Computational Visualistics: Dealing with Pictures in Computer Science

JÖRG R. J. SCHIRRA

Building blocks from many disciplines have to be integrated into a general science of images. Computational visualistics has been formed as a contributing field embracing all aspects of dealing with images computationally. Two basic concepts of computer science are introduced. Applied to the concept »image«, they determine the methodological core of computational visualistics. As the contribution of computer science to the subject of image theory, interactive pictures are examined. Finally, relations to other "image sciences" are sketched.

Bausteine vieler Disziplinen müssen in eine allgemeine Bildwissenschaft integriert werden. Als Beitrag aus der Informatik versteht sich die Computervisualistik, die alle Aspekte rechnergestützten Umgangs mit Bildern umfaßt. Zwei Grundbegriffe der Informatik werden vorgestellt und bestimmen, auf den Begriff „Bild" angewendet, den methodologischen Kern der Computervisualistik. Als Beitrag der Informatik zum Gegenstand der Bildtheorie werden interaktive Bilder betrachtet. Schließlich werden die Beziehungen zu anderen „Bildwissenschaften" kurz umrissen.

1 Image Science and Computer Science

Images take a rather prominent place in contemporary life in the western societies. Together with language, they have been connected to human culture from the very beginning. For about one century – after several millennia of written word's dominance – their part is increasing again remarkably. Steps toward a general science of images, which we may call 'general visualistics' in analogy to general linguistics, have only been taken recently [SACHS-HOMBACH *2003*]. So far, a unique scientific basis for circumscribing and describing the heterogeneous phenomenon "image" in an interpersonally verifiable manner has still been missing while distinct aspects falling in the domain of visualistics have predominantly been dealt with in several other disciplines, among them in particular philosophy, psychology, and art history. Last (though not least), important contributions to certain aspects of a new science of images have come from computer science.

In computer science, too, considering pictures evolved originally along several more or less independent questions, which lead to proper sub-disciplines: computer graphics is certainly the most "visible" among them. Only just recently, the effort has been increased to finally form a unique and partially autonomous branch of computer science dedicated to images in general. In analogy to computational linguistics, the artificial expression 'computational visualistics' is used for addressing the whole range of investigating scientifically pictures "in" the computer [SCHIRRA *2000*].

Computer science, the endeavor of studying scientifically computers and information processing, has two different roots determining its methodology. In some aspects, computer science is a typical *structural science* like mathematics and logic [WEIZSÄCKER *1971, 22*]: their subjects are purely abstract entities and the relations between them – entities like Turing machines far off of our living practice, at best linked to everyday life by means of an additional interpretation relation. With respect to some other aspects, computer scientists are like electrical engineers interested in *engineering* problems, an interest resulting in concrete artifacts that have already changed our life dramatically during the past few decades and continue to do so with growing acceleration. The fluctuation of the focus of attention between structural science and engineering is characteristic for all investigations in computer science, and thus, is valid for the dealing with pictorial data, as well. On the one hand, certain abstract data types for pictorial representations are investigated and designed from a purely structural point of view. For example, efficiency properties are examined, or minimal substructures for particular tasks determined. On the other hand, concrete algorithms (based on those data structures) for, e.g., picture processing are "software-engineered" and used in diagnosis – with considerable influence on our social structure.

In the following, a survey is given about the methodology of computational visualistics and its main fields of interest. First, an introduction is given of the two basic concepts determining the methods of computer science: »data structure« and »implementation« (2). Applied to the concept »image«, the methodological core of computational visualistics can be deduced (3). Interactive pictures provide a characteristic example: they are the contribution of computer science to the subject of image theory (4). The final section (5) sketches the relations between computational visualistics and other "image sciences".

2 Data Structure and Implementation

It is not wrong to view computer science – as we often do colloquially – as the discipline dealing scientifically with computers and information processing. However, a better understanding evolves if we consider »data structure« and »implementation« as the basic concepts and main subjects of the field. The two concepts can more easily be related to central concepts in the philosophical theory of argumentation, and thus open up a better view on the methodology of computer science.

The processing of data is certainly a crucial theme for computer scientists, but it depends completely on the fact that data is always structured and grouped into types. Each such type implies a set of possibilities to "do something" with that kind of data: numbers can be added or multiplied (etc.); polygons in a geometric model can be moved or turned, mirrored or strained (etc.); but not *vice versa*. Usually, several data types and their interactions are relevant for a task. As it is only important here that we can perform some operations with one kind of data so that certain relations hold between their results while ignoring the concrete manner of how those operations are actually realized, computer scientists consider *abstract data structures* – abstract entities that grasp exactly the essential properties. Algebraic formulae or logical expressions are often used to that purpose: the former describe which operations transform the instances of which data type into what other type's instances; the latter determine which properties remain invariant after a certain sequence of operations. The methodology of computer science as a structural science is, then, partially covered by this first question: How can we find for a given class of problems an adequate (and efficient) data structure so that a procedural solution – an algorithm – can be given by means of combining the operations of that data structure?

A close relationship between abstract data structures and the understanding of a field of concepts can be seen when taking into account the philosophical theory of rational argumentation – an association that is also particularly well suited for studying the relations between computer science and other disciplines. Let us therefore elaborate this unusual approach to the subjects and methodologies of computer science a bit further.

We refer by the expression 'the concept »*X*«' – e.g., by the expression 'the concept »image«' – to everything that is structurally common to all explanations of '*X*' and its synonyms (e.g., the expression 'image'), that is, everything that "remains the same independent of how or in what language I formulate or show it" [WITTGENSTEIN *1953, I.23*]. Then, we never examine one concept alone. It is always a system of concepts that are mutually related and cannot be defined independently from each other, like »king«, »queen«, »knight, and »medieval society« (or alternatively »chess«); or, of course, »image«, »perception«, and »sign«. They belong to the same *field of concepts*. From the perspective of structural science, we can therefore view data types as a formalized version of certain concepts, and the corresponding data structure as the appropriate field of concepts. While concepts and their fields in everyday life often lack precision or may even be inconsistently organized, abstract data types must (usually) satisfy formal rules of consistency and completeness.

Relations between several fields of concepts are of particular interest for the theory of argumentation. The internal relations of one field may indeed be used to explain correct or wrong applications of the concepts of that field (or the expressions for these concepts) – presupposing however that all the parties involved in the debate agree that the field considered is appropriate at all. But in order to firstly motivate that presupposition for a critical-minded interlocutor: in order to explain why the internal rules are adequate conceptual rules in the frame of a rational argumentation, field-external relations have to be thrown into the game. Relations to those fields of concepts all the parties of the argumentation already agree upon have to play an important role [ROS *1999*]. We may try to reconstruct for our opponent the conceptual structures of the field in question as a systematic combination of the concepts already shared.

Take for example the schema used for introducing a new type of numbers in mathematics: imagine we only know about integer numbers and are to be introduced to rational numbers. Perhaps, somebody – let us say, a globe trotter interested in mathematics – told us about that kind of numbers he heard of in Arabia, and we on first view experience the described entities and their properties as rather strange. Or we spontaneously invented the specification (the description of the internal rules) like in a combinatorial game without being aware of doing more than a "game of glass-beads". In any case: the only thing we know for the time being is the abstract and purely symbolic specification of that concept. Whether such entities really exist, i.e., whether we here deal with a useful and correctly constructed concept, is still completely unclear.

How could our dialog partner (the mathematical globe trotter) convince us that these mathematical entities, which for us seem so curious, are possible and useful ("real objects", so to speak)? He could try to show us how to introduce this concept from the fields of concepts we already have (namely integer numbers). That is, he can show us the schema for constructing the concept of rational numbers by means of the one of integer numbers (as equivalence sets of pairs of integers, to be precise, i.e., as fractional numbers). This schema must specify how the primitives and the operations of the rational numbers are constructed using the primitives and operations of the integers. This could be done by means of a constructive operation that is neither part of the field of integers nor of the rational numbers. Our teaching dialog partner may postulate, for example, that every instance of a rational number can be represented by pairs of instances of integer numbers: we are very well able to recognize such pairs, following our preconditions. If we accept this introduction schema for the rational numbers, we are additionally able to *justify* (ground) the internal rules of the new field as given in its specification by means of the attributes of integers: that the equivalence class x/x (for all integers x that are not 0) is the neutral element of rational multiplication can now be derived from the rules of the integers and the fractional combination schema (etc.).

Analogously in computer science, an abstract data structure can be *implemented* by means of other data structures: the implementation provides us with "real" instances of data types that had only been symbolically defined by means of the abstract descriptions of the data types contained. Furthermore, a computer scientist may try to motivate – in a scientific paper or talk, for example – that an abstract data structure (and a particular algorithm defined within) does indeed "make sense" (i.e., does what we want it to do). She may do so by pointing out the construction schema of the data structure by means of those data structures supposedly accepted by her audience in advance – i.e., by giving a corresponding implementation.

Thus, »implementation« is a central concept of computer science derived from the notion of data processing. But it is also closely linked to computers, the second subject of computer science in the colloquial understanding: for the engineering perspective, computers are in fact implementation engines. If, for example, a group of engineers has reached an agreement that a certain artifact of electrical engineering indeed realizes the data structure of the integer numbers – i.e., the artifact "acts" like that (at least if no technical error occurs) – then, of course, the engineers can perform particular calculations with integer numbers by means of the artifact. But they may also take several copies of the artifact for constructing another technical artifact: one they are motivated to view as a realization of another data structure, e.g., the rational numbers, if its construction mirrors the abstract implementation schema of that data structure on the basis of the integer numbers. Therefore, realizations of an abstract implementation schema are often called 'technical implementation'. The engineers may use the new artifact for doing calculations with rational numbers. But they may also convince other persons (who agree already on the interpretation of the "integer artifacts") of that understanding of their "rational number machine" by explaining the abstract implementation schema.

Computers are a particular sort of engineering artifacts that – by general understanding – provide us through a chain of realizations of more elementary structures (e.g., assembler and register machines, binary numbers and logical gates, electron flows and magnetic bubbles, to name but a few) a technical implementation of a broad spectrum of useful data structures chosen in a way that one can employ them to implement more or less easily any other data structures (this is, of course, a variant of the famous Church/Turing thesis [KLEENE *1967, 232*]). And the search for a correct technical implementation of algorithms has to be counted as the second main method of computer science.

3 The Data Structure with »Image« – An Overview

For a science of images within computer science, quite obviously the abstract data type »image« (or perhaps several such types) stands in the center of interest together with the corresponding data structure(s) and the potential relations of implementation. Keeping the distinctions of the preceding section in mind, a reasonable *ad hoc* organization of the field could be derived by distinguishing the examinations of computational visualistics along the following three paths: we may be interested (a) in a purely field-internal consideration that concentrates exclusively on the abstract data structure around the type »image«, the basic operations that determine the structure, and the algorithms that can be defined with those operations; or (b) in the relations of implementation that may lead from more elementary data structures to the structure with the type »im-

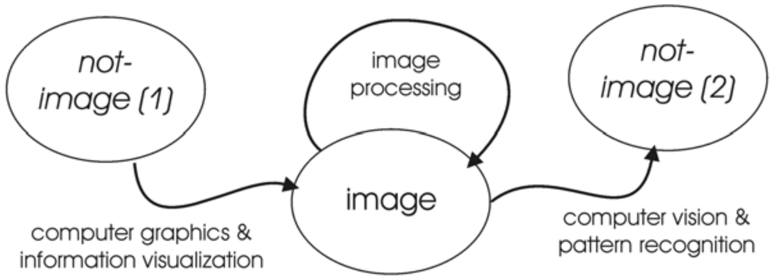

Figure 2: Sketch on the operations with the data type »image«

age«, and that would allow us to technically implement the image algorithms of particular value for us; or (c) in the relations of implementation that open up even more complex data structures on top of the one including the type »image«.

In fact, each of the "traditional" image-related sub-disciplines of computer science considers all three aspects to various degrees. The distinction actually establishing the disciplines follows a simpler semantic pattern resulting from the types of operations and algorithms around the data type »image«, which relate an instance of »image« with something that either is or is not of the same type (cf. Fig. 1). Only an overview is given in the following list:

- *Algorithms from »image« to »image«*
  In the field called *image processing* (cf., e.g., [GONZALEZ & WOODS *2002*]), the focus of attention is formed by the operations that take (at least) one picture (and potentially several secondary parameters that are not images) and relate it to another picture. With these operations, we can define algorithms for improving the quality of images (e.g., contrast reinforcement), and procedures for extracting certain parts of an image (e.g., edge finding) or for stamping out pictorial patterns following a particular Gestalt criterion (e.g., blue screen technique). Compression algorithms for the efficient storing or transmitting of pictorial data also belong into this field.

- *Algorithms from »image« to "not-image"*
  Two disciplines share the operations transforming images into non-pictorial data types. The field of *pattern recognition* (cf. e.g., [DUDA, HART & STORK *2002*]) is actually not restricted to pictures. But it has performed important precursory work for computational visualistics since the early 1950's in those areas that essentially classify information in given images: the identification of simple geometric Gestalts (e.g., "circular region"), the classification of letters (recognition of handwriting), the "seeing" of spatial objects in the images or even the association of stylistic attributes of the representation. That is, the images are to be associated with instances of a non-pictorial data type forming a description of some of their aspects. The neighboring field of *computer vision* (cf., e.g., [MARR *1982*]) is the part of AI (Artificial Intelli-

gence) in which computer scientists try to teach – loosely speaking – computers the ability of visual perception. Therefore, a problem rather belongs to computer vision to the degree to which its goal is "semantic", i.e., the result approximates the human seeing of objects in a picture.

- *Algorithms from "not-image" to »image«*
  The investigation of possibilities gained by the operations that result in instances of the data type »image« but take as a starting point instances of non-pictorial data types is performed in particular in *computer graphics* and *information visualization*. The former deals with images in the closer sense, i.e., those pictures showing spatial configurations of objects (in the colloquial meaning of 'object') in a more or less naturalistic representation like, e.g., in virtual architecture. The starting point of the picture-generating algorithms in computer graphics is usually a data type that allows us to describe the geometry in three dimensions and the lighting of the scene to be depicted together with the important optical properties of the surfaces considered (cf. section 4, and [FOLEY *ET AL. 1996*]). Scientists in information visualization are interested in presenting pictorially any other data type, in particular those that consist of non-visual components in a "space" of states: in order to do so, a convention of visual presentation has firstly to be determined – e.g., a code of colors or certain icons (cf., e.g., [WARE *1999*]). The well-known fractal images (e.g., of the MANDELBRODT set) form a borderline case of information visualization since an abstract mathematical property has been visualized.

The algorithms behind the arrows in Figure 1 may indeed consist of complicated combinations of all three possibilities mentioned above: For example, we may consider a procedure in computer graphics that is put in sequence after an algorithm of computer vision in order to solve a complex problem in image processing. Within that tri-partite framework, investigations in computational visualistics have a focus on structural aspects or on engineering problems – mirroring the traditional differentiation between a more mathematically oriented theoretical informatics, and the engineering-oriented practical and applied computer science.

4 Interactivity and the Data Structure with »Picture«

To the range of image uses, computer science has contributed essentially the *interactive* control of fast image-generating procedures. Computer graphics in particular has gained by that option: apart from still or moving pictures produced in advance, fantasy scenes modeled with the computer can be used in a broad range of "real-time applications", as they are called. They reach from computer games or simulators for pilot training to hyper media textbooks on anatomy or virtual reconstructions of important pieces of architecture (Fig. 2).

Figure 2: Screenshot from the virtual reconstruction of the House Behrens (View from the music room to the dining-room)

The interactive system allows the users to study the atmospheric total effects of the lost interior design of the important *Jugendstil* building of 1901 by moving freely around and even manipulate some objects, like doors. The house of PETER BEHRENS in Darmstadt, destroyed during Word War II, is famous for the unique coherence of design paradigmatic for the aesthetic considerations of that time [BUCHHOLZ & SCHIRRA *2001*]

The perspectives can be chosen spontaneously and (almost) freely by the users of such systems. Modifications of the underlying model can be initiated by user input, as well, so that their results are immediately visible in the picture. Even object-centered changes of rendering styles can be interactively triggered (Fig. 3).

The construction process for naturalistic ("photo-realistic") computer graphics consists essentially of two phases (Fig. 4). First a three-dimensional geometric model of the scene to be presented in the picture has to be provided (modeling). The actual calculation of the image is performed on that model by means of a projection into the two dimensions of the image plane (rendering). The geometric model is a (formalized) description based on a data structure that allows the computational visualist to describe three-dimensional geometric objects. For example, the surfaces can be given by means of polygons. The description also contains information about the optical properties of the surfaces and about the illumination of the scene. Taking into account the parameters of the "camera", i.e., the position and orientation of the imaginary point of view,

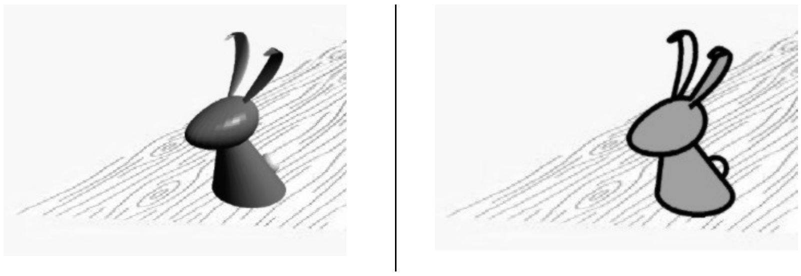

Figure 3: Simple computer-generated variations in presentation styles

From one geometric model, scenes with mixed presentation styles are rendered to enforce the rhetoric function of the picture content's elements. For topic parts, which are to be interpreted as background, a reduced presentation is sufficient. Predicative parts, as the visual appearance of the bunny in the left example or the mere configuration of its parts in the right example, are shown in more detailed presentations styles according to the communicative focus [SACHS-HOMBACH & SCHIRRA *2002*] & [HALPER *ET AL. 2002*].

the rendering procedure creates an instance of the data type »image«. Today, corresponding algorithms are quite well optimized; their implementation is based on specific graphic hardware. In consequence, the projections can be calculated so fast that changes initiated by the user – e.g., movements of the camera or of a scene object – cause corresponding alterations in the image without perceptible delay.

Pictures produced in that manner can be easily changed and redone. In contrast to manipulations of the real world, changes in the geometric model or projection parameters are essentially reversible without problems, an advantage over and above photography that seems to make computational visualistics attractive for the production of naturalistic quasi-photos (or even of artificial cinema). Correspondingly, the task of increasing the degree of naturalism has dominated the research in computer graphics for a long time [WATT, POLICARPO *2001*]. Nevertheless, we have no hint so far that naturalism is the only possible goal, or that it can be achieved particularly well in contrast to other styles by model-based computational visualization. For many applications, pictures of spatial scenes represented with reduced naturalism are indeed more useful [STROTHOTTE *1998*].

Of course, the fully-developed data structure around the type »image« must be accommodated to the two defining aspects of pictures – the close relation to visual perception of spatial scenes, and the use as a sign. If one of these aspect is missing, interactivity as the special contribution computer science has to offer to picture uses cannot be adequately dealt with.

First: it is characteristic for the concept »image« to contain the potential of deceiving somebody who perceives it. By not registering the presence of a picture, the scene depicted may be mistaken for the real thing. The experience of immersion – of being immersed in a virtual world – is particularly important for interactive computer pictures, for example in their application in computer games. The apparently immediate feedback of (stylized) proper motions ob-

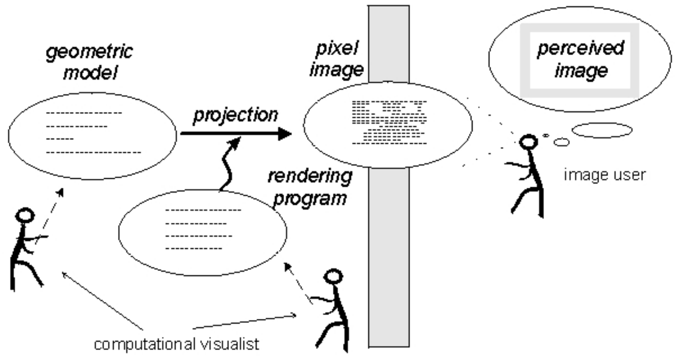

Figure 4: Visualization of the construction process for computer graphics

scures the differences between the perception (actually taking place) of an image to the perception of the scene that is only represented. Obviously, that scene – the image content – must be considered as part of the data structure »image«, at least in the reduced form of a geometric model. The intern relations of that field between image content and the picture vehicle must be governed by the principles of (visual) perception, including Gestalt criteria and in particular the constitution schema of sortal objects [SCHIRRA *forthcoming*].

Second: it is also a characteristic part of the concept »image« that a picture is conceived of as a sign, i.e., something directed in a sign act by somebody to someone else (or the sender in a different role) in order to focus or keep focused the common attention to something. Variations in the presentation style are ideal tools for adapting a picture to a particular communicative function at hand. During the last decade, a major project in computer graphics has become to develop alternative rendering procedures that do not project the model photorealistically into the image. Several forms of transformed perspective like fisheye techniques, which are able to present details within a larger context, belong here, as well as procedures for generating pictures in the style of free-hand sketches, or cross-hatching algorithms leading to results similar to copper-engravings. Mixing presentations styles by means of smart algorithms to generate rhetorically enriched pictures obviously depends on a clear definition of the communicative functions intended for the pictures.

In order to guaranty the communicative function of computer-generated images, it must be clear in each single case who is communicating with whom. This is not always obvious, in particular if we look at interactive systems. Although the picture is still produced by means of the rendering algorithm, this happens at some point in time and place apart from the person to be considered as the sender in this communication (Fig. 5). We may call the situational separation between the design activities of the computational visualist and the actual image production that is finally induced by the image user by the expression 'tele-rendering'. In fact, tele-rendering does not transfer single messages but whole classes of messages, one instance of which is realized in a particular user

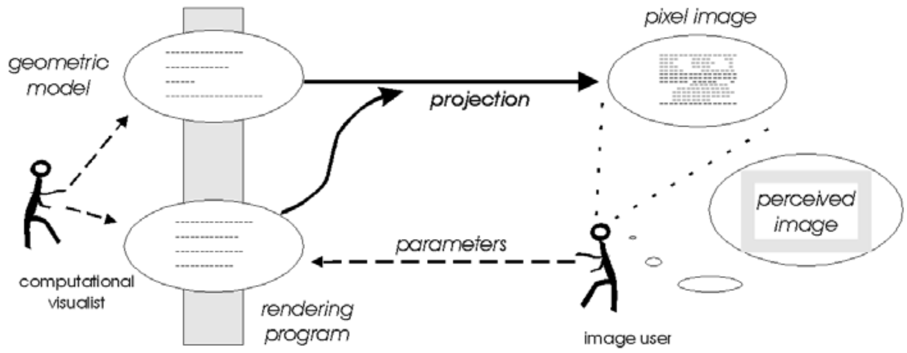

Figure 5: Schema of Tele-Rendering

session depending on the user's interaction. The selection is performed following rules given by the developer of the system but without his or her immediate influence. It is therefore reasonable to conceive of this type of pictorial communication originally gained by means of computers as a member of a different class of media – along with language generation, a centerpiece of AI research and another core of interactive systems. Taking up a well-known distinction of media theory [PROSS *1972*],[1] we suggest to call this type "media of class IV".

Note that computer graphics does not necessarily imply tele-rendering although it has opened the way for the latter: computer graphics' potential to easily change the model or the style of rendering provides a significant variability of rhetoric elements adaptable to the communicative context. The use of pictures in a medium of class IV has however severe consequences for many pragmatic aspects. For example, we have to re-judge the question of who has to take responsibility for a particular presentation of a picture in a certain context of use of the interactive system, since both sender and receiver have only an indirect part in the picture's creation.

Take for example a textbook on human anatomy and its interactive pendant. In the book and in the interactive version, pictures illustrating anatomic objects, some of their relations, and some of their attributes are offered. The standard situation of use appears as a (pictorial) soliloquy: for example, a student uses the pictorial sign for focusing his attention on those anatomic matters in order to learn them. Or a physician wants to refresh her memory by means of showing that sign to herself. Although acting as sender and receiver simultaneously, the student and the physician have to trust the original picture producer and the technical devices transporting the sign vehicle to them.

[1] Media of class I do not involve any technical devices that open the possibility of temporally or spatially separating the communicative partners. Class II media, like books or letters, involve devices on the producers side and overcome essentially temporal separation. Class III media, like TV or telephone, have technical devices on both sides of the communication channel so that a spatial separation of the communicative partners can be accomplished. In any case, the units transmitted are single messages, not classes of messages.

For the traditionally printed textbook, this trust is essentially established by means of the social institution of the initial production process: the produced picture is persistent; it usually does not change significantly. This attribute is also viewed as a disadvantage of the traditional medium, which is finally cured by the interactive version. The users of an interactive textbook are not restricted to static, pre-fabricated images anymore; they can easily chose other perspectives. They may turn, scale, move or remove parts of the anatomic objects displayed, zoom in or out, and even change the style of the presentation. To that purpose, the image is rendered at presentation time. The rhetoric force of each concrete picture generated for a specific user must be carefully adapted by the interactive system to the particular communicational setting at hand if an insufficient act of pictorial communication with potentially fatal consequences is to be prevented.

In this context, a tool is being discussed for controlling pragmatic aspects of pictorial communication that has already been employed successfully in natural language systems: a *user model* also has to be part of the data structure »image«. In essence, the parameters of rendering like camera position, frame, style parameters and the selection of parts of the geometric model to be shown are determined depending on three factors: the knowledge the user is assumed to have *a priori*, information that has already been transferred by prior picture presentations (or other communication) in the same session, and the communicative interests as far as they are deducible by the user's actions with the system.

5 Special Relations to Other Picture Sciences

The access of computational visualistics to pictures is stamped by the ideal task of providing easily controllable tools for other image users that help them to solving their particular problems or to support their interests. Here, the expression 'tool' refers essentially to a compound of technical implementations of picture-generating and picture-processing algorithms that can be run on a concrete computer. Thus, it is in particular the engineering aspect of computer science that is demanded. Too much space would be needed to simply list all the possible domains of application: as examples, medicine and animation films may be at least mentioned. In general, no discipline with interests in image science can do without computers these days in order to assist their work or to present their results. Apart from quite general tools that do not offer a particular functionality and allow a lay person an easy access, systems for special tasks have to be constructed that offer highly efficient solutions but often can only be controlled by a specialist after a considerable amount of training.

The data structures underlying those algorithms should approximate as much as possible the fields of concepts that structure the application domain under question. The task of transforming that information adequately into data structures challenges particularly computer science in its aspect as a structural science. Then, philosophy plays a more central role as a general reference discipline for computational visualistics than art history, which is more prominent in other respects of image science: while the former strives for a general basis for any discussion of the concept »image«, the latter is mainly engaged with historical phenomena and considers in particular aesthetic pictures.

The link between computational visualistics and art history is opened on a completely different level by means of computer art. The expression 'computer art' does not refer to a particular style distinguishing those works from other kinds of visual art. Rather, the computer being the tool with which the artist creates a piece of art counts as the criterion.

Considerations from psychology, ergonomics, and design have an important influence in computer science on the creation of the user interfaces necessary for the tools developed. As *graphical* user interfaces, which today form the majority, fall clearly in the range of computational visualistics, corresponding interdisciplinary links to image-related aspects of those disciplines are inevitable.

Since all the disciplines mentioned above do not only profit from computer science but give important impulses for the tasks of computational visualists, corresponding basic knowledge has to be deep-seated in the education programmes of computational visualistics. Such an interdisciplinary approach then also contributes to the establishment of a general visualistics presently at issue.

Referenzen

BUCHHOLZ, KAI, SCHIRRA, J. R. J.: Das Haus als Gesamtkunstwerk – eine Herausforderung an die Computervisualistik. In: K. SACHS-HOMBACH (Hrsg.): *Bildhandeln – Interdisziplinäre Forschungen zur Pragmatik bildhafter Darstellungsformen*. Magdeburg: Skriptum, 2001, 241–268.

DUDA, RICHARD O.; HART, P. E.; STORK, D. G.: *Pattern Classification*. New York: Wiley, 2002.

FOLEY, J. D.; VAN DAM, A.; FEINER, ST. K.; HUGHES, J. F.: *Computer Graphics, Principles and Practice*. Reading, MA: Addison Wesley, ^{2}1996.

GONZALEZ, RAFAEL C.; WOODS, R. E.: *Digital Image Processing*. Reading, MA: Addison Wesley, 2002.

HALPER, NICK & SCHLECHTWEG, ST. & STROTHOTTE, TH.: Creating Non-Photorealistic Images the Designer's Way. In Spencer, St. N. (ed.): *Proceedings NPAR 2002: Symposium on Non-Photorealistic Animation and Rendering.* New York: ACM SIGGRAPH, 2002, 97–104.

KLEENE STEPHEN COLE: *Mathematical Logic*. New York: John Wiley, 1967.

MARR, DAVID: *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. San Francisco: W.H. Freeman, 1982.

PROSS, HARRY: *Medienforschung: Film, Funk, Presse, Fernsehn*. Darmstadt: Habel, 1972.

ROS, ARNO: Was ist Philosophie? In: RICHARD RAATZSCH (Hrsg.): *Philosophieren über Philosophie*. Leipzig: Leipziger Universitätsverlag, 1999, 36–58.

SACHS-HOMBACH, KLAUS: *Elemente einer allgemeinen Bildwissenschaft*. Habilitationsschrift, FGSE, Univ. Magdeburg 2003. (to appear at: Halem Verlag)

SACHS-HOMBACH, KLAUS & SCHIRRA, JÖRG R. J.: Selecting Styles For Tele-Rendering. Toward a Rhetoric in Computational Visualistics. In: *Proc. 2nd International Symposium on Smart Graphics* 2002, Hawthorne, NY: ACM Press, 102–106.

SCHIRRA, JÖRG R. J.: *Variations and Application Conditions Of the Data Type »Image« – The Foundation of Computational Visualistics*, Habilitationsschrift, FIN, Univ. Magdeburg, forthcoming.

SCHIRRA, JÖRG R. J.: A New Theme for Educating New Engineers: Computational Visualistics. In: *Global Journal of Engineering Education*, 4(1):73–82, 2000.

STROTHOTTE, THOMAS ET AL.: *Computational Visualization*. Berlin: Springer, 1998.

WARE, COLIN: *Information Visualization – Perception for Design*. San Francisco: Morgan Kaufman, 1999.

WATT, A., POLICARPO, F.: *3D Games, Real-Time Rendering and Software Technology*. Reading, MA: Addison-Wesley, 2001.

WEIZSÄCKER, CARL-FRIEDRICH V.: *Die Einheit der Natur*. München: Klett-Cotta, 1971.

WITTGENSTEIN, LUDWIG: *Philosophical Investigations.* Oxford: Basil Blackwell, 1953.

The interactive House Behrens is accesible under:
http://www.JRJS.de/Work/Projects/Behrens/index.html