Journal of Universal Computer Science, vol. 7, no. 5 (2001), 400-409
submitted: 14/4/01, accepted: 18/5/01, appeared: 28/5/01

Graphics Content in Digital Libraries: Old Problems, Recent Solutions, Future Demands

Dieter W. Fellner
Braunschweig University of Technology, Germany
d.fellner@tu-bs.de

Abstract: Working with the ubiquitous 'Web' we immediately realize its limitations when it comes to the delivery or exchange of non-textual, particularly graphical, information. Graphical information is still predominantly represented by raster images, either in a fairly low resolution to warrant acceptable transmission times or in high resolutions to please the reader's perception thereby challenging his or her patience (as these large data sets take their time to travel over congested internet highways).

Comparing the current situation with efforts and developments of the past, e.g. the Videotex systems developed in the time period from 1977 to 1985, we see that a proper integration of graphics from the very beginning has, once again, been overlooked.

The situation is even worse going from two-dimensional images to three-dimensional models or scenes. VRML, originally designed to address this very demand has failed to establish itself as a reliable tool for the time window given and recent advances in graphics technology as well as digital library technology demand new approaches which VRML, at least in its current form, won't be able to deliver.

After summarizing the situation for 2D graphics in digital documents or digital libraries this paper concentrates on the 3D graphics aspects of recent digital library developments and tries to identify the future challenges the community needs to master.
Category: I.3.5, H.3.7

1 Introduction

Digital Libraries have gained much attention over the past years. The interest – not only in the area of Computer Science – is caused by the enormous growth of all kinds of (electronic) publications as well as by the widespread availability of advanced desktop computing technology and network connectivity.

Electronic documents – particularly those consisting of many different media types like text, diagrams, images, 3D scenes, animations, and audio, all of them being first class citizens – are beginning to change the entire publication process in all scientific fields. With the new technologies at hand, authors and educators can now utilize animations and simulations together with a rich blend of multimedia data to explain complicated phenomena and distribute them in an unprecedented way.

Of course, in the context of Digital Libraries also the term *publication process* needs to be seen with a wider focus ranging from the classical production of a scientific paper to, for example, the modeling of a (virtual) 3D environment explaining the effects of different BRDF's on global illumination.

Speaking of geometric models it is worth mentioning that engineering disciplines for many centuries had already adopted this 'generalized publishing paradigm' with technical diagrams (though typically 2D) always being the main media of communication and documentation. Of course, most other disciplines would focus on plain text, occasionally augmented with some figures.

With respect to the development of the graphics functionality of the 'Web' we can't avoid drawing a parallel to the development of the technological and functional predecessor – the Videotex systems starting with *PRESTEL* and developing into the so called European *CEPT systems* of the mid 80's [CEPT, 1981, ECMA, 1984]. As with the 'Web' early Videotex systems would only support text and the extremely crude graphics (so called *alpha-mosaics* or, in the advanced version, DRCS – Dynamically Redefinable Character Sets) which were based on dedicated fonts encoded according to ISO 2022. It took several years before the need for graphical representations composed of geometric primitives – offering 'unlimited' resolution due to the geometric specification – has been widely acknowledged [CEPT, 1987, Maurer, 1984, Fellner and Posch, 1987].

In almost the same way HTML started off with raster images being the only way to deliver graphical information to the consumer of a web page. Admittedly, the concept of plug-ins for viewers would allow geometric data like CGM [ISO, 1987] to be included from almost the very beginning. But the community would only see JPEG [ISO-CCITT, 1990] and GIF [CompuServe, 1987] images – the latter now basically replaced by PNG[1] images – for many years.

Geometrically encoded graphics took as long as the year 2000 to arrive with the formal approval of *Scalable Vector Graphics* SVG[2] by the Web Consortium. Now, graphical content can be encoded by information providers in a resolution independent manner and, as for technical drawings, in a much more compact way compared to the equivalent raster representation. Of course, there is a demand for both representations: raster images, particularly when encoded in a JPEG fashion, work best for photographs of natural environments and algebraic/semantic encodings will outperform other techniques when applied to predominantly geometric content, e.g. a digital library holding patent data.

2 3D Graphics

To illustrate the tight coupling between Computer Graphics and Digital Libraries as well as the contributions of Computer Graphics to the Digital Library development we can just look at the three different modeling paradigms:

- In the *polygon based approach* surfaces of geometric objects are approximated by meshes of planar polygons (which need to be pre-processed with level-of-detail and mesh-decimation algorithms before they become usable).
- The *functional approach* refers to all modeling techniques preserving the semantic information of the objects thereby providing a semantic level-of-detail. Examples are algebraic surfaces, plant modeling based on procedural Lindenmeyer systems, Constructive Solid Geometry (CSG) and generative modeling.

[1] http: www.w3.org/Graphics/PNG
[2] http: www.w3.org/TR/SVG

- The *lightfield approach* allows the display of 3D objects based on a set of 2D images but without the actual 3D model available at the time of rendering.

All three techniques, are not only at the heart of major research and development activities within the core Computer Graphics area they are also key technologies to make Digital Libraries usable. In return, Digital Libraries provide a new framework to challenge Computer Graphics with a new level of complexity and user interface quality.

A fundamental problem of all application areas heavily relying on 3D techniques is the drastic increase in model complexity we are currently facing. Consequently, this is also the most challenging task for the smooth integration of 3D objects into digital libraries.

Limiting factors are the sheer size of the models as well as the enormous model complexity resulting from highly nested and detailed constructions. Local and global variations as a result of dynamic computations on the models add an additional and, quite typically, significant load.

2.1 Polygon-based Approach

Particularly the demand for 3D models from real-world objects has driven the development of efficient techniques and workflows for 3D scanning, model acquisition based on discrete scanner inputs and the interactive transmission of very complex polygon meshes. These goals also define the R&D directions to be pursued in order to bring 3D models in large volumes to everybody's desktop over standard communication links.

Figure 1: *Progressive transmission of a complex 3D object (the Stanford Buddha) created from multiple 3D scans. Initial representation with 3,774 triangles (left), intermediary representation with 30,392 triangles (middle), complete mesh consisting of 1,087,716 triangles (right). (courtesy L. Kobbelt, R. Schneider, H-P. Seidel; RWTH Aachen and MPI Saarbrücken)*

Creating 3D models in large volumes at affordable costs implies the need for a fully automated and complete scanning tools, not only capturing the 3D geometry but also the surface texture and reflection behavior. And working with these models at interactive speed and transmitting them over standard communication links makes effective compression a absolute must.

Triangle Meshes are a well established component for such an environment because of the widely existing hardware support in rendering triangle sets and because of the implicit robustness of triangles compared to other types of polygons or polynomials of higher degree.

Making 3D models accessible over the internet introduces two limiting factors: bandwidth and local rendering speed of the client computer. So will the complete transmission of an aircraft engine with approx. 25,000 parts yielding 100 Mio triangles over an ISDN line approx. take 6.5 days! Not to mention that an average PC at home might not be prepared to handle the 100 Mio triangles at interactive speed.

Both limitations can be avoided by encoding the mesh in a *hierarchical representation*, similar to image pyramids for raster images. There, soon after the first bits of information have arrived the first, admittedly crude, version of the picture becomes visible. In the same way, a *Progressive Mesh (PM)* [Hoppe, 1996] first transmits only a coarse approximation of the mesh and refines it continuously until the original mesh becomes available at the receiving end. Only a small percentage of the total information volume is sufficient to present the model in a clearly recognizable way (Fig. 1) and, despite the hierarchical transmission, the total amount of transmitted information does not increase compared to the transmission of the original mesh.

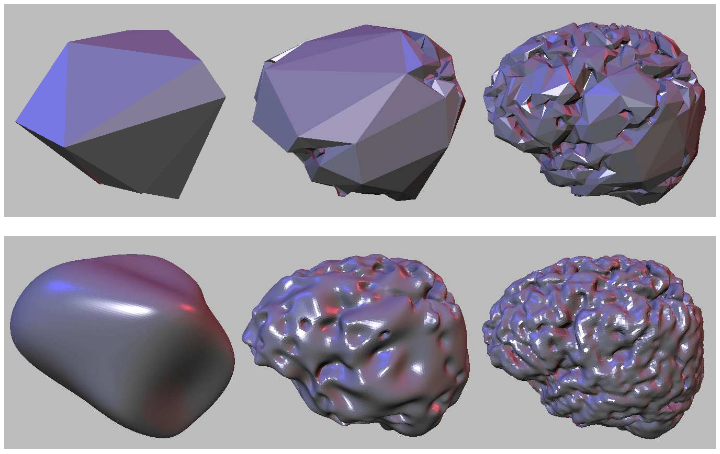

Figure 2: *Top row: progressive transmission of a brain dataset based on PM format. Bottom row: progressive transmission of same dataset based on Wavelets. (courtesy L. Kobbelt, L. Vorsatz, U. Labsik, H-P. Seidel; RWTH Aachen and MPI Saarbrücken)*

Advanced techniques for the creation of a PM out of a given mesh are typically of incremental nature collapsing edges and/or vertices to produce a coarse initial mesh stored in a standard format plus the detail information to refine the initial mesh back to its original shape [Kobbelt et al., 1998]. Alternatively, wavelet techniques based on Polynomial Splines and Subdivision Surfaces [Dæhlen et al., 2000, Labsik et al., 2000] have been developed to cater for a smooth representation of the model at early stages of the transmission process [Kobbelt et al., 1999] (Fig. 2).

With all the advances on PM's in the past years there are still a number of issues to be resolved. The most critical issue refers to PM's transmitted over networks were packets might get lost. Current hierarchical transmission schemes simply cannot copy with packet losses and make a complete retransmission necessary. Another topic is the handling of dynamic modifications to models represented by PM's.

2.2 Generative Modeling

The focus of current digital library research in the area of content-based retrieval and information mining on classical text documents is on *adequate descriptions* – called metadata – of the documents' content which can then be used for further processing. In a similar way we could expect 3D models to carry metadata fully describing the (complex) construction process – in contrast to 'just' describing the surface of the object or even just an approximation to it.

3D models are documents with an extremely rich structure compared to text documents which are typically sequential in nature (links or references are typically processed one after the other). 3D documents have their information embedded in 3D space and, consequently, retrievals will address spatial distributions (which parts are located where?) and correlations as well as semantic content (which sub-parts form the model?). To answer these retrievals effectively is a measure for the suitability and expressiveness of the chosen representation. Of course, it will also be a major criterion for the full integration of 3D objects into standard Digital Libraries.

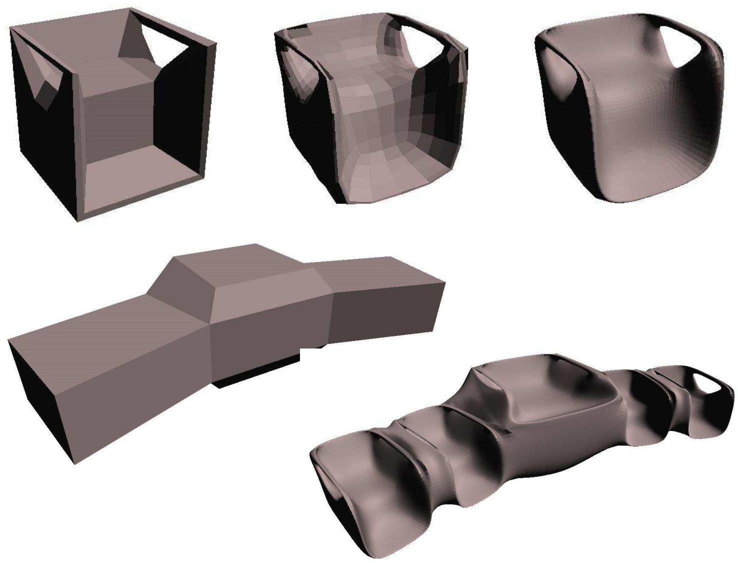

Figure 3: *Generative Modeling: Objects become* tools

Based on the results of our involvement in the V^3D^2-Initiative[3] we are now convinced that a major step forward can only be achieved by the *combination* of

- a new approach for object **Modeling and Description** together with
- an efficient and **Hierarchical Scene Structure**.

Figure 3 illustrates the concept of generative modeling [Snyder, 1992,Havemann, 1999]. The top row shows the model of a chair (in different levels of smoothness) which has been created by a series of modeling operations. These operations can then be combined to a *single* operation, say 'chair'. The bottom row of Fig. 3 shows an elongated structure which is converted into a row of five chairs by applying the new operation 'chair' five times to the cubical elements of the basic structure. The description of the new object 'row of chairs' only consists of the description of the elongated structure, the description of the tool 'chair' and five lines specifying the location where the tool 'chair' has to be applied.

The three different levels of smoothness in the top row and the two levels in the bottom row are the result of increasing refinement levels of the underlying modeling primitives - the subdivision surfaces. *Subdivision surfaces* can be used to define freeform surfaces over irregular control point meshes and introduce discontinuities like spikes or creases in a controlled way. Moreover, they help to reduce freeform modeling to polygonal modeling, as any polygonal mesh can be used as control polygon. Thereby, they drastically reduce the degrees of freedom (DOFs) in freeform modeling and consequently fit very well to a generative modeling framework. The advantage of using subdivision surfaces in this context is twofold: it reduces the degrees of freedom and it allows adaptive tesselation during an interactive session. In contrast to polygonal models, subdivision surfaces do not suffer from an approximation quality that is a priori limited. Instead, they can be tesselated on demand to any resolution needed, which is essential for high-fidelity close-ups.

The basic mesh for the 'chair' consists of 38 elements (top left). Refining it 4 times yields a mesh with 9728 elements (top right). The basic mesh for the 'row of chairs' consists of 198 elements resulting in 50,698 elements after 4 refinement steps (bottom right).

Many steps in the modeling process are repeated several times with different parameters on different objects. Consequently, it is desirable to automate 3D modeling using some form of geometric programming language. When a user can specify variables and functions in a geometric program to let object parameters be computed automatically, even *dynamic* models become possible. Furthermore, programmed models have a different space-time tradeoff: When low-level primitives are generated only on demand from higher-level descriptions, space is traded for model-evaluation complexity. With a geometric modeling language that permits compact and comprehensive descriptions of very detailed models, this model description has to be quickly translated to OpenGL primitives at *runtime*. Consequently, more work needs to be invested into *language-based 3D-modeling* and into *efficient model evaluation and visualization*. In fact, this approach initiates a paradigm change from traditional object-based modeling

[3] `http://graphics.tu-bs.de/V3D2`

to function-based, i.e. generative, modeling. Objects are not described in terms of triangles anymore, but merely in terms of the function sequence which was used to generate it.

The second aspect, the *Hierarchical Structuring*, is illustrated in Figure 4. It shows different levels of the computed hierarchy of an industrial dataset (the interior of an Airbus model)[4]. The automatically created hierarchy provides all relevant information on the spatial distribution of the geometric primitives, be it elementary triangles or complex compound-objects like the generative model from Fig. 3. Most notably, this is information is already available fairly high up in the hierarchy, i.e. at a coarse spatial resolution.

The different Levels-of-Detail (LOD) – in Fig. 4 visualized as bounding boxes – are created on the fly providing enough information to recognized the structures even at a very low refinement level. The top-left image (at refinement level 10 of the binary structure) already shows structural details like chairs. And two refinement levels more (image top center) start to exhibit details like arm and back rests. The number of bounding volumes at that level is less than 4,000 or less than 2% of the total of geometric primitives in this model.

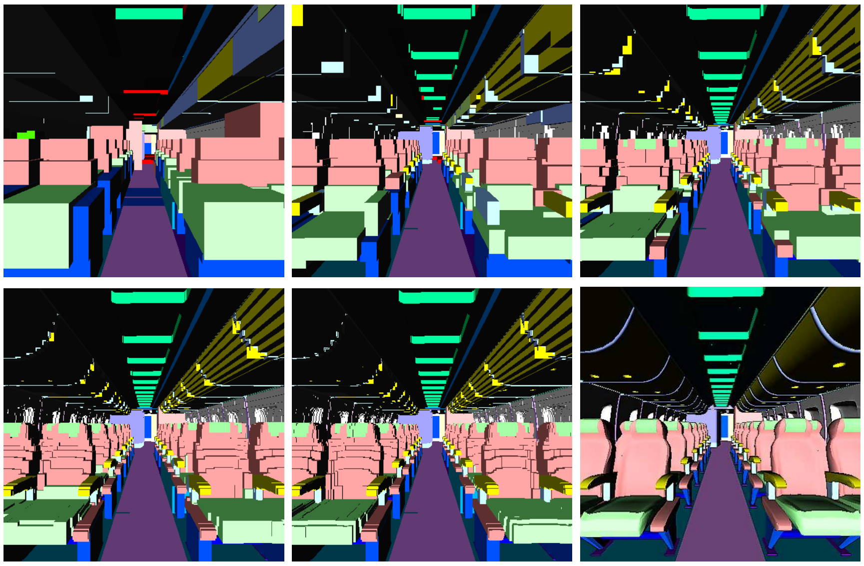

Figure 4: *Visualization of the computed hierarchy of bounding volumes for a complex model (interior of an airplane). The top row shows the interior nodes of the binary tree for the levels 10, 12, and 14 (left to right). The bottom row shows the interior nodes for the levels 16 and 18 followed by a Gouraud shaded image of the original geometry (left to right).*

The resulting hierarchy serves for practically all rendering tasks from interactive exploration of large data sets on standard computing and standard graphics equipment to

[4]courtesy LightWork Design Ltd.

high quality rendering like ray tracing, radiosity or photon maps [Fellner et al., 1998, Müller and Fellner, 1999, Müller et al., 2000].

2.3 Image Based Rendering

The last approach for the model creation presented here is radically different to the other two discussed so far. Based on the idea that each photo is (by definition) a photorealistic rendering of the object the lightfield rendering builds on photos as the elementary rendering primitive.

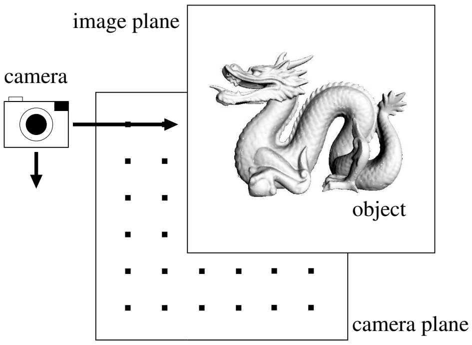

Figure 5: *Creating the Lightfield data structure: The Lightfield consists of a set of pictures of the object, resulting from a regular sampling of the camera plane. (courtesy I. Peter and W. Straßer; Univ. Tübingen)*

A *Lightfield* [Levoy and Hanrahan, 1996] or *Lumigraph* [Gortler et al., 1996] stores a set of pictures of an object in a special data structure. The pictures each hold a rectangular region of the image plane through which the object can be seen from different viewing angles. As illustrated in Fig. 5 the set of pictures is taken by regularly sampling the camera plane (which is parallel to the image plane) with a camera. This results in a 2D sampling array of 2D data values (the pictures), i.e., a four-dimensional lightfield which can be used to reconstruct arbitrary views of the object, provided the new viewing position is within or close to the viewport of the camera plane through which the pictures have been taken.

The advantage of this approach is its generality as is only relies on the photographic data of the object. No modeling or geometric object reconstruction steps are involved in acquiring an object's lightfield.

The obvious disadvantages are the lack of surface characteristics or photogrammetric parameters (the pictures are taken under one particular illumination), the lack of geometric information, and the size of the lightfield. Even a low-resolution lightfield of 32×32 with an image resolution of 256×256 with 24 bit color occupies 192 MB if stored naively and approx. 9 MB if stored with conventional compression techniques like vector quantization.

To make this technique usable in the context of distributed digital libraries new and problem-specific approaches for the compression and transmission of lightfields need to be developed. Another research challenge in this area is the integration of light-fields into classical scenes consisting of explicit geometry like triangle meshes and/or semantic models.

3 Conclusion

The main message of this presentation is the need to tightly integrate graphical information in its 2D *and* 3D form into electronic documents in order to make digital libraries a versatile technology with the potential to include many application fields currently considered as stand-alone domains. The paper motivates a 'generalized view' on the term *document* as a collection of various media types in one compound structure and, further to presenting recent contributions from the field of Computer Graphics, raises several issues which should stimulate further research work in the graphics field to make Digital Libraries of the future more accessible.

The first experiences with this approach reported here are based on the results of a Strategic Initiative named *Distributed Processing and Delivery of Generalized Digital Documents (V^3D^2)* (see also http://graphics.tu-bs.de/V3D2 funded by the German Research Foundation (DFG) to address basic research challenges in the field of Digital Libraries.

Acknowledgements

The support from the German Research Foundation (DFG) for V^3D^2 Initiative in general and from the cooperating core graphics projects within this initiative in particular is gratefully acknowledged. Special thanks go to Sven Havemann and Gordon Müller for leading the research on generative modeling and hierarchical structuring.

References

[CEPT, 1981] CEPT (1981). *Videotex Presentation Layer Data Syntax (Issue 1) T/CD 06-01*. CEPT, Innsbruck, Austria.

[CEPT, 1987] CEPT (1987). *Videotex Presentation Layer Data Syntax (Issue 2); Part 2 – Geometric Display*. CEPT.

[CompuServe, 1987] CompuServe (1987). *Graphics Interchange Format (GIF) – A standard defining a mechanism for the storage and transmission of raster-based graphics information*. CompuServe Inc.

[Dæhlen et al., 2000] Dæhlen, M., Lyche, T., Mørken, K., Schneider, R., and Seidel, H.-P. (2000). Multiresolution analysis over triangles based on quadratic Hermite interpolation. *Journal of Computational and Applied Mathematics*, (119):97–114.

[ECMA, 1984] ECMA (1984). *Graphics Virtual Device Presentation Layer Protocol Syntax, TG PC N44 Rev 3*. ECMA, Geneva.

[Fellner et al., 1998] Fellner, D. W., Havemann, S., and Müller, G. (1998). Modeling of and navigation in complex 3D documents. *Computers & Graphics*, 22(6):647–653.

[Fellner and Posch, 1987] Fellner, D. W. and Posch, R. (1987). Bildschirmtext – an open videotex network for text and graphic applications. *Computers & Graphics*, 11(4):359–367.

[Gortler et al., 1996] Gortler, S. J., Grzeszczuk, R., Szeliski, R., and Cohen, M. F. (1996). The Lumigraph. In *Proc. SIGGRAPH '96*, pages 43–54.

[Havemann, 1999] Havemann, S. (1999). Effizienter austausch von 3d dokumenten auf basis von generativer modellierung. In Engels, G. and Schäfer, W., editors, *INFORMATIK 99, Paderborn*. Springer.

[Hoppe, 1996] Hoppe, H. (1996). Progressive meshes. In *Proc. SIGGRAPH '96*, pages 99–108. ACM.

[ISO, 1987] ISO (1987). *Information Processing Systems – Computer Graphics – Metafile for the Storage and Transfer of Picture Description Information (CGM), IS 8632*. ISO.

[ISO-CCITT, 1990] ISO-CCITT (1990). *JPEG Technical Specification, Revision 5; ISO JTC1/SC2/WG8 JPEG-8-R5*. ISO-CCITT.

[Kobbelt et al., 1998] Kobbelt, L., Campagna, S., and Seidel, H.-P. (1998). A general framework for mesh decimation. In *Proc. Graphics Interface '98*, pages 43–50.

[Kobbelt et al., 1999] Kobbelt, L., Vorsatz, J., Labsik, U., and Seidel, H.-P. (1999). A shrink wrapping approach to remeshing polygonal surfaces. *Computer Graphics Forum*, 18:119–130.

[Labsik et al., 2000] Labsik, U., Kobbelt, L., Schneider, R., and Seidel, H.-P. (2000). Progressive transmission of subdivision surfaces. *Computational Geometry Journal*, 15:25–39.

[Levoy and Hanrahan, 1996] Levoy, M. and Hanrahan, P. (1996). Light Field Rendering. In *Proc. SIGGRAPH '96*, pages 31–42.

[Maurer, 1984] Maurer, H. (1984). The austrian approach to videotex. *Cybernetics and Systems Research*, 2:589–592.

[Müller and Fellner, 1999] Müller, G. and Fellner, D. W. (1999). Hybrid scene structuring with application to ray tracing. In Mudur, S. P., Shikhare, D., Encarancao, J. L., and Rossignac, J., editors, *Intl. Conf. on Visual Computing ICVC '99*, pages 19–26, Goa, India.

[Müller et al., 2000] Müller, G., Schäfer, S., and Fellner, D. W. (2000). Automatic creation of object hierarchies for radiosity clustering. *Computer Graphics Forum*, 19(4):213–221.

[Snyder, 1992] Snyder, J. M. (1992). *Generative Modeling for Computer Graphics and CAD*. Academic Press, San Diego, CA.