Zuwendungsempfänger: **Förderkennzeichen:**
Uni Bonn 01 IRA 02D

Textbericht zum BMBF Projekt OpenSG PLUS

Berichtszeitraum: **1.2.2001-31.12.2003** / Berichtstermin: 30.6.04

1. Aufgabenstellung

In diesem Projekt sollten von der Universität Bonn zum einen Techniken und Verfahren effizienten Rendering großer Szenen mit OpenSG PLUS (Large Scene Support) und zum anderen Methoden zur Visualisierung von NURBS-Flächenverbänden (High Level Primitives) entwickelt werden. Dabei konzentrieren wir uns auf Methoden zur LOD-Auswahl für Objekte im Szenengraphen, auf das blickpunkts- und beleuchtungsabhängige Rendering von Objekten, auf die Erzeugung adaptiv verfeinerbarer Multiresolutionmodelle für unstrukturierte Dreiecksnetze sowie die adaptive „On-the-Fly“ Approximation von NURBS-Flächenverbänden durch Dreiecksnetze.

2. Voraussetzungen

Bereits während seiner Dissertation [K95] beschäftigte sich der Antragsteller intensiv mit der adaptiven Approximation und der Echtzeitvisualisierung von NURBS-Flächenverbänden. Basierend auf zweidimensionalen Delaunay-Triangulierungen wurde eine Zeit- und speichereffiziente progressive Darstellung für NURBS-Flächen entwickelt [K94, KS95, KS96], die sehr erfolgreich auch für Terrainvisualisierung [KCH97] eingesetzt wird.
Daneben entwickelten wir eine Reihe von Netzdezimierungsalgorithmen. Der wichtigste Beitrag zu diesem Thema war sicherlich die Einführung des Haussdorff-Abstands zur Kontrolle des Fehlers zwischen approximierendem Netz und dem Orginal [KLS96]. Darüber hinaus entwickelten wir Datenstrukturen und Algorithmen zur selektiven Verfeinerung polygonaler Netze [KK97,K98]. Die Projektion des Hausdorff-Abstands auf den Bildschirm erlaubt das selektive Verfeinern von Objekten ohne Verluste in der resultierenden Bildtgeometrie. Aufbauend auf den Datenstrukturen zur selektiven Verfeinerung beschäftigten wir uns dann intensiv mit der textur- und beleuchtungsabhängigen „On-the-fly“-Verfeinerung polygonaler Netze für optimales Rendering ohne Verluste der Bildqualität [KSS98, SK98]. Schließlich kombinierten wir diese Verfahren mit den oben beschriebenen Verfahren zur progressiven Darstellung von NURBS-Flächenverbänden zur „On-the-fly“-Visualisierung eines CAD-Modells der A-Klasse von Daimler Benz [G98].
Alle Resultate sind durch zahlreiche Publikationen in Konferenzbeiträgen und Journals dokumentiert.

3. Planung und Ablauf

Aufgrund der Bedeutung von NURBS für das Projekt, wurde bereits auf dem OpenSG Kick-Off-Meeting einstimmig beschlossen, die Arbeiten zum Tesselieren und Rendern von NURBS vorzuziehen und schon am Anfang des Projekts damit zu beginnen. Bis auf diese Änderung wurden alle Arbeiten dem ursprünglichen Projektplan gemäß durchgeführt.

4. Wissenschaftlicher und technischer Stand

Der wissenschaftliche und technische Stand zu Beginn des Projektes stellte sich wie folgt dar:

Szenengraph und Level-Of-Detail Rendering

Durch Verwendung unterschiedlicher Level of Detail (LOD) konnten selbst komplexe Objekte in Echtzeit gerendert werden. Seit der Einführung von LODs wurden immer bessere Methoden entwickelt: Diskrete LODs, einfache Progressive Darstellungen und schließlich blickpunktabhängige progressive LODs.
Während die verwendete Approximation von Objekten sich bei blickpunktsabhängigen progressiven LODs durch das resultierende Bild steuern läßt, gab es bis zum Beginn des Projektes für große Modell nur Verfahren, in denen ein einziges, statisches Modell, wie z.B. Terrain, dargestellt wurde. Für virtuelle Umgebungen ist jedoch die Kombination von blickpunktsabhängigen dynamischer LOD-Modellen mit einem hierarchischen Szenengraphen notwendig. Erste Ansätze zum intelligenten Aufteilen einer maximal pro Zeiteinheit darstellbaren Zahl von Dreiecken auf mehrere Objekte wurden schon 1993 von Funkhouser und Sequin [FS93] entwickelt. Dieses Modell wurde später noch von Mason und Blake

verbessert [MB97], wie es jedoch für progressive oder selektiv verfeinerbare Modelle angewendet werden kann war nicht klar. Zusammenfassend lässt sich sagen, dass die damals vorgeschlagenen und später auch noch verbesserten Methoden noch nicht ausgereicht haben, um blickpunktsabhängige dynamische LOD-Modelle gewinnbringend mit einem Szenengraphenkonzept zu kombinieren. Es gab auch keine Methoden zur Kombination von progressiven Techniken mit „View-Frustum Culling“ oder mit Verdeckungsrechnung.

Effiziente Datenstrukturen für polygonale Netze

Der Durchsatz bei der Verarbeitung von Geometriedaten hängt wesentlich von den zugrundeliegenden Datenstrukturen ab. Insbesondere bei polygonalen Netzen muss der effiziente Zugriff auf einzelne Elemente und das lokale ”Navigieren” (Nachbarschaftssuche) gewährleistet sein.
Die meisten in der Literatur beschriebenen Datenstrukturen erlaubten nur die Repräsentation von Zweimannigfaltigkeitsnetzen [B72,M88]. Obwohl die verwendeten Oberflächennetze „fast überall“ Zweimannigfaltigkeiten sind, gibt es in den meisten Netzen immer Ausnahmepunkte und –Kanten, deren Nachbarschaftsbeziehungen zu anderen Punkten, Kanten und Flächen in Zweimannigfaltigkeitsdatenstrukturen nicht gespeichert werden können. Ein Ausweg ist, Datenstrukturen für Dreimannigfaltigkeitsnetze, wie z.B. die Radial Edge Struktur [W85], oder die Simplexdatenstruktur [PH97] zu verwenden. Leider ist die Erzeugung, Pflege und Traversierung dieser Strukturen sehr viel komplizierter, als die der Zweimannigfaltigkeitsdatenstrukturen, was sich in der Effizienz der auf diesen Datenstrukturen implementierten Algorithmen niederschlägt. Darüber hinaus benötigen Dreimannigfaltigkeitsdatenstrukturen einen wesentlich höheren Speicherplatz.
Obwohl einige Arbeiten sich mit dem Problem der Speicherung zusätzlicher Geometrie- und Texturattribute befasst haben [CK98], war auch in diesem Bereich noch mehr Forschungsaufwand notwendig, um maximal flexibel zu sein.
Für effiziente Algorithmen auf polygonalen Netzen mussten daher neue Datenstrukturen entwickelt werden, die anwendungsspezifisch den effizienten Zugriff auf einzelne Elemente bei möglichst geringem Speicherbedarf ermöglichen. Dies erreicht man durch eine optimale Abstimmung der gespeicherten Referenzen innerhalb der Datenstruktur auf das in der jeweiligen Anwendung zu erwartende Zugriffsprofil.

On-the-fly Erzeugung blickpunktabhängiger LODs

Progressive Darstellungen von Dreiecksnetzen stellen eine Sequenz unterschiedlicher Level of Detail von Objekten zur Verfügung. Für unstrukturierte Dreiecksnetze basiert die Berechnung der unterschiedlichen LODs in der Regel auf der Anwendung lokaler Vereinfachungsoperationen [H96, PH97, K94, KCH98, KS95, KLS96, KSS98a, GH97, RR96, RB93, SLZ92]. Die Inversen der lokalen Vereinfachungsoperationen definieren auf kanonische Weise eine lineare Folge von LOD Approximationen der Objekte. Diese Folge inverser Operationen erlaubt progressive Übertragung, auch progressives Streaming genannt.
Häufig sind schon progressive Darstellungen für Renderingzwecke ausreichend. Meist jedoch besitzen nicht alle Details dieselbe visuelle Bedeutung. Die Möglichkeit, mit einem groben Modell zu starten und dann auf visuell wichtige Teile zu fokussieren, heißt selektive Verfeinerung. Selektive Verfeinerung wird vor allem dazu verwendet, redundante Teile, welche außerhalb des Sichtvolumens, auf Objektrückseiten, weit entfernt vom Beobachter liegen oder teilweise verdeckt sind, zu vermeiden. Bei beleuchtungsabhängiger selektiver Verfeinerung werden z.B. diejenigen Teile der Objekte verfeinert, die Highlights enthalten. Selektive Verfeinerung kann auch dann angewandt werden, wenn sich die Geometrie der Objekte und damit Details zeitlich verändern.
Obwohl schon eine ganze Reihe von Algorithmen zur „On-the-Fly“ Generierung selektiv verfeinerter Netze entwickelt und implementiert wurden [H97,KLS96, KK97,K98a,K98b,KS98,KS99,KSS98,LE97,XV96], waren folgende Fragestellungen noch nicht zufriedenstellend bearbeitet:

- Möglichkeiten zur Distanz- und Texturabhäniger Verfeinerung während des Rendering
- Modelle zur Beleuchtungsabhängigen Verfeinerung, hier insbesondere die Kombination mit Normalenmaps
- Erzeugung und Einsatz von Displacement-Maps

Hinzu kam, daß bis zu Projektbeginn bekannte progressive LODs die Verwendung von Triangle-Strips, die für schnelles Rendering unabdingbar sind, stark einschränkten. Darüber hinaus gibt es praktisch keine Verfahren, blickpunktsabhänige progressive LODs für dynamische und deformierbare Objekte „On-the-fly“ zu erzeugen.

NURBS

„Trimmed Nurbs“ sind die Standardrepräsentation im CAD/CAM Bereich. Große Modelle bestehen aus zehntausenden von einzelnen Patches, die entlang sogenannter Trimmingkurven aneinandergesetzt sind. Algorithmen, die in einem Vorverarbeitungsschritt die NURBS-Patches durch statische Dreiecksnetze approximieren, sind ungeeignet, wenn der Betrachtungsabstand stark variiert. Schon relativ früh wurden deshalb Algorithmen zur schnellen, möglichst echtzeitfähigen adaptiven Approximation von getrimmten NURBS-Patches entwickelt [RHD89, AS91]. Während diese Algorithmen sich darauf beschränkten, die einzelnen Patches hinreichend genau adaptiv abzutasten, versuchten spätere Arbeiten das Problem der T-Vertices entlang der Trimmingkurven benachbarter Patches zu lösen [K94,KS95,KM97]. Offen blieb die Frage, wie man mit Trimmingkurven umgeht, die so klein sind, dass ihre Projektionen auf den Bildschirm in den Bereich von einem oder zwei Pixeln fallen. Erste Schritte zur Beantwortung dieser Fragestellungen werden in [KS96, KMZH97, G98] gemacht. Alle diese Ansätze versuchen mehrere Techniken, wie Splineflächensimplifizierung, Parallelverarbeitung und statische Level of Detail Simplifizierung zu kombinieren, um effizientes Rendering zu ermöglichen. Alle der bisher vorgestellten Algorithmen konzentrieren sich jedoch auf die Lösung eines speziellen Problems und keine der Techniken fand Eingang in ein Szenengraph API.

Referenzen

[AS91] S.S. Abi-Ezzi and L.A. Shirman, *Tessellation of curved surfaces under highly varying transformations*, Proceedings of Eurographics, pages 385-397, 1991

[B72] B.G. Baumgart, *Winged-edge polyhedron representation*, Technical report, STAN-CS-320, Stanford University, 1972

[CV96] J. Cohen, A. Varshney, D. Manocha, G. Turk, H. Weber:; P. Agarwal, F. P. Brooks, Jr., and W. Wright, Simplification envelopes, SIGGRAPH 96, 1996

[CC99] P. Cignoni, C. Montani, C. Rocchini, R. Scopigno, M. Tarini, *Preserving attribute values on simplified meshes by re-sampling detail textures*, The Visual Computer, 1999

[CK98] Swen Campagna, Leif Kobbelt, Hans-Peter Seidel, *Directed Edges - A Scalable Representation for Triangle Meshes*, Journal of Graphics Tools, 1998.

[FS93] T. Funkhouser, C. Sequin, Adaptive Display Algorithm for Interactive Frame Rates During Visualisation of Complex Virtual Environments, SIGGRAPH ’93, 1993

[G98] Heiko Goller, *Multiresolutionmodelle von CAD-Daten*, Diplomarbeit, WSI-GRIS Universität Tübingen, 1998

[GH97] M. Garland and P. Heckbert, *Surface Simplification Using Quadric Error Metrics*, SIGGRAPH '97, 1997

[H96] H. Hoppe, *Progressive meshes*, SIGGRAPH ’96, 1996

[H97] H. Hoppe, *View-dependent refinement of progressive meshes*, SIGGRAPH ’97, 1997.

[K94] Reinhard Klein, *Linear approximation of trimmed surfaces*, The Mathematics Of Surfaces VI, pages, 1994

[K95] Reinhard Klein, *Netzgenerierung impliziter und parametrisierter Flächen in einem objektorientierten System*, PhD thesis, WSI/GRIS, 1995

[K98a] Reinhard Klein, *Multiresolution representations for surfaces meshes*, Computer & Graphics, 1998 (Received Computer & Graphics Best Paper Award 1998)

[K98b] R. Klein, *Multiresolution surface modeling*, Conference on Numerisation 3D, 1999

[KCH98] R. Klein, D. Cohen-Or, and T. Hüttner, *Incremental view-dependent multilevel triangulation of terrain*, Visual. Comput. Animat., 1998

[KCS98] L. Kobbelt, S. Campagna, H-P. Seidel, *A general framework for mesh decimation*, Graphics Interface 1998

[KG98] R. Klein, S. Gumhold, *Data compression of multiresolution surfaces*, Eurographics Workshop on Visualization in Scientific Computing, 1998

[KK97] R. Klein, J. Krämer, *Multiresolution representations for surface meshes*, Spring Conference on Computer Graphics, 1997

[KLS96] R. Klein, G. Liebich, and W Strasser, *Mesh reduction with error contro*l, Visualization 96, 1996.

[KM97] S. Kumar, D. Manocha, H. Zhangh, K. Hoff, *Accelerated Walkthrough of Large Spline Models*, Interactive 3D Symposium, 1997

[KS95] R. Klein and W. Strasser, *Large mesh generation from boundary models with parametric face representation*, SIGGRAPH Symposium on Solid Modeling, 1995

[KS97] R. Klein, W. Strasser, *Generation of multiresolution models from cad - data for real time rendering*, Theory and Practice of Geometric Modeling, 1997

[KS98] R. Klein, A. Schilling, *Texture Tiles: Efficient Multiresolution Model for textured surfaces*, Technical Report, Universität Tübingen, 1998
[KS99] Reinhard Klein, Andreas Schilling, *Efficient rendering of multiresolution meshes with guaranteed image quality*, Visual Computer 1999
[KSS98] R. Klein, A. Schilling, W. Strasser, *Illumination dependent refinement of multiresolution meshes*, Computer Graphics International 98, 1998 (Auszeichnung mit dem CGI-Young Researchers Award)
[KSS98a] L. Kobbelt, S. Campagna, H-P. Seidel, *A general framework for mesh decimation*, Graphics Interface 98 Proceedings, 1998
[LE97] D. Luebke and C. Erikson, View-dependent simplification of arbitrary polygonal environments, SIGGRAPH 97, 1997
[M88] Martti Maentylae, *An Introduction to Solid Modeling*. Computer Science Press, 1988
[MB97] Ashton E. W. Mason, Edwin Blake, *Automatic Hierarchical Level of Detail Optimization in Computer Animation*, EUROGRAPHICS'97, 1997
[N99] NVIDIA Corporation. *NVIDIA OpenGL Extension Specifications*, 1999 Available from http://www.nvidia.com.
[PH97] Jovan Popovic, Hugues Hoppe, *Progressive simplicial complexes,* SIGGRAPH 97, 1997
[RH87] A. Rockwood, K. Heaton, T. Davis, Real-time rendering of trimmed surfaces, SIGGRAPH, 1989
[RR96] R. Ronfard, J. Rossignac, *Full-range approximation of triangulated polyhedra*, Computer Graphics Forum, 1996
[RB93] J. Rossignac, P. Borrel, *Multi-resolution 3d approximation for rendering complex scences*, Modeling in Computer Graphics: Methods and Applications, 1993
[SK98] A. Schilling and R. Klein, *Rendering of multiresolution models with texture*, Computer & Graphics, 1999
[SZL92] W. J. Schroeder, J. A. Zarge, W. E. Lorensen, *Decimation of triangle meshes*, SIGGRAPH '92, 1992.
[W85] Kevin Weiler, *Edge-Based Data Structures for Solid Modeling in Curved-Surface Environments*, Computer Graphics and Applications, 1985
[XV96] J. C. Xia, A. Varshney, *Dynamic view-dependent simplification for polygonal models*, Visualization '96, 1996.

5. Zusammenarbeit

Die wichtigste Verbindung zu anderen Projekten war die gemeinsame Spezifikation der Geometriedatenstruktur sowie deren Implementierung und Weiterentwicklung. Darüber hinaus arbeiteten wir an der Spezifikation von Hierarchien mit.
Die intensivste Zusammenarbeit im Bereich Geometriedaten mit andern Partnern im Projekt bezog sich auf die Einbindung der OpenMesh Datenstruktur in das bestehende OpenSG Framework, besonders im Hinblick auf LOD Darstellungen triangulierter Modelle und die im Teilprojekt 3.3 implementierte on-the-fly LOD Generierung getrimmter NURBS Modelle.
Weitere Kooperationen bezogen sich auf das für große Szenen notwendige Occlusion Culling. Hier ging es darum, das an der Uni Tübingen entwickelte Verfahren mit der LOD-Hierarchie zu kombinieren.
Im Zusammenhang mit Methoden zum „Appearance Preserving" Rendering von NURBS Flächen wurde auf Methoden aus TP 4 „High Level Shading" zurückgegriffen.
Die ingesamt umfangreichste Zusammenarbeit fand mit den Entwicklern der Kernfunktionalität von OpenSG statt, sowohl hinsichtlich der Definition von Anforderungen als auch um die neu entwickelten Methoden in die Szenengraph API zu integrieren.

6. Ergebnisse

Sowohl in den Bereichen „Level of Detail" und „Large Scene Support" als auch im Bereich der „High-Level Primitive" entstanden wichtige, neue Beiträge, die sich zum einen auf neue Techniken zum Out-of-Core-Rendering und zum anderen auf ein verbessertes Rendering von NURBS-Modellen beziehen. Im Einzelnen lassen sich die Beiträge den von der Universität Bonn bearbeiteten Teilprojekten wie folgt zuordnen:

Teilprojekt 2.3 (LOD-Selektion)

- Entwicklung und Implementierung richtungs- und entfernungsabhängiger Fehler-abschätzungen zur optimalen Auswahl von LODs
- Implementierung von View-Frustum- und Backface-Culling Techniken

- Implementierung eines Load-Balancing Verfahrens, um durch die Abwägung zwischen der Approximationsgüte der zu rendernden Modelle und der dafür nötigen Zeiten stabilere Bildwiederholraten zu erreichen
- Verbesserung des LOD Managements für den Seam-Graphen (siehe unten)
- Unabhängige LODs auf Modellteilen durch Gap Filling
- Neue Methoden zu LOD Selektion, Prefetching und Caching für Rendering von Modellen, die nicht komplett in den Speicher geladen werden können (Out-of-Core).

Teilprojekt 3.1 (On-the-fly LODs)

- Weiterentwicklung des Algorithmus zum Erzeugen von LODs für triangulierte Modelle mit kleinen Löchern: Implementierung weiterer Simplifizierungs-Operatoren (Edge-Edge Collapse, Triangle-Vertex-Collapse), die ebenfalls topologie-verändernde Simplifizierung ermöglichen
- Implementierung von Kollisionsdetektionsverfahren zur Vermeidung von Durchdringungen bei der topologie-verändernden Simplifizierung z.B. mehrlagiger Modelle (z.B. Kleidungsstücke auf Körper)
- Weiterentwicklung und Optimierung des Algorithmus zum Erzeugen von LODs für triangulierte Modelle zur Out-of-Core Simplifizierung von komplexen Dreiecksmodellen
- Entwicklung eines neuen Verfahrens zum „Appearance Preserving“ Rendering von LOD-Modellen
- Entwicklung eines neuen Texturierungsalgorithmus für Dreiecksmodelle.

Teilprojekt 3.3 (NURBS)

- Parser zum Einlesen von getrimmten NURBS-Flächen im Inventor Format
- Verbesserung des Näh-Algorithmus zum Schließen von kleinen Lücken (Cracks) in existierenden NURBS Modellen. Die Existenz dieser Modellierungsfehler ist typisch für diese Klasse von Modellen, entsprechend hoch die Relevanz des neuen Algorithmus
- Einführung des Seam-Graphen zur LOD Generierung und LOD Selektion von getrimmten NURBS-Modellen
- Verwendung einer neuen Fehlerschranke für die Approximation der getrimmten NURBS-Flächen durch Dreiecke, die keine zwischenzeitliche Konvertierung der NURBS-Flächen in Bezier-Flächen verlangt. Dies resultiert in einer deutlich besseren Approximation der NURBS-Flächen, da im Vergleich zum alten Algorithmus deutlich weniger Dreiecke notwendig sind, um den selben Approximationsfehlers zu garantieren
- Implementierung eines schnelleren Triangulierungsalgorithmus für getrimmte NURBS Flächen
- Definition und integrative Implementierung des OSGSurface Knotens zur Repräsentierung einzelner high-level Primitive wie getrimmter NURBS oder Bézier Patches
- Vereinfachung des Vernähens durch den neuen Gap Filling Algorithmus zum Schließen von kleinen Lücken (Cracks) in existierenden NURBS Modellen unter Benutzung neuer Graphikhardware.
- Integration von Normalen Maps für bessere Bildqualität
- Reparametrisierung von NURBS Modellen für Texturmapping
- Optimierung des schnellen Triangulierungsalgorithmus für getrimmte NURBS Flächen durch verlustfreie Vereinfachung der Flächen und Trimmingkurven
- „Appearance Preserving“ Tesselation von NURBS Modellen für Anwendungen in der Qualitätssicherung
- Paxistests der Implementierung des OSGSurface Knotens zur Repräsentierung einzelner high-level Primitive wie getrimmter NURBS oder Bézier Patches.

7. Voraussichtlicher Nutzen

Da es sich bei OpenSG Plus um ein Open Source Projekt handelt, gab es keine Schutzrechtsanmeldungen. Durch die Erweiterungen des OpenSG Szenengraphen im Rahmen des OpenSG Plus Projektes wurde die API interessanter für kommerzielle Anwendungen. In einem durch OpenSG motivierten EU-Projekt (IST RealReflect, IST Project No 34744) wird OpenSG als Standardszenengraph API eingesetzt und durch weitere Funktionalitäten erweitert. Kommerzielle Partner in RealReflect nutzen OpenSG inzwischen als Szenengraph-API für ihre VR-Systemsoftware.

8. Fortschritt bei anderen Stellen

Im Bereich LOD-Selektion und Large Scene Support gab es parallel zu unseren Entwicklungen in OpenSG ähnliche Anstrengungen sowohl in Europa als auch in den USA, die zum Teil erst in 2004

veröffentlicht werden. Die dort entwickelten Methoden basieren ebenfalls wie die im Projekt entwickelten auf hierarchischen LODs, haben aber hinsichtlich Performanz und Implementierungsaufwand Nachteile. Keine dieser Methoden wurde in eine Szenengraph API integriert. Hinsichtlich NURBS Rendering gab z.B. an der Universität Erlangen Anstrengungen, die Qualität des Rendering zu verbessern und für Qualitätssicherung zu nutzen. Sie nutzen ähnliche Methoden wie wir ganz zu Anfang des Projekts [2,3,4,5]. Diese haben sich während des Projekts aber im Kontext großer und dynamischer Modelle als nur bedingt anwendbar erwiesen und wurden deshalb von uns durch neuere und wie sich herausstellte wesentlich effizientere Verfahren ersetzt [16]. Im internationalen Bereich gab es unseres Wissens keine wesentlichen Veröffentlichungen zu diesem Thema. Die Firma SGI arbeitet aber nach wie vor intensiv an diesem Thema und hatte Ende 2003 Kontakte zu uns aufgenommen und sich die Ergebnisse vorort in Bonn angesehen.

9. Veröffentlichungen

Im Rahmen des Projekts entstanden folgende Veröffentlichungen:

1. P. Borodin, R. Klein: *Progressive Meshes with Controlled Topology Modifications.* OpenSG Symposium, 2002
2. F. Kahlesz, Á. Balázs, R. Klein: *NURBS Rendering in OpenSG Plus.* OpenSG Symposium, 2002
3. F. Kahlesz, Á. Balázs, R. Klein: *Multiresolution Rendering by Sewing Trimmed NURBS Surfaces.* The 7th ACM Symposium on Solid Modeling and Applications, 2002
4. M. Guthe, J. Meseth, R. Klein: *Fast and memory efficient view-dependent trimmed NURBS rendering*. Pacific Graphics, 2002
5. M. Guthe, R. Klein: *Efficient NURBS Rendering using View-Dependent LOD and Normal Maps.* Journal of WSCG, Februar 2003
6. M. Guthe, R. Klein: *Automatic texture atlas generation from trimmed NURBS models*, Eurographics 2003, September 2003
7. S. Gumhold, P. Borodin, R. Klein: *Intersection Free Simplification*, The 4th Israel-Korea Bi-National Conference on Geometric Modeling and Computer Graphics, Februar 2003
8. Á. Balázs, M. Guthe, R. Klein: *Fat Borders: Gap Filling for Efficient View-Dependent LOD Rendering.* Technical Report No. CG-2003-2, ISSN 1610-8892, 2003
9. P. Borodin, S. Gumhold, M. Guthe, R. Klein: *High-Quality Simplification with Generalized Pair Contractions*, GraphiCon'2003
10. M. Guthe, P. Borodin, R. Klein: *Efficient View-Dependent Out-of-Core Visualization*, The 4th International Conference on Virtual Reality and its Application in Industry (VRAI'2003), Oktober 2003
11. P. Borodin, M. Guthe, R. Klein: *Out-of-Core Simplification with Guaranteed Error Tolerance*, VMV 2003, München
12. P. Degener, J. Meseth, R. Klein *An Adaptable Surface Parametrization Method*, The 12th International Meshing Roundtable 2003, September 2003
13. A. Balazs, M. Guthe, R. Klein: *Fat Borders:Gap Filling For Effcient View-dependent LOD NURBS Rendering,* Journal of Computer & Graphics, Februar 2004
14. M. Guthe, R.Klein: *Streaming HLODs: An out-of-.core viewer for network visualization of polygon models*, Journal of Computer & Graphics, Februar 2004
15. Á. Balázs, M. Guthe, R. Klein: *Efficient trimmed NURBS tessellation*, Journal of WSCG, Februar 2004
16. M. Guthe, Á. Balázs, R. Klein: *Interactive High Quality Trimmed NURBS Visualization Using Appearance Preserving Tessellation*, TCVG Symposium on Visualization, Mai 2004
17. M. Guthe, P. Borodin, Á. Balázs, R. Klein: *Real-Time Appearance Preserving Out-of-Core Rendering with Shadows*, Eurographics Symposium on Rendering, June 2004
18. M. Guthe, A. Balazs, R. Klein: *Real-time, appearance preserving out-of-core trimmed NURBS rendering*, submitted to VMV 2004, Stanford