripting language provides enough flexibility.

References

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

Karp, P., Feiner, S. (1993)
"Automated Presentation Planning of Animation using Task Decomposition with Heuristic Reasoning", *Proc. of Graphics Interface*, Toronto, May, pp. 118-127

Pommert, A., Pflesser, B., Riemer, M., Schiemann, T., Schubert, R., Tiede, U., Höhne, K. H. (1994)
"Advances in Medical Volume Visualization", *Eurographics '94, State of the Art Report*, Eurographics Technical Report, ISSN 1017-4656, pp. 111-139

Preim, B., Ritter, A, Steinicke, G. (1996)
"Gestaltung von Animationen zur Erklärung komplexer Modelle", *Proc. of Fachtagung für Simulation und Animation für Präsentation, Planung und Bildung*, Magdeburg, February, pp. 255-266

Preim, B., Ritter, A., Strothotte, T., Pohle, T., Bartram, L., Forsey, D. R. (1995)
"Consistency of Rendered Images and Their Textual Labels", *Proc. of CompuGraphics*, Alvor, Portugal, December, pp. 201-210

Rüger, M., Preim, B., Ritter, A. (1996)
"Zoom Navigation – Exploring Large Information and Application Spaces", *Proc. of Advanced Visual Interfaces*, Gubbio, Italy, May (to appear)

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

Zeltzer, D. (1990)
"Task-Level Graphical Simulation: Abstraction, Representation and Control", in: Badler, N., Barsky, B., Zeltzer, D. *Making them Move: Mechanics, Control and Animation in Articulated Figures*, pp. 3-33