



Introduction to Visualization and Computer Graphics

DH2320

Prof. Dr. Tino Weinkauf

***Introduction to
Visualization and Computer Graphics
and Image/Video Processing***

Introduction

DH2320
Introduction to
Visualization and Computer Graphics

DD2258
Introduction to
Visualization, Computer Graphics
and Image/Video Processing

10 joint lectures on visualization and computer graphics

6 joint homework assignments on visualization and computer graphics

2 lectures on image/video

1 lab assignment on image/video

Joint written exam on visualization and computer graphics

6 credits

7.5 credits

- Tino Weinkauf
`weinkauf@kth.se`
Lindstedtsvägen 5, Room 4420
- Office hours:
by appointment (e-mail)
- Haibo Li, `haiboli@kth.se`
Lindstedtsvägen 3, Floor 6
- Websites:
 - General overview:
<https://www.kth.se/social/course/DH2320/>
<https://www.kth.se/social/course/DD2258/>
 - Active work, assignments, discussions on Canvas:
<https://kth.instructure.com/courses/4190>

- Announcements, schedule, class material:
<https://kth.instructure.com/courses/4190>
- The lecture slides are typically available before the lecture.

- Lectures & Tutorials:

- See schedule for details, but in general:
 - 1-2 Lectures per week
 - 1 Tutorial per week
- Wednesday and Friday afternoon
- Location: mainly D3, sometimes in other places.
Check the schedule!

- You have to register for the lecture
 - Grading (exercises, exam) requires registration
 - You are welcome to just sit in and listen
 - Registration is required for credits
 - Course registration is open between 8-18 January.
 - For newly admitted students to program term 1: between 16-18 January 2018.

- In some courses at CSC there are PhD-students from other universities in Sweden following the course.
 - Since information, material and communication often is done via KTH systems and the student need a **KTH-account** to sign in these systems these students have had problems to attain the same information as the other students.
- It is possible for **external PhD-students to get a KTH-account**: Go to the reception at CSC at Lindstedtsvägen 3, level 4, where you can sign a form.
- It will only take a few days to get the account.

● Disability

- *Support via Funka*

If you have a disability, you may receive support from Funka.

<https://www.kth.se/en/student/studentliv/funktionsnedsattning>

- *Inform the teacher*

We recommend that you inform the teacher regarding any need you may have. Funka does not automatically inform the teacher.

● Funktionsnedsättning

- Om du har en funktionsnedsättning kan du få stöd via Funka:

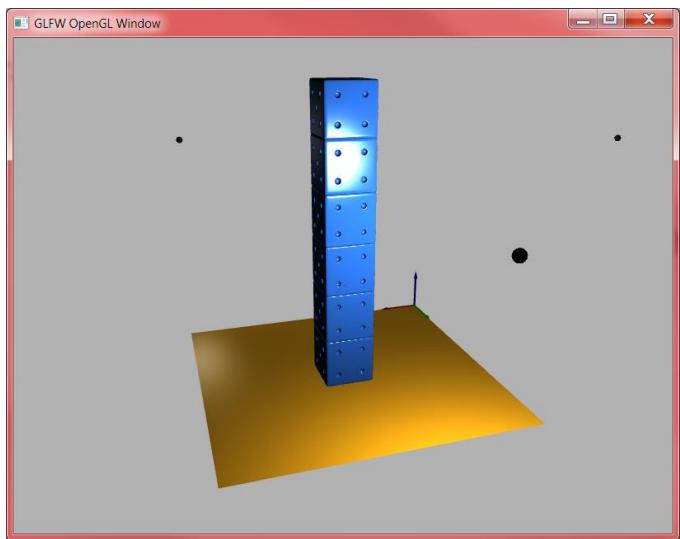
<https://www.kth.se/student/studentliv/funktionsnedsattning>

Informera dessutom kursledaren om du har särskilda behov. Visa då upp intyg från Funka.

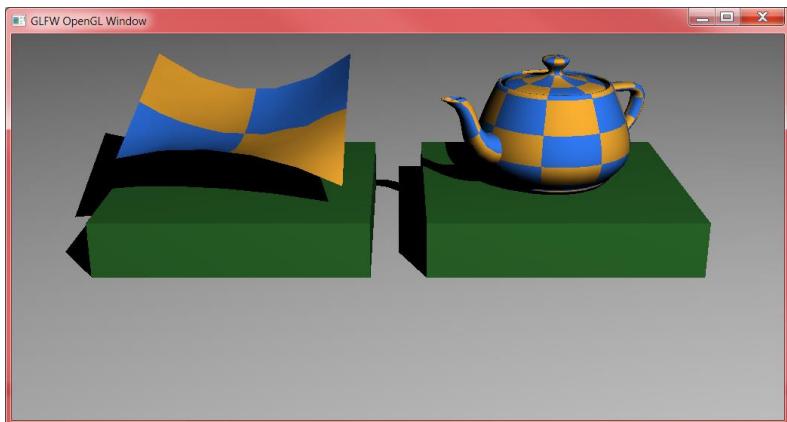
- To pass the lecture, you need to...
 - Work on **all homework assignments**
 - Obtain at least **50%** of the assignments score
 - **Pass** the final written exam
 - DH2320: Pass / Fail
 - DD2258: A-F
- DH2320: 6 CP
- DD2258: 7.5 CP

- Concept
 - Theory & practice
- Theoretical Assignments
 - Each student must prepare a write-up
 - Hand-in solutions on paper (written, printed) before they are discussed in class
 - Will be returned a week later
 - Solutions will be discussed in the tutorial course

- Practical Assignments
 - Programming assignments
 - Group work: **groups of approx. three students**
 - A C++ framework will be provided (Linux/Windows/Mac)
 - Windows users:
Visual Studio Express is available for free download
Other versions of Visual Studio are available through Microsoft Imagine
<https://intra.kth.se/en/it/programvara/microsoft-imagine-1.675383>
 - Linux users:
Multiple options: Console, K-Develop, QT Creator
 - Mac users:
XCode



Linear Transformations



Advanced Raytracing



Volume Rendering

- Practical Assignments: Grading
 - Grading in peer review
 - Group must show up entirely
 - Randomized assignment of pairs of groups
 - A grades the work of B
 - B grades the work of C
 - Everybody is graded individually, based on:
 - The group's implementation
 - Personal knowledge about the implementation
 - Everybody must be able to explain all of the code

- Anke Friederici
 - ankef@kth.se
 - LV 5, Room 4424



- Wiebke Köpp
 - wiebkek@kth.se
 - LV 5, Room 4424



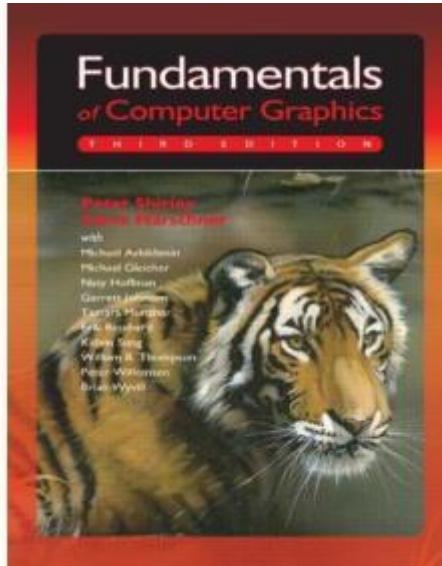
- Practical Assignments
 - Groups of three students
 - Form groups yourselves
 - Need to be formed by week 4, i.e., next week
 - Signup in Canvas for groups:
Open for signup today.
 - Bring your own equipment (laptop)
 - Possible for everyone?

- First Tutorial courses:
 - Using the programming environment (personal advice)
 - Introduction to the provided C++ framework
 - Help with forming groups
 - **Bring your laptop!**
- Introduction to Inviwo / CMake / Visual Studio / Compiling:
Friday, Jan 19, 15-17 in E35 & E36
- Introduction to C++ Programming / CMake / Visual Studio / Compiling:
Friday, Jan 26, 15-17 in Q15 & Q17

- Written exam
 - 2 hours
 - Similar to theoretical homework assignments
- Exam date: Saturday, March 17, 2018, 9:00-11:00
- Re-Exam date: Wednesday, April 4, 2018, 8:00-10:00

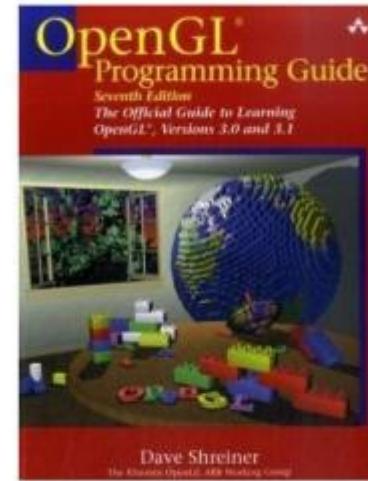
- **Questions & Suggestions**

- Please let us know if there are any issues anytime
- We appreciate your feedback! Please let us know:
 - ...if you find a certain part of the lecture hard to understand or not well explained.
 - ...any suggestions how to improve the lecture or the exercises.
 - ...any other questions, suggestions or concerns.
- Office hours: Appointments can be coordinated via e-mail



Peter Shirley
Fundamentals of Computer Graphics
AK Peters, 3. Edition

Dave Shreiner
OpenGL Programming Guide
Morgan Kaufmann, 7. Edition



- Books (cont'd)
 - J. D. Foley, A. van Dam, S. K. Feiner, J. F. Hughes: Computer Graphics - Principles and Practice (second Edition). Addison-Wesley Publishing Company, Inc., 1996
 - D. Salomon: Computer Graphics Geometric Modeling, Springer, 1999
 - A. Watt: 3D Computer Graphics. Addison-Wesley Publishing Company, Inc., 2000
- Journals
 - Computer Graphics Forum
 - IEEE CG & Applications
 - ACM Transactions on Graphics
 - ACM Transactions on Visualization and Computer Graphics

The lecture slides are partly based on material from

- Prof. Holger Theisel (Universität Magdeburg)
- Prof. Michael Wand (Universität Mainz)
- Prof. Heidrun Schumann (Universität Rostock)
- Prof. Marcus Magnor (Universität Braunschweig)
- Jun.-Prof. Thorsten Grosch (Universität Magdeburg)
- ...and other colleagues.
- Thanks!

- Tino Weinkauf
- Studied computer science in Rostock, Germany
 - Special focus: computer graphics and visualization
- Worked at Zuse Institute Berlin, Germany (2001 – 2009).
 - Flow analysis and visualization
- Received PhD in computer science (Dr. Ing.) from University of Magdeburg, Germany (2008).
- Post-doc at New York University, U.S.A. (2009 – 2011)
 - Feodor Lynen fellowship from the Alexander von Humboldt foundation
- Head of independent research group at Max-Planck-Institute for Informatics, Saarbrücken, Germany (2011 – 2014)
- Professor of Visualization, KTH Stockholm (since 2015)
- Research interests: Visualization, Data Analysis, Computer Graphics, Topology, Geometric Modeling, Shape Analysis

- Name
- Where do you come from?
- Where did you do your Bachelor?
- In what field?
 - Master in Computer Graphics
 - Master in Visualization
 - Master in Geoinformatics
 - Master in Mathematics
 - Master in Physics
 - Something else?

<https://goo.gl/forms/JXpxxCJeheSAW2Fz1>



Introduction to Visualization and Computer Graphics

DH2320

Prof. Dr. Tino Weinkauf

Introduction to Visualization and Computer Graphics

Terms and Definitions

Visual Computing is the field of

- acquiring,
- analyzing,
- processing, and
- synthesizing

visual data by means of computers.

Computer Vision

image analysis

Visualization & Computer Graphics

image synthesis

model

Analysis

model

Geometric
Modeling

Rendering

digital image

Image
Processing

Computational
Photography

Acquisition

sensor

screen

Computer Vision

image analysis

Visualization & Computer Graphics

image synthesis

model

Analysis

model

Rendering

digital image

Acquisition

sensor

Display

screen

Data

Geometry

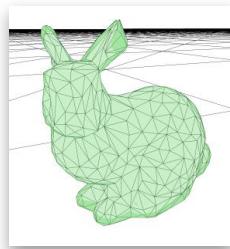
Image/Video

Purpose

*filtering
mapping*

rendering

Entertainment

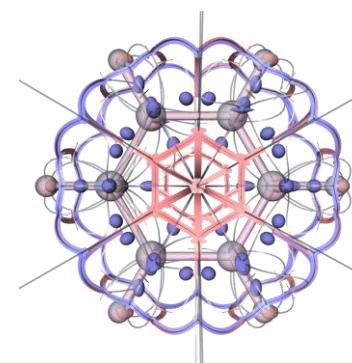


Education



Visualization

Insight



01010101	0
01010101	0
01010101	0
00111010	1
01010101	0
01010110	0





Introduction to Visualization and Computer Graphics

DH2320

Prof. Dr. Tino Weinkauf

Introduction to Visualization and Computer Graphics

Applications



- Games
 - Has to “look” good
 - Natural phenomena
 - Ad-hoc techniques are ok
 - For example: textures & shaders to “fake” details

Crysis 2, PC 2011



- Movies

- Has to “look” good
- Natural phenomena
- Ad-hoc and physically based methods
- Often rendering times of 1000 hours for a single frame!

Avatar, 2009



- **Landscape Planning**
 - Realistic rendering of plants for planning and virtual tours

© Kizo | www.arscom.hk

[\[www.laubwerk.com\]](http://www.laubwerk.com), 2015

- Training
 - Flight simulator
 - Driving simulator



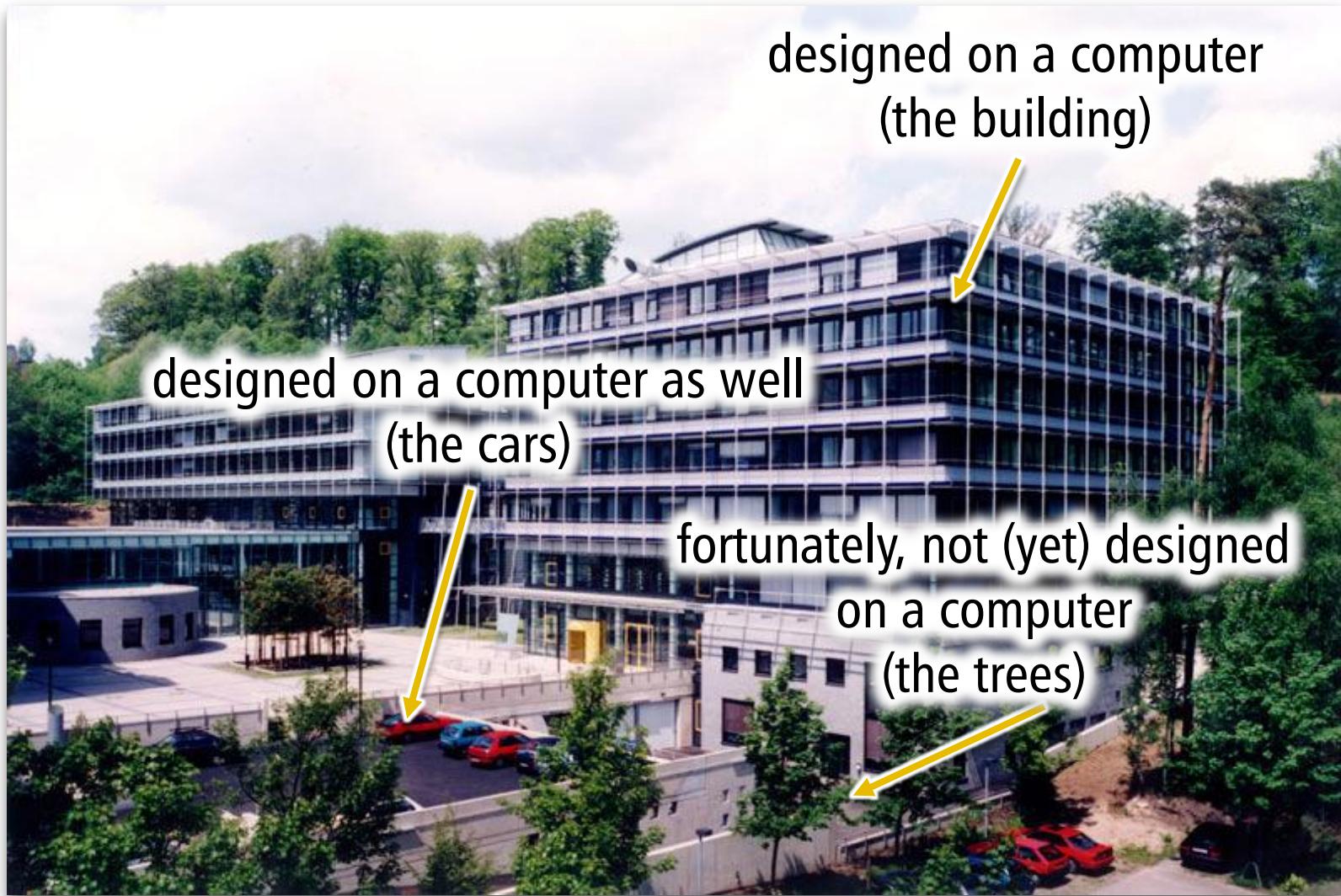
[www.flugsimulator.com, 2015]

- CAD / CAM

- Precision Guarantees
- Geometric constraints
(e.g. exact circles)
- Modeling guided by rules
and constraints



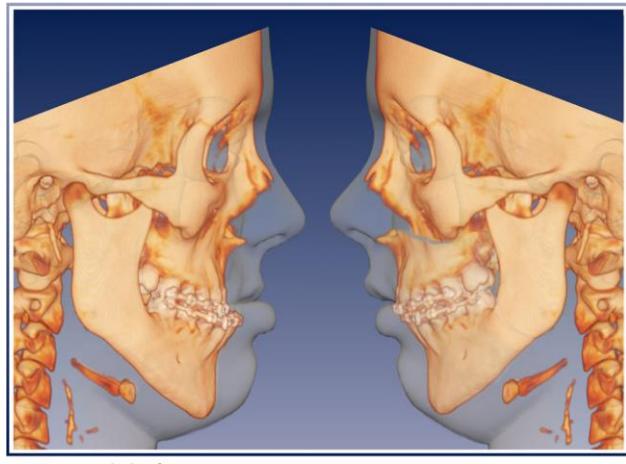
[aimatshape.net]



[c.f. Danny Hillis, Siggraph 2001 keynote]

- Visualization
 - Understanding data
 - Simulation, medicine, empirical sciences, ...
 - Focus on analysis or presentation of insights
 - Human perception important

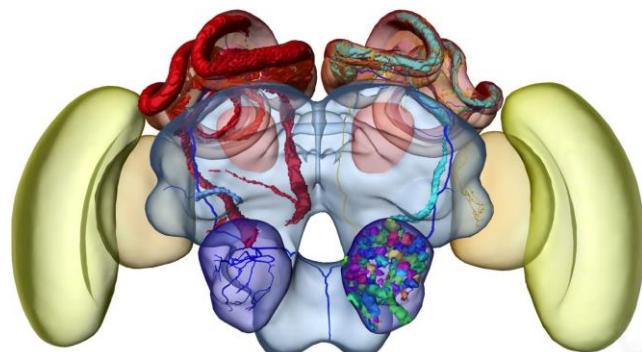
(Some) Applications of Visualization



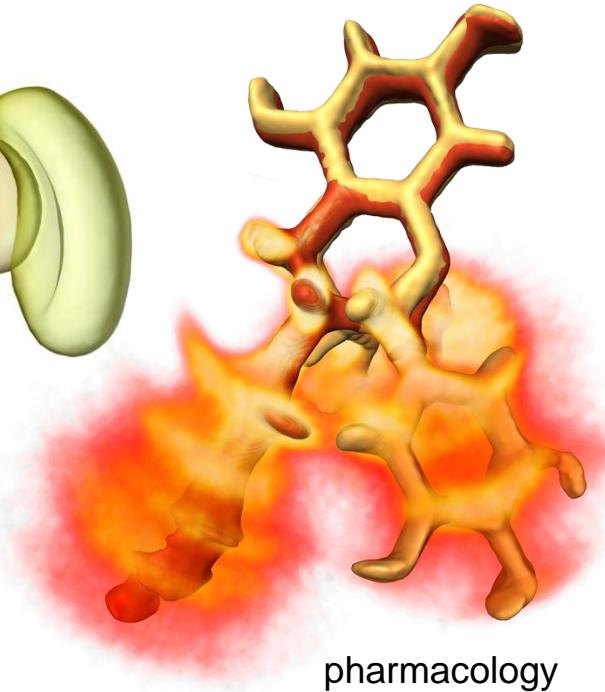
Imago animi vultus

Cicero, 106 - 43 v. Chr.

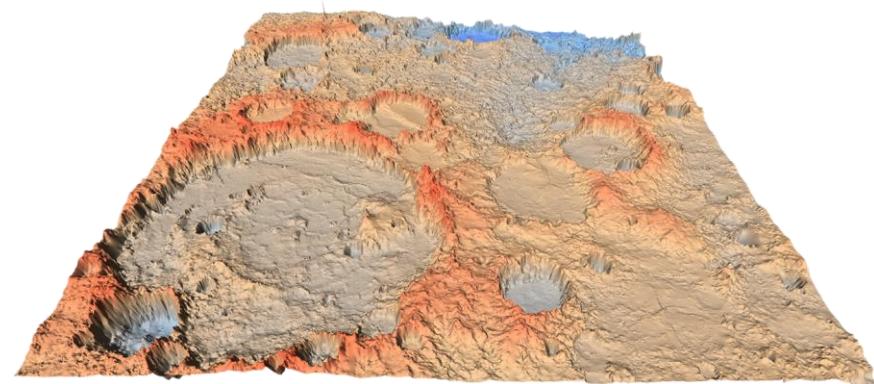
medicine



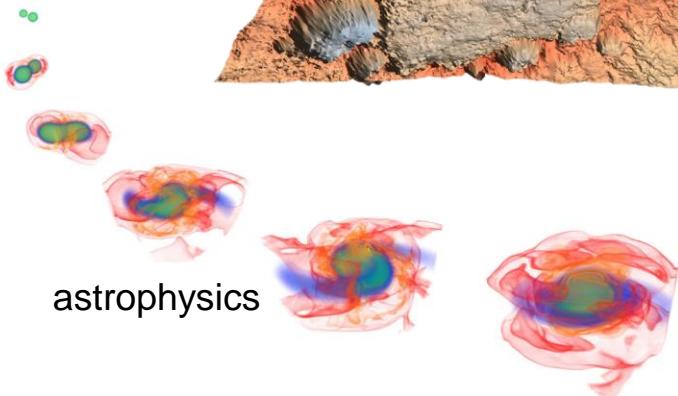
biology



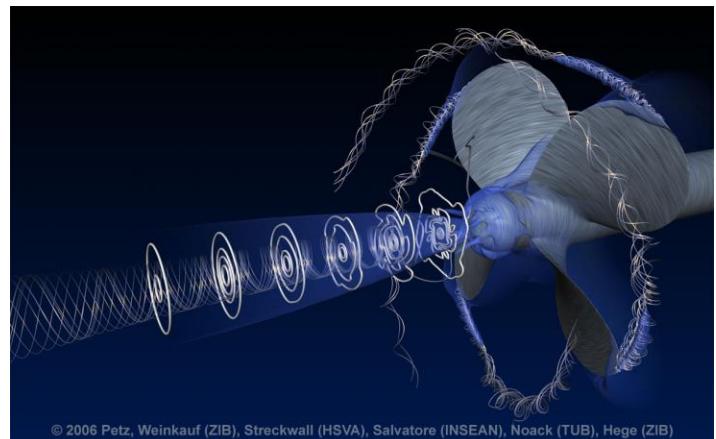
pharmacology



planetology



astrophysics



© 2006 Petz, Weinkauf (ZIB), Streckwall (HSVA), Salvatore (INSEAN), Noack (TUB), Hege (ZIB)

engineering



Introduction to Visualization and Computer Graphics

DH2320

Prof. Dr. Tino Weinkauf

Introduction to Visualization and Computer Graphics

History

- Idea of visualization very old
- **Euclid's "Elements"**: drawings to represent and illustrate properties in geometry.
- **Middle Ages**: astronomical maps with arrow plots to visualize prevailing winds over the oceans.
- **18th century**: height lines used in topographical maps

- **Alexander von Humboldt** (German scientist and explorer, 1769 – 1859)

Investigations of temperature gradients on the northern hemisphere. (1817)

- **René Descartes** (French philosopher, mathematician, physicist, 1596 – 1650)

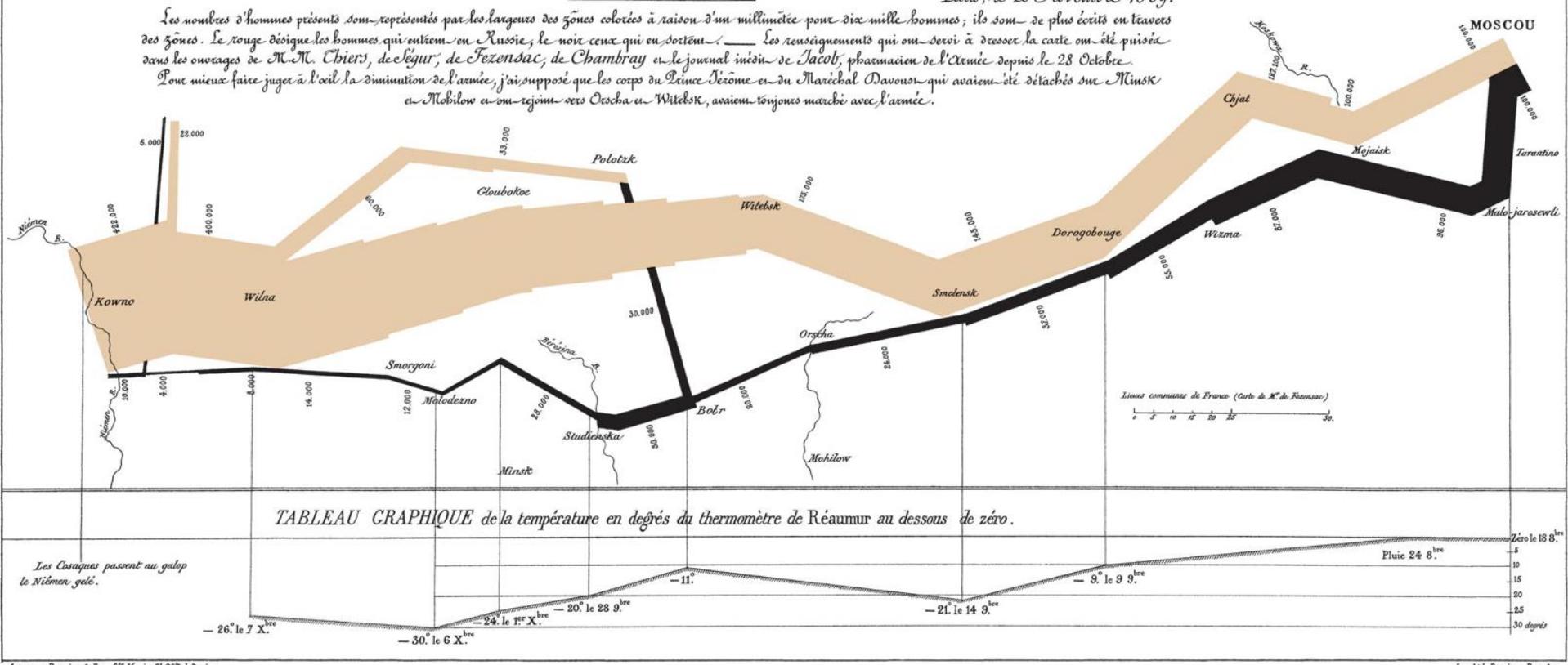
"Imagination or visualization, and in particular the use of diagrams, has a crucial part to play in scientific investigations". (1637)

Carte Figurative des pertes successives en hommes de l'Armée Française dans la Campagne de Russie 1812-1813.

Dessinée par M. Minard, Inspecteur Général des Ponts et Chaussées en retraite à Paris, le 20 Novembre 1869.

Les nombres d'hommes présents sont représentés par les largeurs des zones colorées à raison d'un millimètre pour dix mille hommes; ils sont de plus écrits en travers des zones. Le rouge désigne les hommes qui entrent en Russie; le noir ceux qui en sortent. — Les renseignements qui ont servi à dresser la carte ont été puisés dans les ouvrages de M. M. Chier, de Segur, de Tézenas, de Chambray et le journal médical de Jacob, pharmacien de l'Armée depuis le 28 Octobre.

Pour mieux faire juger à l'œil la diminution de l'armée, j'ai supposé que les corps du Prince Jérôme et du Maréchal Davout, qui avaient été détachés sur Minsk et Mohilow et se rejoignent vers Orsha et Witebsk, avaient toujours marché avec l'armée.



1869 Cartography by Charles Joseph Minard

Napoleons campaign against Russia (1812/13)

- **Wilhelm Conrad Röntgen** (German physicist, 1845 – 1923)

X-rays (1895)
first Nobel Prize in Physics (1901)



- **Rosalind Franklin** (British biophysicist, 1920 – 1958)

X-ray diffraction images of DNA (1952)

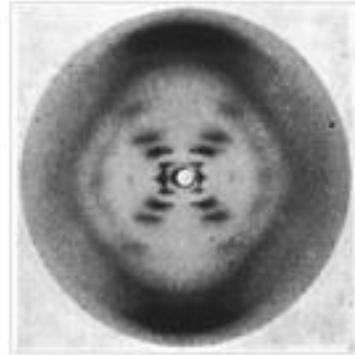


Photo 51
X-ray diffraction image of sodium
salt of DNA. B configuration

Nobel prize went to Watson, Crick, and Wilkins in 1962

Milestones in Flight History Dryden Flight Research Center



C-5A

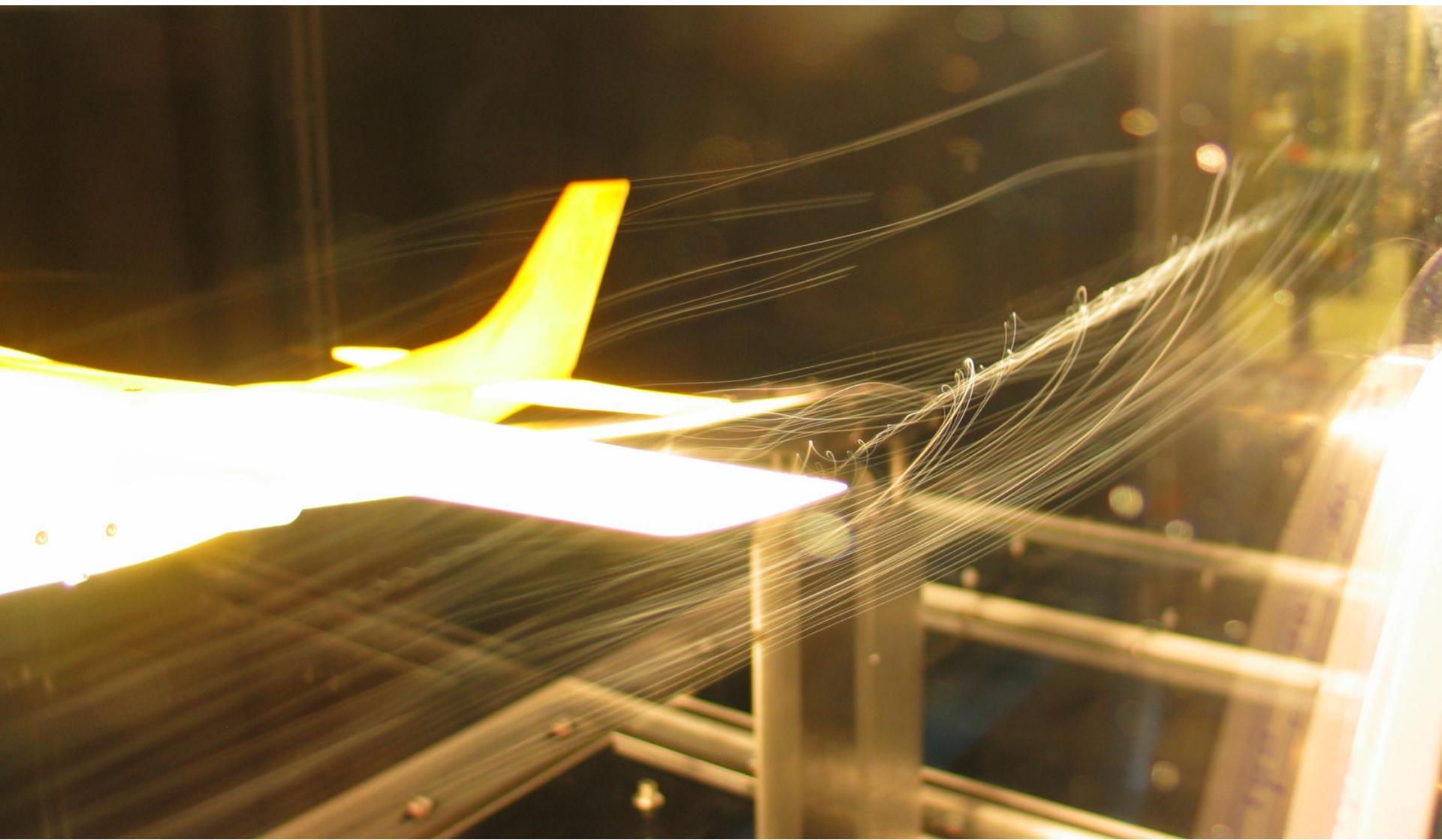
Wing Vortice Tests at Langley

Circa 1970s



Smoke angel

A C-17 Globemaster III from the 14th Airlift Squadron, Charleston Air Force Base, S.C. flies off after releasing flares over the Atlantic Ocean near Charleston, S.C., during a training mission on Tuesday, May 16, 2006. The "smoke angel" is caused by the vortex from the engines.
(U.S. Air Force photo/Tech. Sgt. Russell E. Cooley IV)



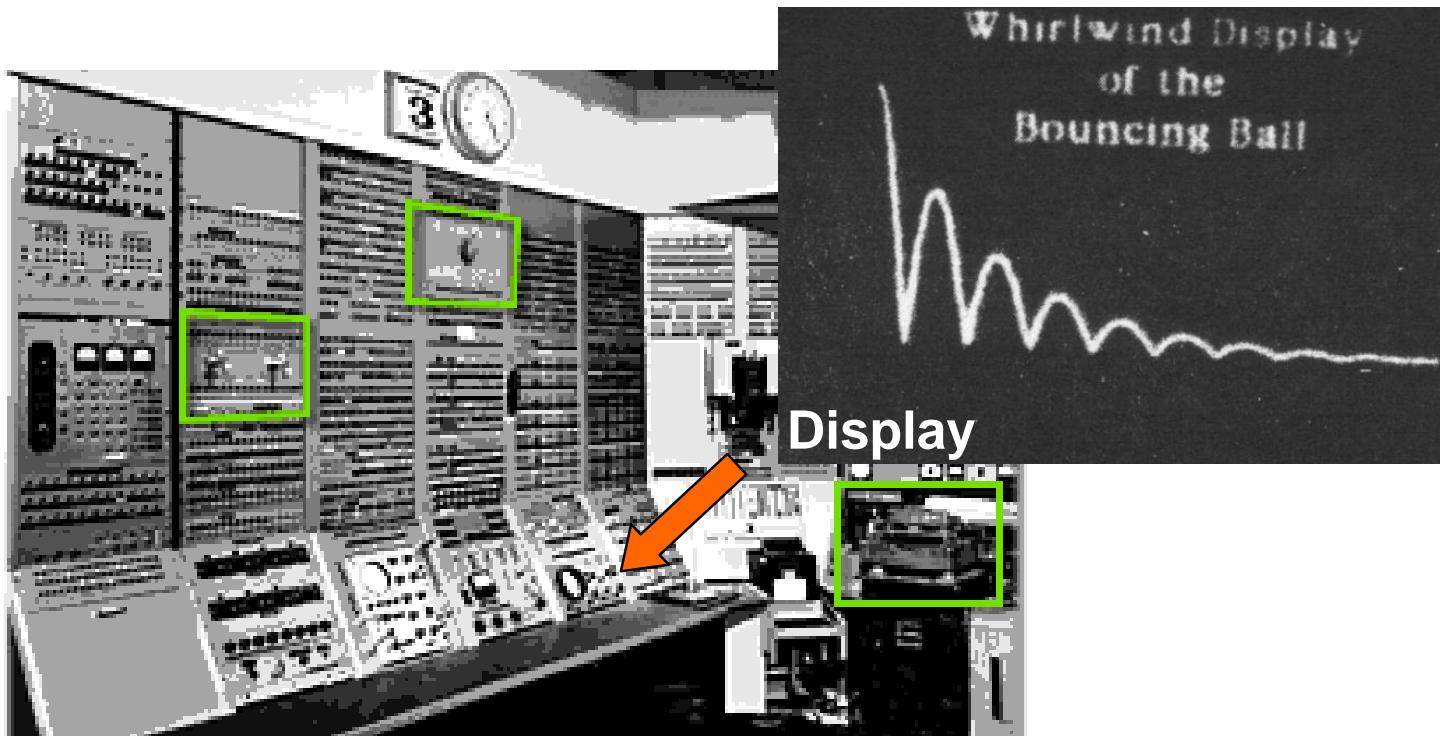
A wind tunnel model of a Cessna 182 showing a wingtip vortex.
Tested in the RPI (Rensselaer Polytechnic Institute) Subsonic Wind Tunnel.

By Ben FrantzDale (2007).

- upcoming computer technology: new challenges!
- virtual experiments, where the real ones are too expensive or dangerous
- larger data sets
- new opportunities to create visual representations (Computer Graphics)
- 1987: Visualization becomes discipline of its own
 - 1987 Marching Cubes
 - 1987 Parallel Coordinates
 - 1989 Vector Field Topology
 - 1993 Line Integral Convolution

- Since 1990: annual IEEE Visualization Conference
- Since 1999: annual Eurographics Symposium/Conference on Visualization (EuroVis)
- journals, books...
- many research groups worldwide, strong funding

- 1949:
First computer graphics on the *whirlwind* computer at MIT
 - *Bouncing Ball* program of C. Adams



- 1952:
Indication of flying objects on radar screens
 - SAGE computer with 82 graphics consoles for air control
 - First use of the light pen

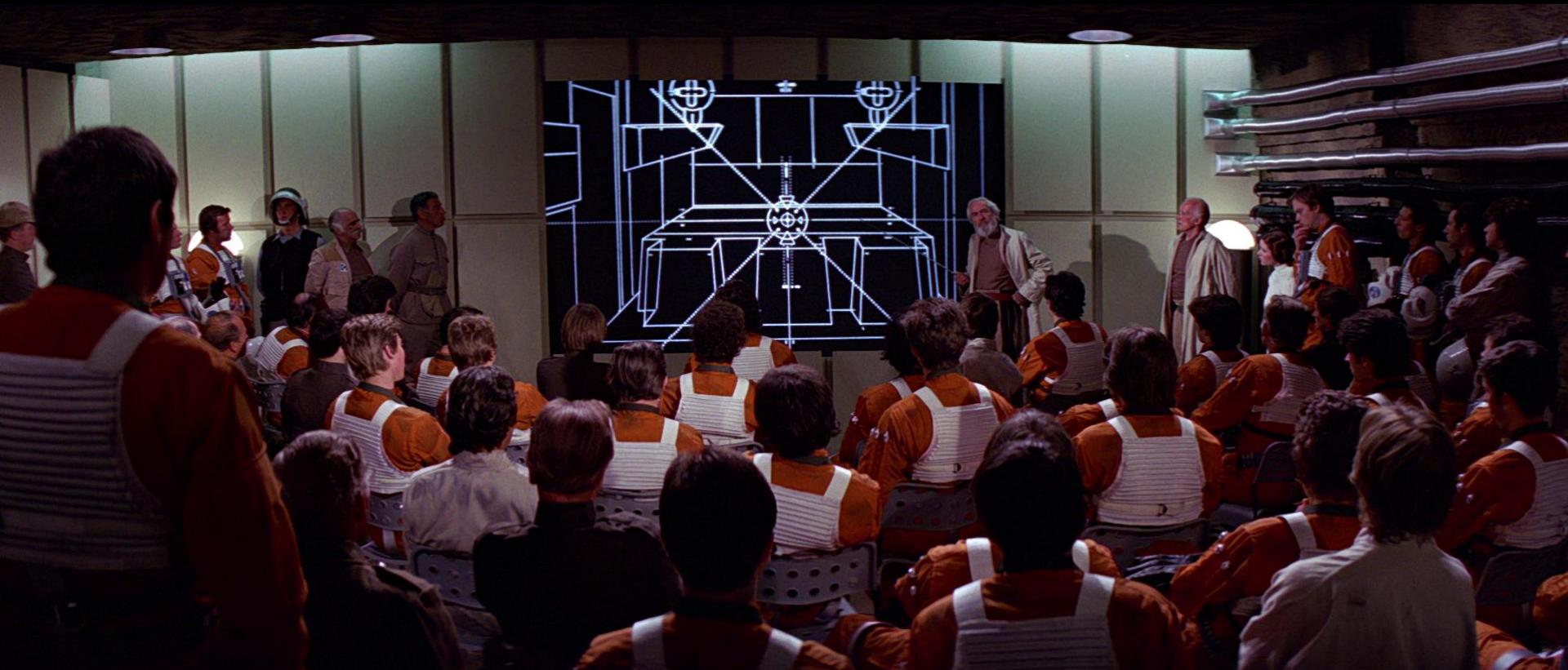


- Beginning of the 70s: first commercial CAD/CAM systems
- 1973: first ACM SIGGRAPH conference
 - SIGGRAPH: Special Interest Group on Computer Graphics
 - ACM: Association of Computing Machinery
 - 1200 participants in 1973
 - Now: approx. 20000 participants

*1975: M. Newell (Univ. of Utah) models the **Utah tea pot** – a computer graphics icon.*



Star Wars (1977)



(c) Twenty Century Fox

Tron (1982)



(c) Walt Disney Productions

Koronis Rift (C64, 1985)



(c) Lucasfilm Games

Luxo Jr. (1986)



(c) Pixar

Stunt Car Racer (Amiga, 1989)

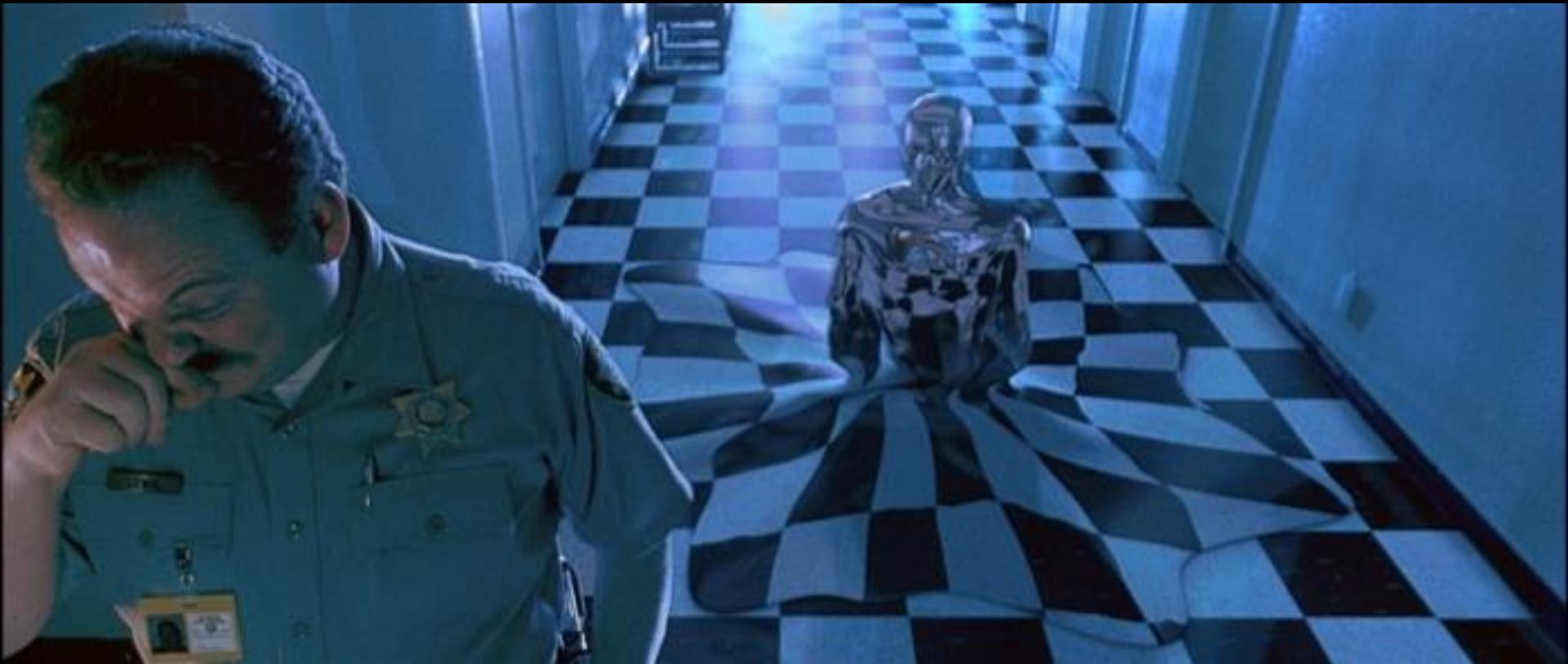


The Abyss (1989)



(c) 20th Century Fox

Terminator II (1991)



Comanche (PC, 1992)



Doom (PC, 1993)



Toy Story (1995)



(c) Pixar

Quake (PC, 1996)



(c) id Software

Final Fantasy (2001)



(c) Columbia-Tristar

The Lord of the Rings (2002)



(c) New Line Cinema

Avatar (2009)



(c) 20th Century Fox

Crysis 2 (PC, 2011)



(c) Crytek

Last Night on Reddit (2014)



(c) Hossein Diba



Introduction to Visualization and Computer Graphics

DH2320

Prof. Dr. Tino Weinkauf

Introduction to Visualization and Computer Graphics

Strong research leads to strong results

“Point-based Global Illumination”

Pixar, Industrial Light & Magic



Point-Based Approximate Color Bleeding

Per H. Christensen
Pixar Technical Memo #08-01
Pixar Animation Studios

Figure 1: (a) Point-based ambient occlusion test from "Tin's Up" © Sony Pictures Imageworks. (b) Point-based color bleeding in a final frame from "Pirates of the Caribbean: Dead Man's Chest" © Disney Enterprises, Inc. Jerry Bruckheimer Inc., image courtesy of Industrial Light & Magic.

Abstract

This technical memo describes a fast point-based method for computing diffuse global illumination (color bleeding). The computation is 40 times faster than ray tracing, uses less memory, but requires more memory than a standard point-based method for mapped surfaces, complex shadows, or many complex light sources. These properties make the method suitable for movie production.

The rays in the method is a point visual (surfel) representation of the directly illuminated geometry in the scene. The surfels in the scene are organized into a point cloud, where each point in the point cloud represents a small cluster of approximated using spherical harmonics. To compute the indirect illumination at a rendering point, we add the light from all surfels using three degrees of accuracy: ray tracing, regular disk sampling, and point sampling. Ray tracing is used for rendering the active nodes and surfels on-demand and caching them. Variations of the method efficiently compute area light distributions using global illumination gathering, and using the HEMI environment map. Global illumination influences reflections, human ambient occlusion, and glinty refraction.

The method has been used in production of more than a dozen feature films, for example for rendering Davy Jones and his crew in two of the "Pirates of the Caribbean" movies.

Keywords: Global illumination, color bleeding, radiosity, area light, ambient occlusion, point clouds, point-based rendering, raytracing, complex scenes, movie production.

1 Introduction

Standard methods for global illumination reach an industry (Dobashi et al. 1994), although they are slow and difficult to implement. Previous efforts (Fournier 1992) have not been widely used in movie production. This is mainly due to their enormous render times and memory requirements for the very complex scenes that are used in

geometry — the light source, surface and displacement shaders are very complex and take a long time to evaluate.

In this technical memo we describe a new point-based approximate global illumination method that is faster than the standard methods. First a sparse representation of the directly illuminated geometry is created in a precomputation phase and stored in a point cloud. (A surfel is a point in a point cloud, which is a point with associated normal, radius, and other data elements.) In our case, each surfel represents color reflected from a small part of a surface.) Then the surfels in the point cloud are organized into a point-based global illumination gathering pass, where each surfel node is approximated using spherical harmonics.

To compute the global illumination at a surface point, we add the illumination from all nodes using three degrees of accuracy: ray-tracing (if no ray-tracing hardware is available, other nearby surfels are approximated as disks), and distance surfels are accounted for by evaluating the shadow of the active node. The active node is the closest node to the point according to distance and scattered onto a narrow cone of "pixels". Finally, the rendered illumination (color bleeding) is computed by summing the contributions of the active nodes in the point-based global illumination gathering pass (for diffuse reflection) and the solid angle of the raster pixels.

Huge point sets are handled efficiently by reading the active nodes and surfels on demand and caching them.

The advantages of our point-based global illumination method are faster computation times, the geometry primitives do not have to be kept in memory, no ray-casting acceleration data structure is needed, and the memory usage is proportional to the number of points in memory at any given time, no scene, displacement mapping and complex light sources and surface shaders don't slow it down, the computation can be done on-the-fly and in real-time production. The disadvantages are that it is a point-based approach, and that the results are not guaranteed to be as precise as ray tracing. However, our goal in this project is not numerical accuracy, just visually

[Fluch der Karibik 2, 2006]

“PantaRay – Visibility Precomputing”

Nvidia, Weta Digital

[Avatar, 2009]



PantaRay: Fast Ray-traced Occlusion Caching of Massive Scenes

Jorge Paredes^{*} NVIDIA Research Luis Fuentes^{*} Weta Digital Matteo Mella[†] NVIDIA Research Timo Aila[‡] NVIDIA Research

Figure 1: The geometry complexity of scenes rendered in the movie Avatar often exceeds a billion polygons and contains widely distant objects and regions are rendered in a level of detail and consistency while the base of most scene characters are modeled to our resolution, others are just rendered as a blur. The sparse resolution of occlusion marks precomputed by our system also spans several orders of magnitude.

1 INTRODUCTION

We describe an acceleration of a novel system for precomputing sparse occlusion marks. These marks are useful for understanding what elements lighting paths that work in the spherical harmonics domain. The novel method was used as a primary light field rendering technique in the movie Avatar. The system can handle scenes of unprecedented complexity through the use of a novel ray tracing algorithm for rendering hidden ray tracing acceleration structures, and a novel use-of-line (UOL) ray tracing algorithm for the computation of directional occlusion and spherical integrations of arbitrary points.

1.1 Categories 1.1.2 Graphics Systems; 1.2.1, 1.2.4, 1.2.6 Hard-copy devices; 1.3.7 Three-Dimensional Graphics and Realism; 1.4.2 Calculating, measuring, and estimating—Raytracing.

Keywords: global illumination, precompute, radiance transfer, caching, ray casting

^{*} To appear in Proceedings of SIGGRAPH Asia 2010, December 1–4, 2010, Seoul, Korea. © 2010 ACM 978-1-4503-0066-3/10/12 \$15.00.

[†] Now at Pixar Animation Studios.

[‡] Now at Disney Research.

Abstract We describe an acceleration of a novel system for precomputing sparse occlusion marks. These marks are useful for understanding what elements lighting paths that work in the spherical harmonics domain. The novel method was used as a primary light field rendering technique in the movie Avatar. The system can handle scenes of unprecedented complexity through the use of a novel ray tracing algorithm for rendering hidden ray tracing acceleration structures, and a novel use-of-line (UOL) ray tracing algorithm for the computation of directional occlusion and spherical integrations of arbitrary points.

The movie Avatar features unprecedented geometric complexity (Figure 1), with production shots containing anywhere from six to over one billion polygons. To make the rendering of such complex scenes manageable while maintaining quality, we developed a system for precomputing sparse occlusion marks and the ability to produce them on demand. Our system builds on prior work in parallel ray tracing methods based on spherical harmonics (SH) [Bala et al. 2004] and hierarchical ray tracing [Aila et al. 2006]. The novel ray tracing algorithm for rendering hidden ray tracing acceleration structures, and a novel use-of-line (UOL) ray tracing algorithm for the computation of directional occlusion and spherical integrations of arbitrary points.

We describe the novel ray tracing algorithm for rendering hidden ray tracing acceleration structures, and a novel use-of-line (UOL) ray tracing algorithm for the computation of directional occlusion and spherical integrations of arbitrary points. The novel ray tracing algorithm for rendering hidden ray tracing acceleration structures, and a novel use-of-line (UOL) ray tracing algorithm for the computation of directional occlusion and spherical integrations of arbitrary points.

The PantaRay system is accurate, memory friendly, parallel and fast designed to handle scenes that are roughly an order of magnitude larger than previous systems. It is also able to handle complex spherical harmonics encoded directional occlusion (SH occlusion) and spherical averaging information for billions of points with highly detailed rendering artifacts.

The key contributions are the introduction of a flexible, memory-based geometry processing architecture, a novel use-of-line algorithm for the computation of directional occlusion and spherical integrations of arbitrary points, and a novel use-of-line (UOL) ray tracing algorithm for the computation of directional occlusion and spherical integrations of arbitrary points.

¹ Based on the “Light Field and Radiance Transfer” paper.

“Volumetric Lighting”

Disney Research

[Rapunzel, 2010]



A Programmable System for Artistic Volumetric Lighting

Stéphane Raveau¹ · Rand Jones² · Andrew Selle² · Dylan Laine² · Michael Karchell² · Wenceslao Jerez¹
¹Disney Research Zurich · ²Walt Disney Animation Studios

Figure 1: The system was used to render artistic volumetric effects for the movie *Tangled*. Our on-set designer’s ability to produce varying light fields is used to match the unique camera style of the film.

Abstract
We present a method for generating art-directed volumetric effects for real-world lighting situations. Our system allows the user to express the way they want certain light fields to behave. Our system makes the user experiment with their ideas about volumetric effects by using an intuitive programming paradigm, and describes the resulting volumetric field in terms of the physical properties it represents. This allows the user to quickly generate the physically-based photon fields needed to solve volumetric problems. We also provide a graphical interface that allows the user to easily program desired light fields directly in a 3D volume. We also provide a large space for artists to rapidly explore a wide range of physically-based art styles, as well as plausible, but unphysical, volumes. The system is built on top of a general-purpose rendering engine and provides a clean pipeline and complete API to enable artists to realize their lighting vision.

CN Categories: I.3.7 [Computer Graphics]: Three-Dimensional Displays and Robotics—Color shading, masking, and texture.

Keywords: Lighting design, artist control, participating media

Links: [DOI](#) [PDF](#)

1. Introduction
Light scattering in participating media is a necessity for many real-world applications. Light scattering is a key element of our visual perception, as well as the complex light transport within it, are difficult problems to simulate (e.g., [Blinn et al. 1982; Dugay et al. 2006; Kautz et al. 2007; Schröder et al. 2008; Shand et al. 2011]). Recent advances have made it feasible to incorporate light scattering into real-time rendering engines [Kautz et al. 2007]. However, most of this work has focused on simulating complex environments and rendering scenes. Physically accurate techniques, on the other hand, have been used to achieve a large body of challenging

graphics. The research addressed the problem of generating volumetric effects for real-world lighting situations in [Raveau et al. 2010]. However, artists authoring and manipulating volumetric lighting fields using arbitrary source sets were introduced in a previous framework, implemented on GPU [McNamee et al. 2008; Tavelin et al. 2009]. Similarly, on-set systems for programmable volumetric lighting have been reported in [Jerez et al. 2010]. In this paper, we introduce a new system to enable artists to realize their lighting vision.

While physically accurate and art-directed rendering have been largely conflicting goals, recent efforts to incorporate physically based rendering into GPU-based rendering engines [Raveau et al. 2010; Jerez et al. 2010; Kautz et al. 2010] have introduced a new paradigm for solving volumetric lighting fields that may provide a more complete solution to the problem of physically accurate and art-directed volumetric lighting. Unfortunately, physically accurate volumetric lighting is often not as controllable as physically-based art-directed volumetric lighting. This is because physically accurate volumetric lighting is often much more computationally expensive than physically-based art-directed volumetric lighting. We carefully combine these two areas and choose to present an art-directing physically-based approach for rendering volumetric lighting at art-directed lighting and rendering fields.

We present a system for generating art-directed volumetric effects that is the most physically accurate art-directed field field. We have an approach to photon density [Chen et al. 2012] that is based on a physically-based rendering engine [Raveau et al. 2010]. We make the following contribution while preserving photon density to allow for artistic control of volumetric effects:

- We introduce the first rendering engine for physically-based participating media that is able to generate art-directed volumetric fields that are both physically accurate and artist-controlled.
- We have an approach to photon density [Chen et al. 2012] that is based on a physically-based rendering engine [Raveau et al. 2010]. We make the following contribution while preserving photon density to allow for artistic control of volumetric effects:
- We offer an on-set system for physically-based art-directed volumetric lighting. We support multi-stage with a pre-computed, pre-integrated environment. While each stage contains a single scene, the physically-based approach, the photon density

“Out-of-Core Global Illumination”

DreamWorks Animation

[Kung Fu Panda 2, 2011]



Figure 8.1 A screenshot from the DreamWorks Animation movie “Kung Fu Panda 2”. The out-of-core method described in our paper shaded the global illumination for the whole frame in 6 minutes 23 seconds. It used 120 million points, and it took an additional 4 minutes 13 seconds to build the out-of-core scene, resulting in 27 million surface nodes. The total amount of memory used was 6.4 GB while the in-core scene only required 1.2 GB. The cache hit rate was 99%. In Section 7, we demonstrate storage of up to 17 billion points in 10 GB of disk space in 2.1 GB memory cap.

Abstract:
We describe a new technique for coherent out-of-core point-based global illumination and ambient occlusion. Point-based global illumination has been used in production for rendering extremely complex scenes, or in core storage, but it has never been used for real-time rendering of a scene. However, a hierarchical representation of the scene allows for direct access to any point in the scene. This makes it possible to efficiently implement a hierarchical depth-dependent shading step that would be extremely inefficient due to three amounts of GPU required. Our method greatly improves PBR algorithms with an out-of-core technique that uses minimal 10% and stores less memory than other methods. In addition, our method is able to handle large scenes by using a hierarchical approach. By doing this, we are able to preprocess the data in core space, an efficient 10 GB and an active render action per frame.

1. Introduction:
Point-based global illumination (PBI) [Fuchs91] is an well known method for computing indirect illumination which is vital for photo-realistic rendering. PBI is an alternative to ray tracing [Kajiya84], which is often used for path tracing [Mitsin98] or radiance caching [Walter01, Togo04], which have evolved in several DreamWorks Animation feature films including “Shrek Forever After”, “Megamind” and others.

In its most basic form, PBI is a level of detail (LOD) algorithm for computing both geometry and shading. It progresses, the geometry is finely sampled using a sequence of points, which are shaded using direct illumination and ambient occlusion. The main advantage of the point-based representation is its direct access to the hierarchy, especially in scenes, where each component contributes to the geometry. This makes it possible to efficiently implement a hierarchical depth-dependent shading step that would be extremely inefficient due to three amounts of GPU required. However, the points are not the vertices can become large, especially when the scene contains many objects. The quality of the final shading is linked to the resolution of the point samples, thus, the process must make these points act as a real world rendering. Generating large amounts of points results in a scene with 200 million nodes. The points there are only 22.6 GB of disk space and the vertex set will exceed 22.7 GB.

Figure 8.1 shows a screenshot from the DreamWorks Animation movie “Kung Fu Panda 2”. The out-of-core method described in our paper shaded the global illumination for the whole frame in 6 minutes 23 seconds. It used 120 million points, and it took an additional 4 minutes 13 seconds to build the out-of-core scene, resulting in 27 million surface nodes. The total amount of memory used was 6.4 GB while the in-core scene only required 1.2 GB. The cache hit rate was 99%. In Section 7, we demonstrate storage of up to 17 billion points in 10 GB of disk space in 2.1 GB memory cap.

© 2012 The Author(s). The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted IGI a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

Coherent Out-of-Core Point-Based Global Illumination
Sven Kornelius¹ and Eric Sedlacek² and Ryan S. Oberman³
¹DreamWorks Animation

“Artistic Simulation of Curly Hair”

Disney, Pixar



[Merida, 2012]

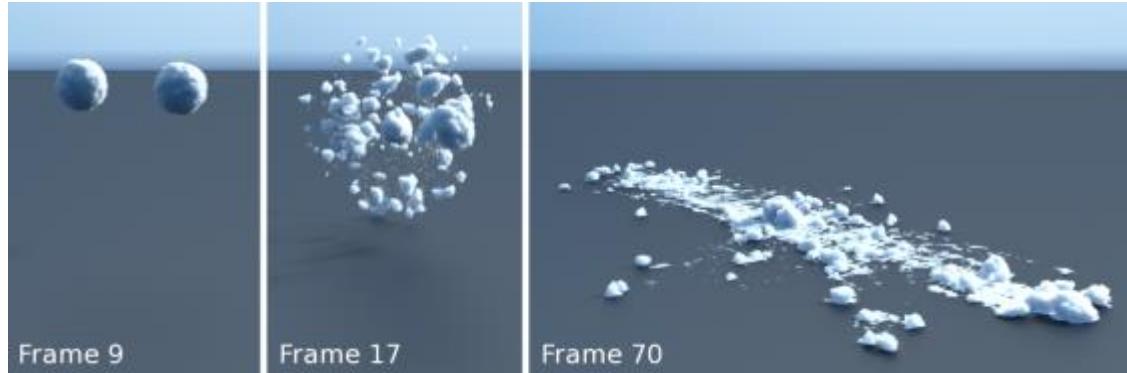


Introduction to Visualization and Computer Graphics, Tino Weinkauf, KTH Stockholm

“Simulation of Snow”

Disney, Pixar

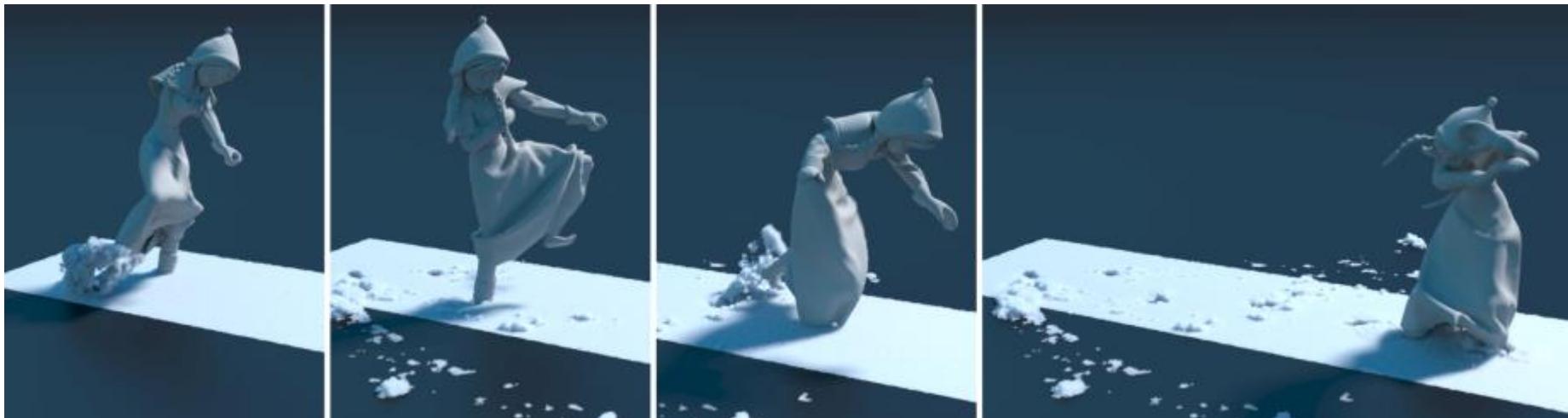
[Eiskönigin, 2013]



Frame 9

Frame 17

Frame 70



Introduction to Visualization and Computer Graphics, Tino Weinkauf, KTH Stockholm

PF: Performance Physics Infrastructure for Effects Simulations
Aleney Strelakova[†] Craig Schroeder[‡] Lawrence Choi[‡] Joseph Turner[‡] Andrew Selle[‡]
[†]University of California Los Angeles [‡]Walt Disney Animation Studios

Abstract

Snow is a challenging natural phenomenon to visually simulate. While the graphics community has previously considered accumulation and melting of snow, simulation of snow dynamics has not been well studied. Additionally, simulating snow as a solid and fluid has been difficult producing convincing snow results. Specifically, soft or dense snow that has both solid and fluid-like properties is difficult to handle. Consequently, this paper proposes a physics-based simulation scheme that can handle incompressible elastic-plastic constitutive models integrated with a hybrid Eulerian-Lagrange Material Point Method. The method is continuous and its hybrid nature allows us to use a regular Cartesian grid to represent the location of active material points. It also naturally allows us to derive a grid-based semi-implicit integration scheme that has a conditioning independent of the number of active material particles. We demonstrate the power of our method with a variety of snow phenomena including complex character interactions.

CN Categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Modeling—Animation; I.3.5 [Simulation and Modeling]: Types of Simulation—Animation

Keywords: material point, snow simulation, physically-based rendering

Links: [DOI](#) [PDF](#) [Wiki](#)

1 Introduction

Snow dynamics are amazingly beautiful yet varied. Whether it is powder snow fluttering in a skier's wake, foot steps crunching on dry snow crust or even parking snow melted tire tracks to make a snowman, snow dynamics are a challenge to simulate. The complexity of snow is compelling for snow to flip and refract (very difficult to model) or a complex wave to settle into a ripples (Held and Flato 1998; Cane et al. 2002; Khalsa et al. 2012). This makes snow a difficult material to model for one type of snow. This requires the need for a specialized solver that handles difficult snow behaviors in a single solver.

Specialized solvers for specific phenomena are frequently used in graphics and computational physics because achieving maximum resolution (and thus visual quality) requires efficiency. While a field solver can produce solid-like elastic effects (and vice versa), it is not the most optimal strategy. When solids and fluids are coupled, a specialized solver can compromise the overall resulting system to get good accuracy and performance for both phenomena. Unfortunately, snow has continuously varying phase effects, sometimes behaving as a rigid or flowing solid and some times exhibiting liquid-like behavior. Therefore, a specialized solver must simultaneously handle a continuum of material properties (flexible in the same domain, even though such a solver may not be encapsulated for a single discrete phenomena).

We present two main contributions to achieve these aims. First we propose a new Material Point Method (MPM) (Dolby et al. 1999) specifically designed to efficiently treat the wide range of material stiffness, collisions and topological changes arising in complex snow scenes. To our knowledge, this is the first time MPM has been applied to problems involving multiple types of material particles (powder with Eulerian-Cartesian grids). Separate them is no inherent need for Lagrangian mesh connectivity. Many researchers in graphics have implemented such hybrid grid and particle solvers (e.g., Lin et al. 2004; Yang et al. 2005; Liu et al. 2006) as a fast using a PUF2D¹ incompressible fluid interaction. In fact MPMs were designed as a generalization of the PUF2D¹ solver to computational and theoretical needs. MPMs are based on the finite difference grid and therefore do not require a mesh or a lattice grid. This is essential given the many topological changes exhibited by practical snow dynamics. Our second contribution is a new physics-based simulation scheme that can handle incompressible elastic-plastic snow behaviors. This solver is designed to achieve our goal of

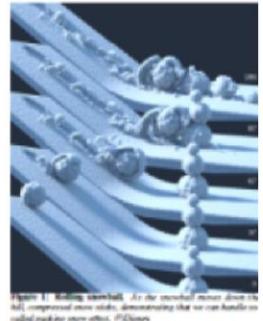


Figure 1: Melting snowball. As the melted water flows down the hill, it compressed more雪, demonstrating that we can handle incompressible packing snow effects. ©Disney



Introduction to Visualization and Computer Graphics

DH2320

Prof. Dr. Tino Weinkauf

Introduction to Visualization and Computer Graphics

Trends

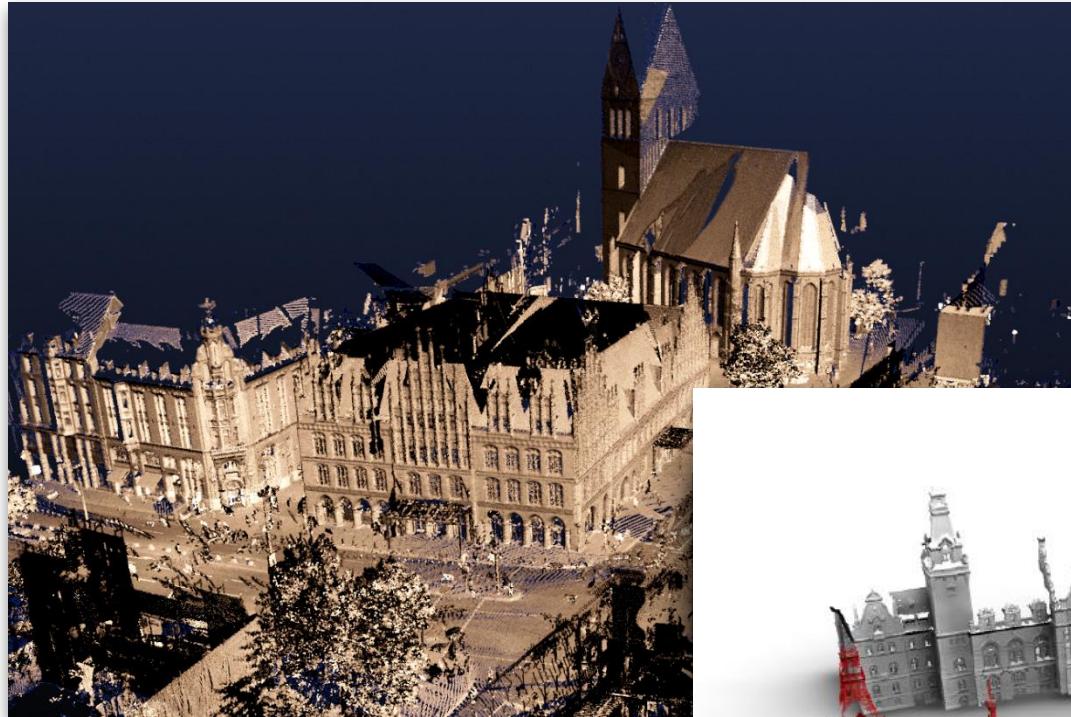
- Learning from real-world data
 - Complexity of reality
 - Machine learning + physical measurement



[Ihrke et al., CCD 2012]



[Christopher Schwartz, Michael Weinmann, Roland Ruiters,
and Reinhard Klein, Bonn University]



[courtesy of Claus Brenner,
IKG Hannover]



[Michael Wand, Martin Bokeloh, Siggraph 2010]

- New challenges ahead
 - Computational photography
 - Fabrication
 - Smart image/video editing
 - 3D computer vision / scene understanding



Introduction to Visualization and Computer Graphics

DH2320

Prof. Dr. Tino Weinkauf

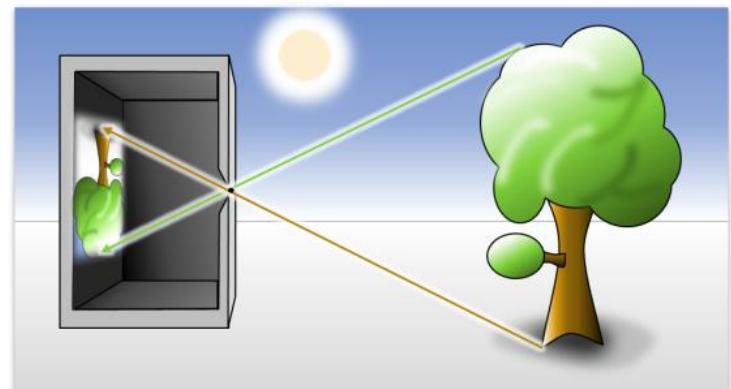
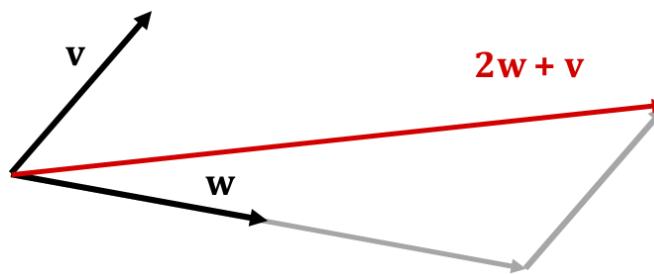
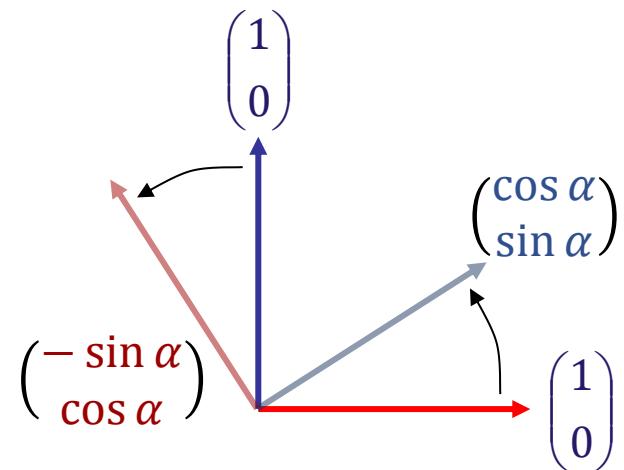
Introduction to Visualization and Computer Graphics

Overview of the Lectures
(tentative schedule)

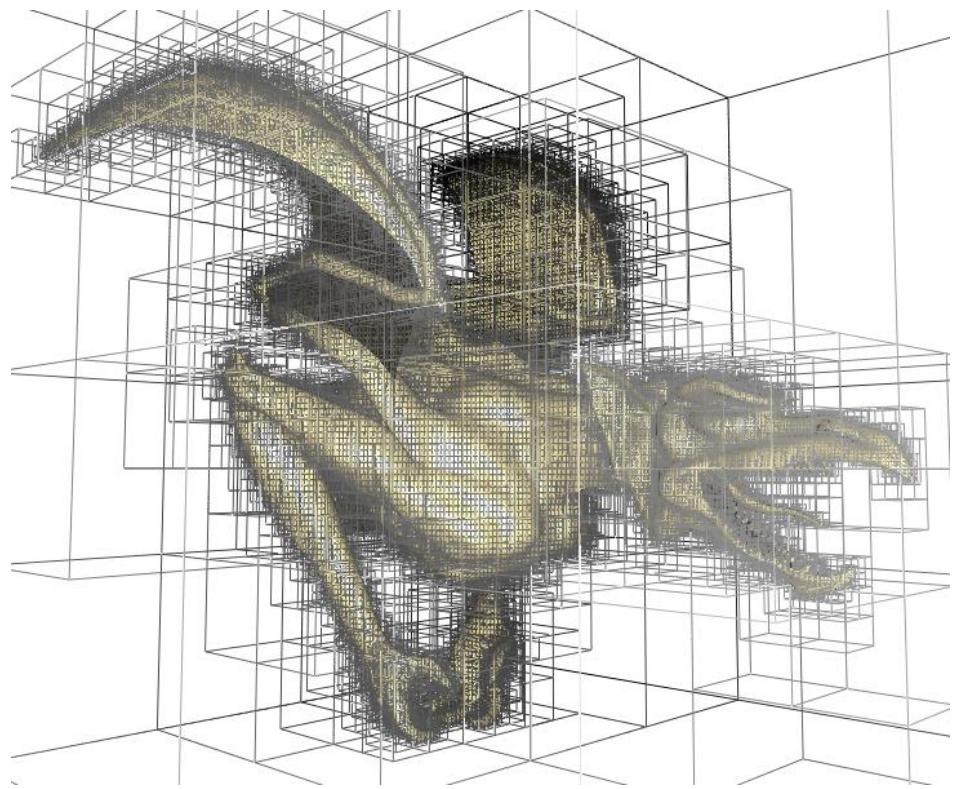
- Linear transformations and homogeneous coordinates
- Spatial data structures and grids
- Modeling meshes
- Interpolation in 2D and 3D grids
- Shading and color
 - color models and perception
- Rendering: rasterization (projection, clipping, visibility)
- Rendering: raytracing
- Raycasting a volume
- All-purpose visualization methods and their best practices

• Mathematics

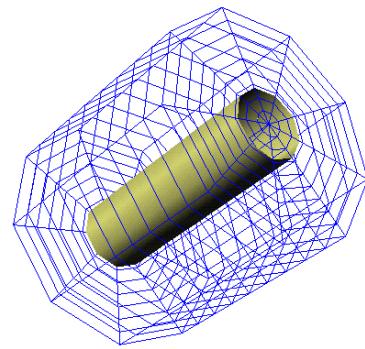
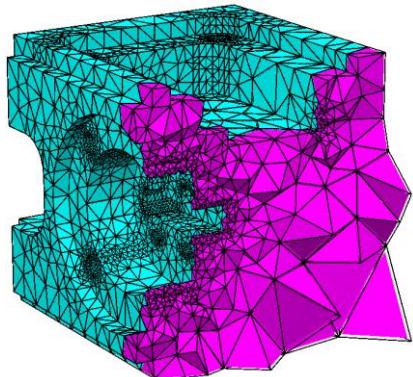
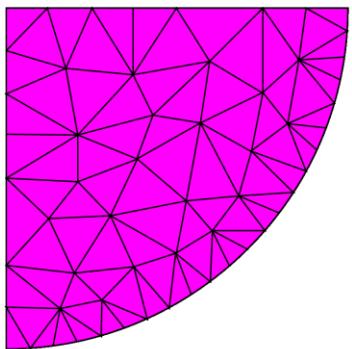
- Linear algebra
 - vectors / points
 - linear maps / matrices
- Projective geometry
 - Homogeneous coordinates
 - Perspective transformations



- Spatial data structures and grids
 - Quadtree / Octree
 - Bounding Volume Hierarchy
 - Structured grids (uniform, rectilinear, curvilinear)
 - Unstructured grids (triangle meshes, tetrahedral meshes)

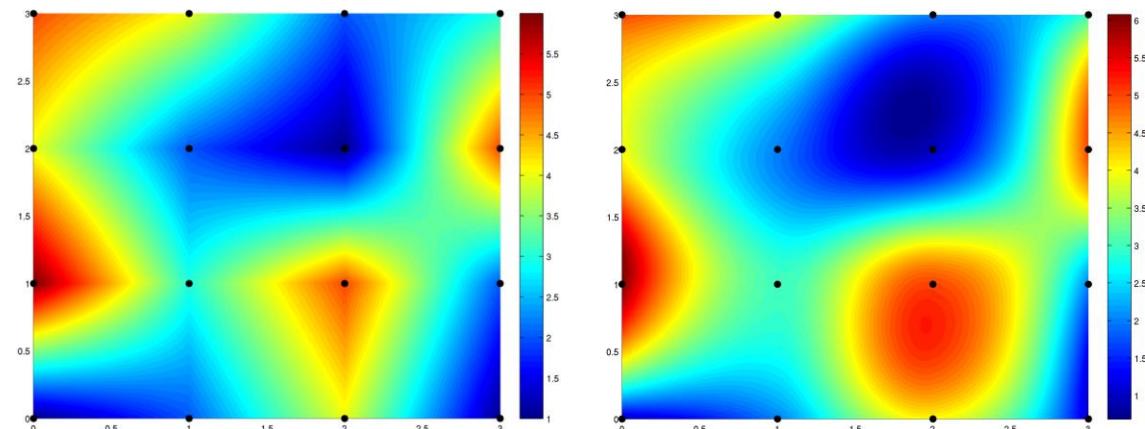
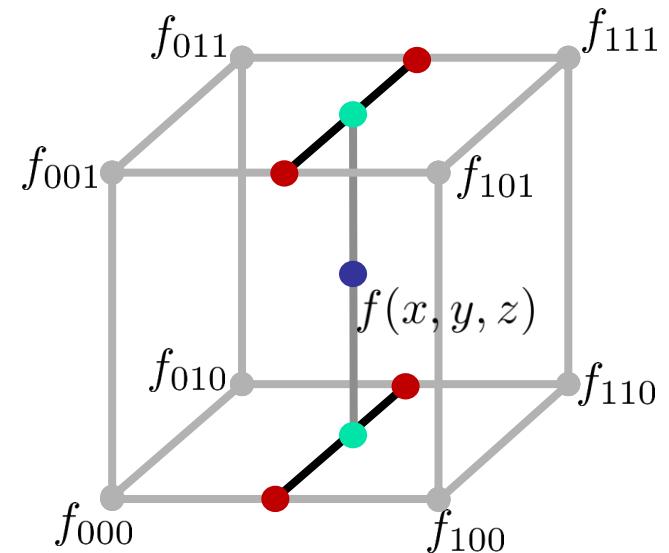
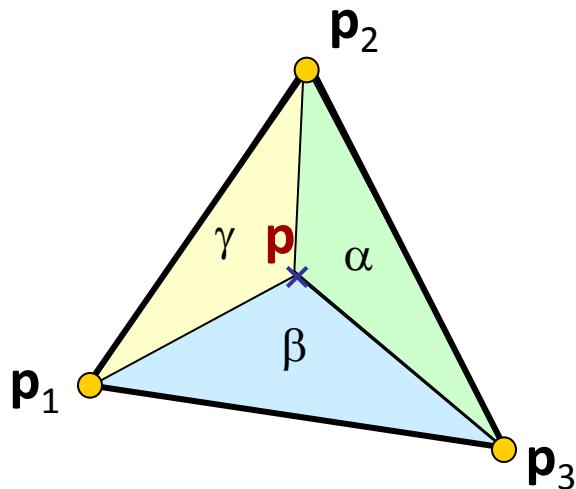


http://www.math.ucsb.edu/~chohong/vortex_in_a_box.jpg

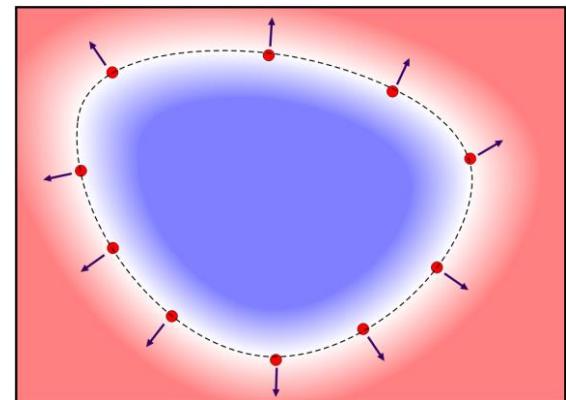
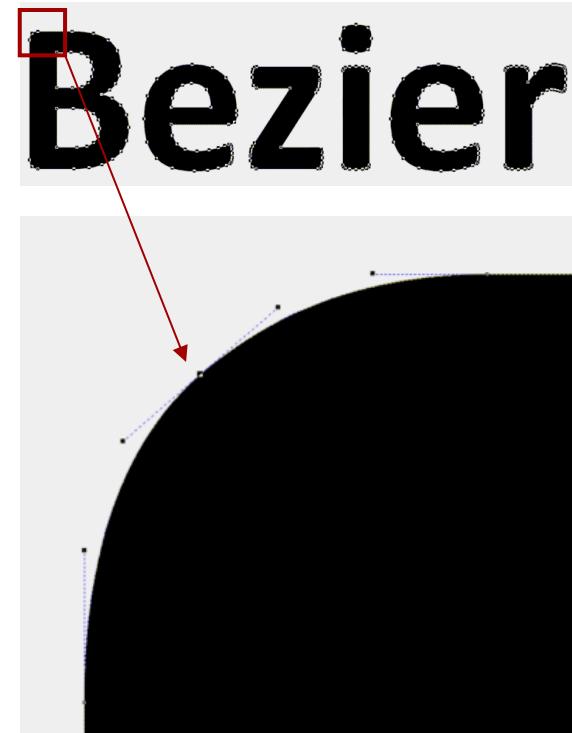
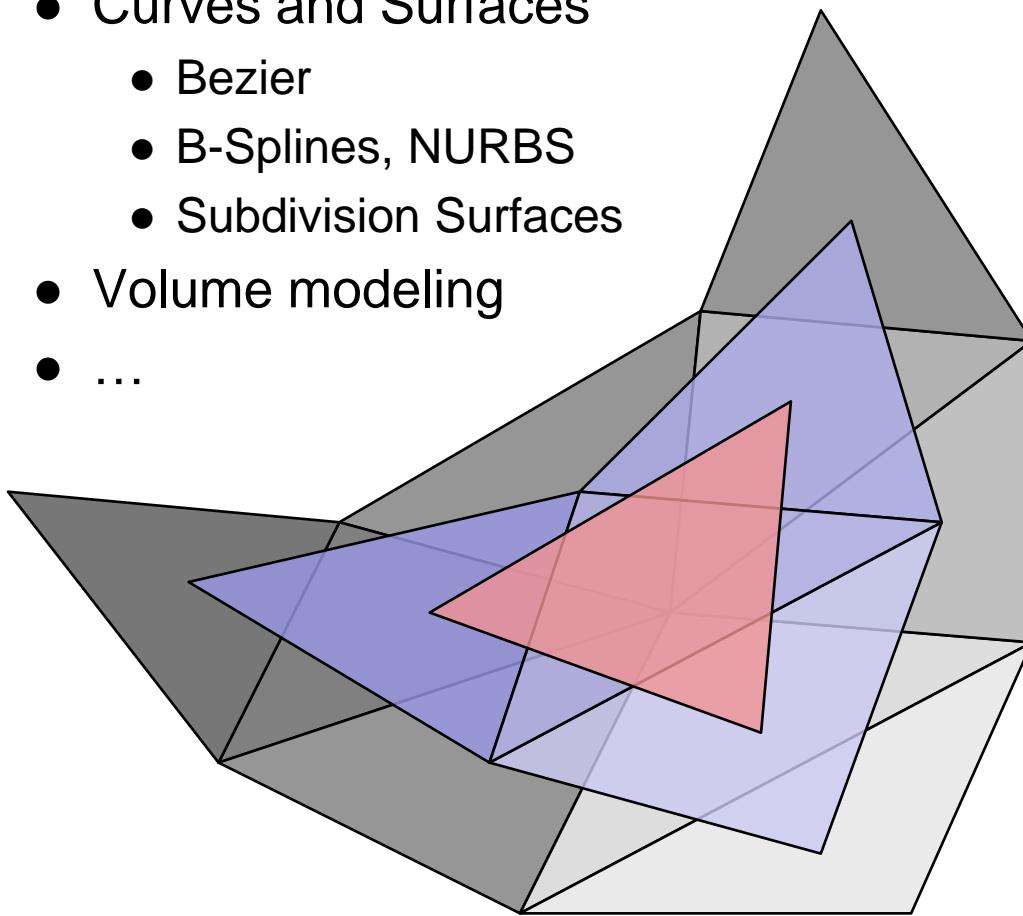


- Interpolation

- Linear interpolation
- Bilinear interpolation
- Trilinear interpolation
- Barycentric coordinates

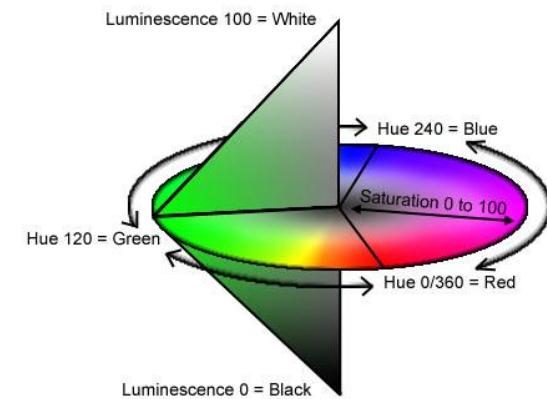
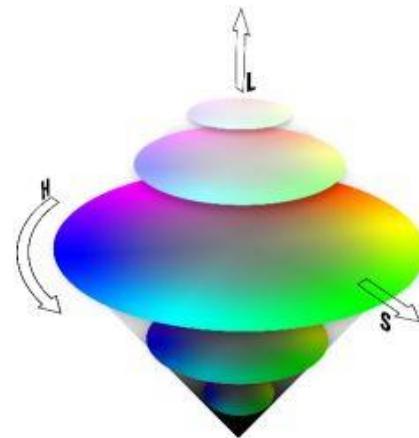
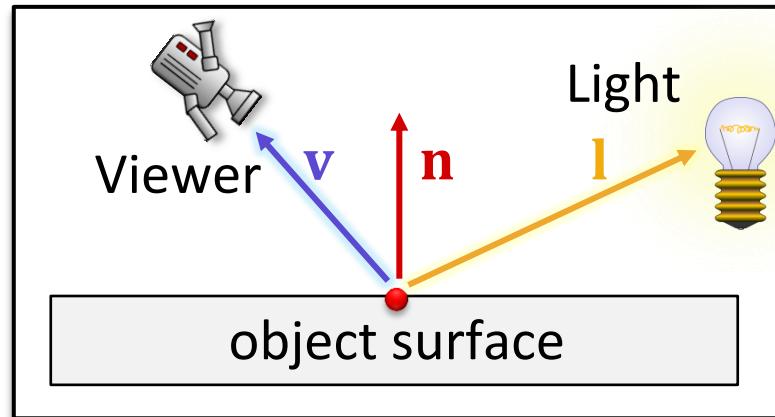
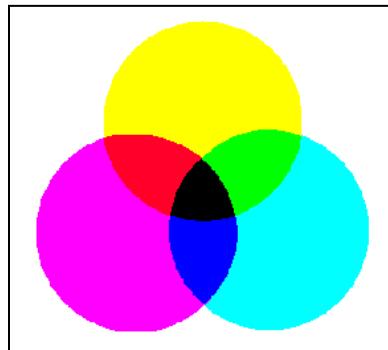
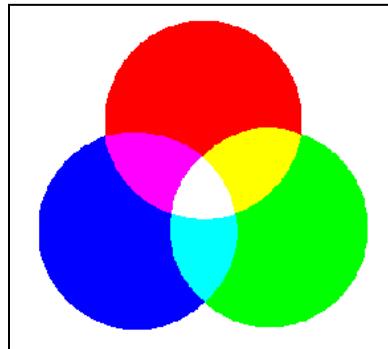


- Modeling
 - Overview of modeling methods
 - Solid models
 - Curves and Surfaces
 - Bezier
 - B-Splines, NURBS
 - Subdivision Surfaces
 - Volume modeling
 - ...



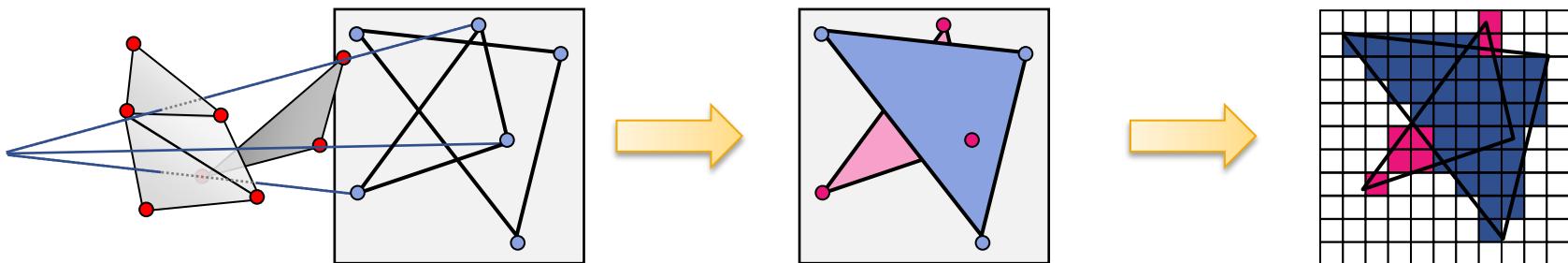
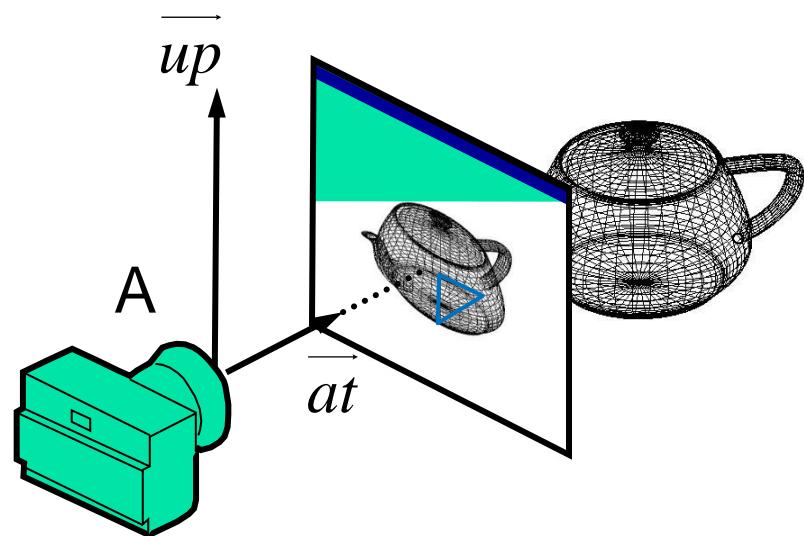
- Shading and Color

- Phong illumination model
- Color models
- Color perception

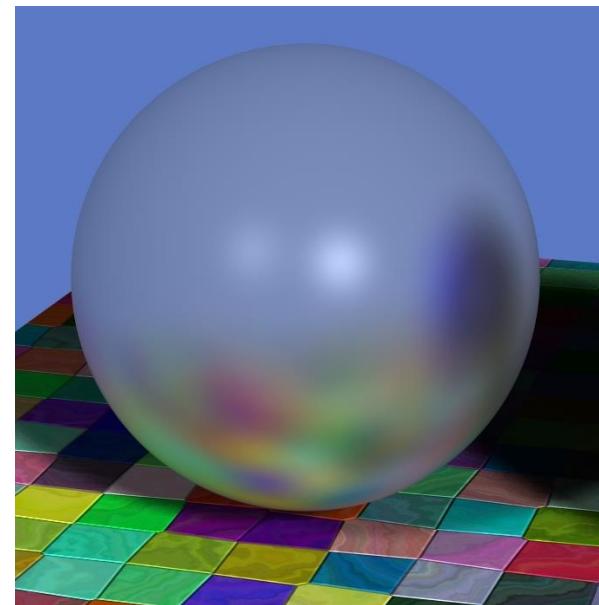
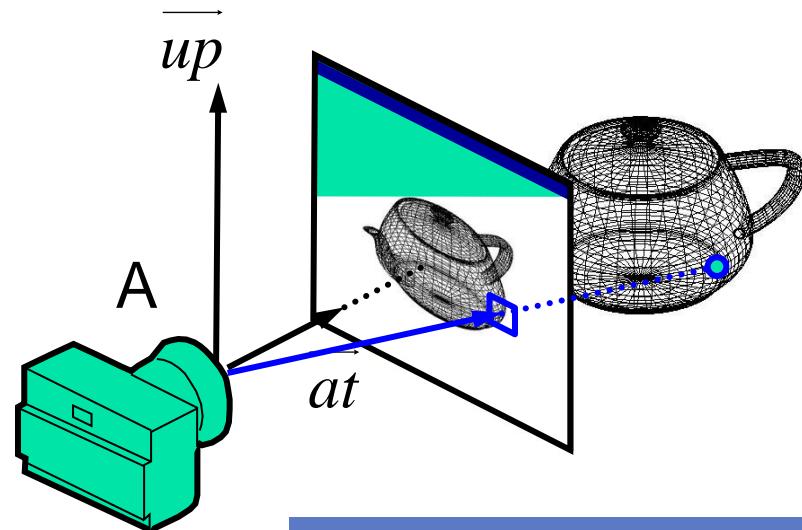
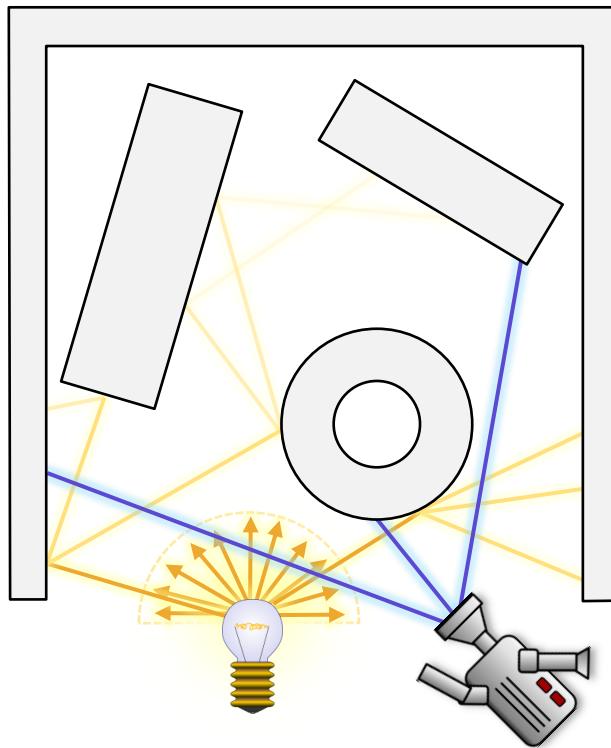


- Rasterization

- Projection
- Clipping
- Visibility

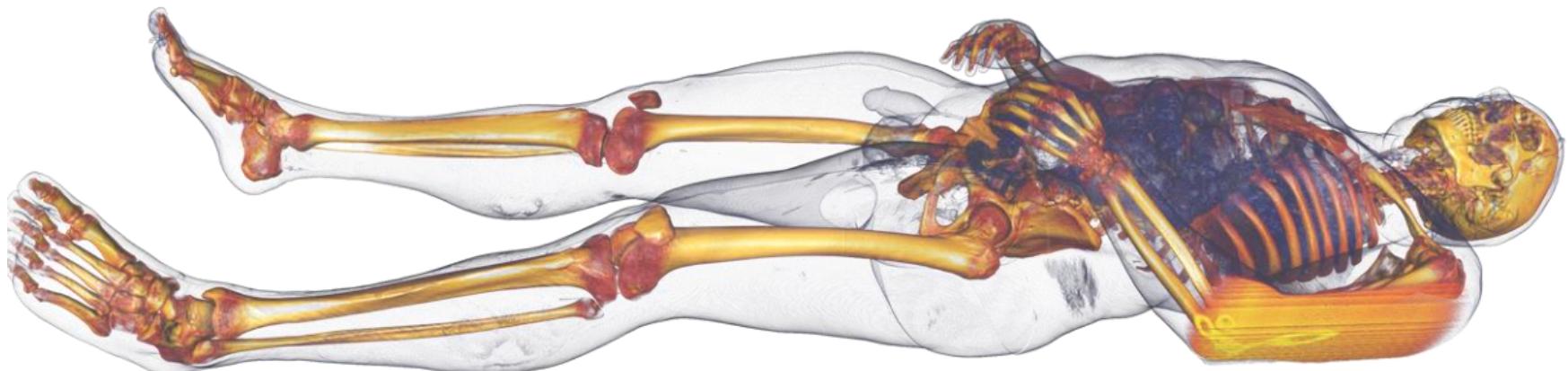


- Raytracing

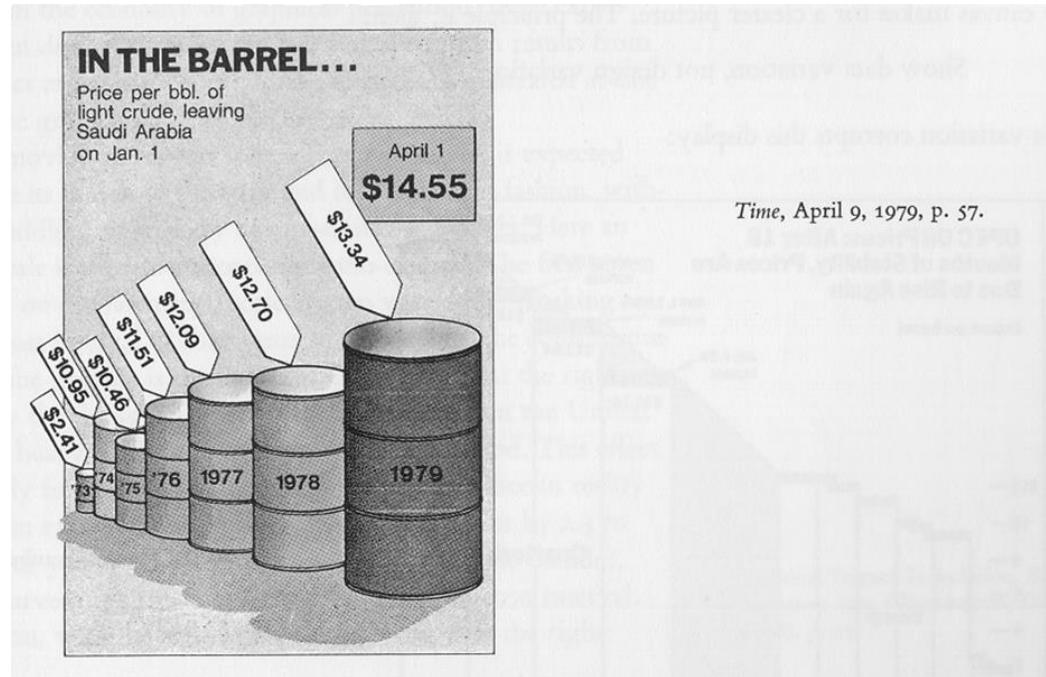


- Raycasting

- Visualization method for 3D scalar fields
- Main applications in life sciences (medicine, biology, ...)



- All-purpose visualization methods & best practices
 - Line plots, Bar plots, Histograms
 - How not to lie with visualization
- Exam preparation
 - You ask about the content of the lecture, i.e., clarifications.





Introduction to Visualization and Computer Graphics

DH2320

Prof. Dr. Tino Weinkauf

Introduction to Visualization and Computer Graphics

Overview of follow-up courses

