

# **CS 247 – Scientific Visualization**

## **Lecture 1: Introduction**

Markus Hadwiger, KAUST

# Lecture Overview



## Goals

- Basics: Learn the most important techniques in scientific visualization
- Practice: Implement scalar and vector/flow field visualization techniques in OpenGL

## Time and location

- Sunday/Wednesday, 11:00 – 12:25, Bldg 9, Room 3221.

## Course webpage:

[https://vccvisualization.org/CS247\\_Scientific\\_Visualization/](https://vccvisualization.org/CS247_Scientific_Visualization/)

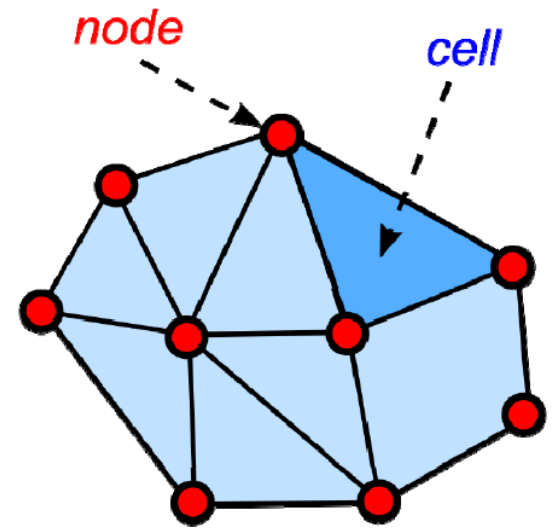
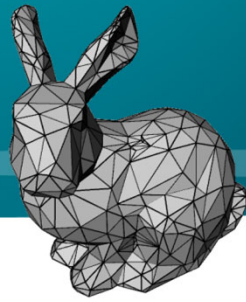
## Contact

- Markus Hadwiger `markus.hadwiger@kaust.edu.sa`
- Programming assignments `kaust.cs247@gmail.com`
  - Alberto Jaspe `alberto.jaspe@kaust.edu.sa`
  - Reem Alghamdi `reem.alghamdi@kaust.edu.sa`

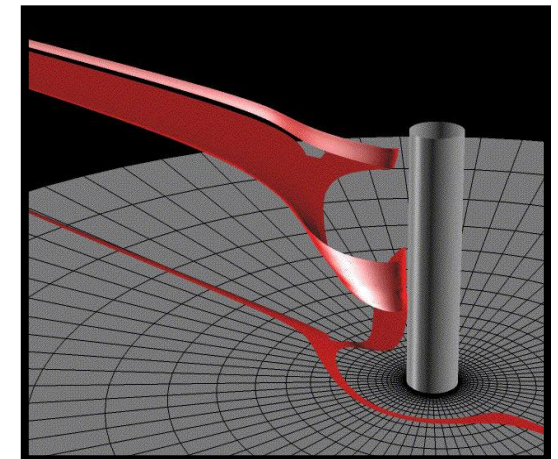
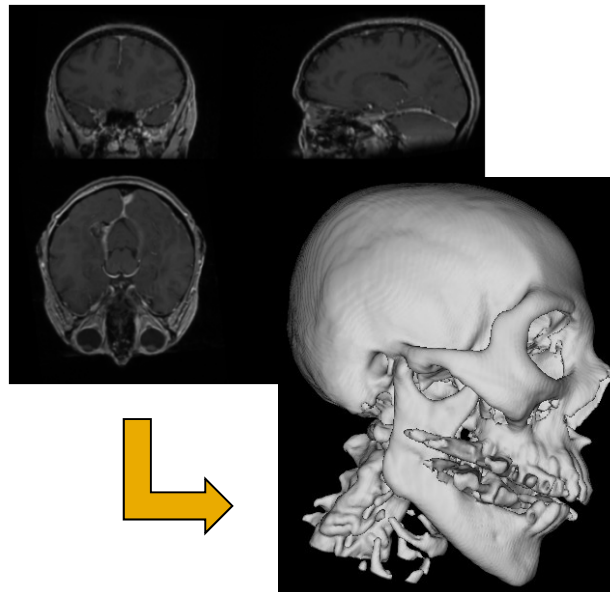
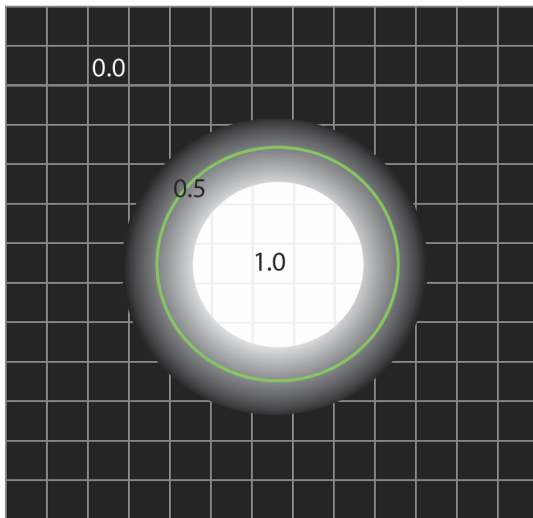
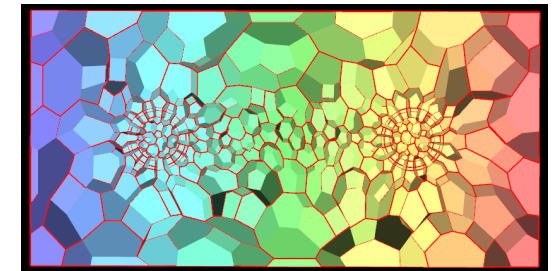
## Prerequisites

- **C/C++ programming**, computer graphics, linear algebra, multi-variable calculus
- OpenGL experience (a basic graphics course, ...) very helpful !

# Syllabus (1)



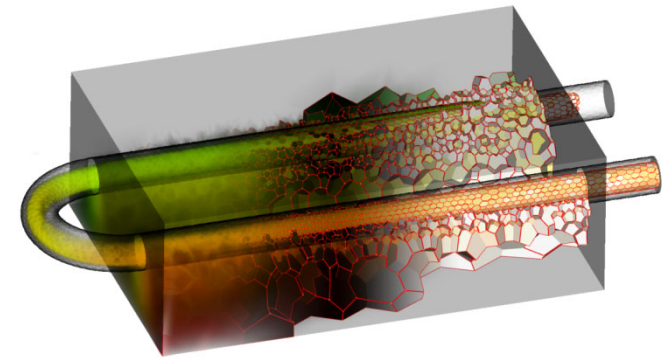
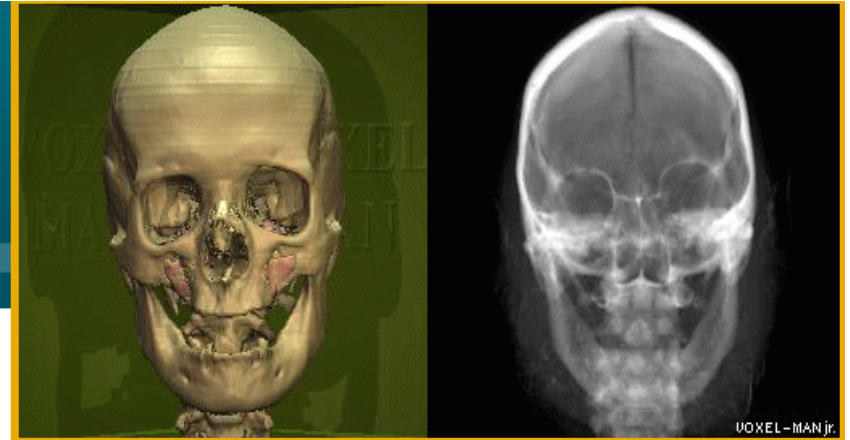
- Introduction
- Visualization basics, pipeline, and examples
- First scalar visualization example: iso-contouring
- GPU and computer graphics primer
- Data representation (grid types, data structures)



# Syllabus (2)

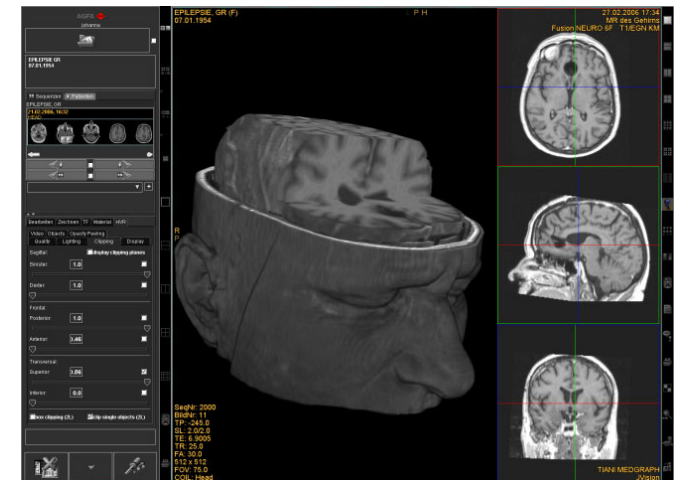
## Scalar field visualization

- Iso-surface rendering
- Volume rendering
- Transfer functions
- Volume lighting
- Unstructured grid visualization



## Applications

- Medical visualization
- Industrial CT (computed tomography)
- CFD (computational fluid dynamics) visualization of scalar quantities



# Syllabus (3)

## Vector field and flow visualization

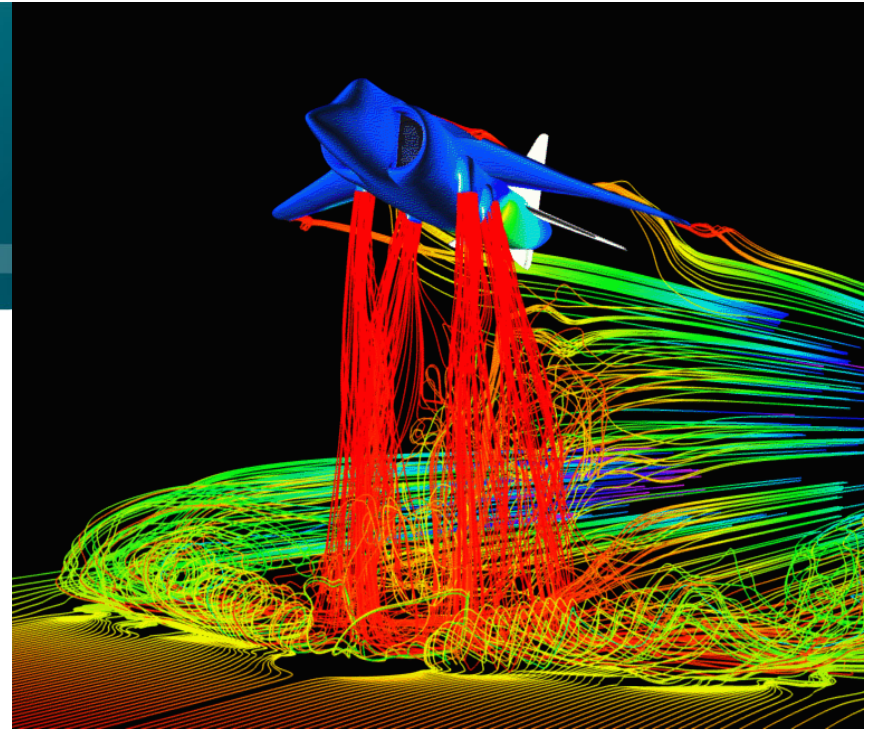
- Direct vs. indirect techniques
- Particle tracing
- Integral curves and surfaces
- Dense flow visualization techniques

## Applications

- CFD flow visualization
- Weather visualization

## If time permits

- Basic tensor visualization
- Visualization systems



# Lecture Structure and Grading



## Lectures

Weekly reading assignments (required + sometimes additional optional ones)

- Part of quiz questions (see later)

## Programming assignments

- 6+1 programming assignments; short written report + personal presentation for each

## Quizzes

- 4 quizzes, 30 min each;  
announced a week in advance, roughly every 3-4 weeks
- From lectures, (required) reading assignments, programming assignments

Grading: 60% prog. assignments; 40% quizzes

No mid-term/final exam!

# Resources



Course webpage:

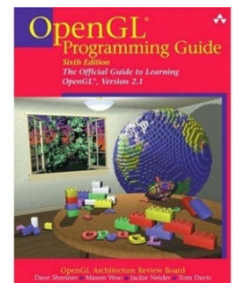
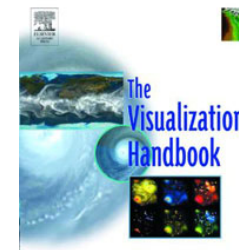
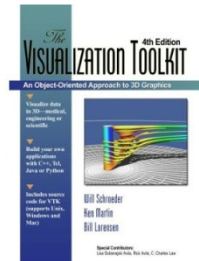
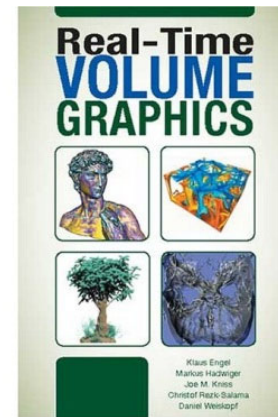
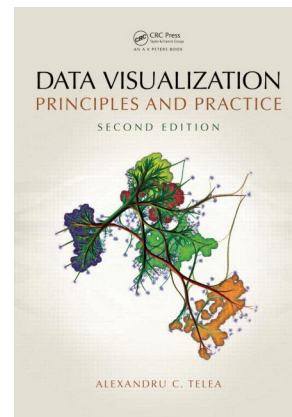
[https://vccvisualization.org/CS247\\_Scientific\\_Visualization/](https://vccvisualization.org/CS247_Scientific_Visualization/)

Textbooks:

- Data Visualization: Principles and Practice
- Real-Time Volume Graphics

Additional books:

- The Visualization Toolkit:  
An Object-Oriented Approach  
to 3D Graphics (4th Edition)
- The Visualization Handbook
- OpenGL Programming Guide (9<sup>th</sup> edition, OpenGL 4.5)  
[www.opengl.org/documentation/red\\_book/](http://www.opengl.org/documentation/red_book/)



For GPU, GPGPU, and graphics programming, also look here:

[https://vccvisualization.org/CS380\\_GPU\\_and\\_GPGPU\\_Programming/](https://vccvisualization.org/CS380_GPU_and_GPGPU_Programming/)

# Programming Assignments (1)



6 assignments (+1 introductory)

- Based on C/C++ and OpenGL
- You get a basic framework from us (in git repository)

## Organization

1. Use *git* + *github classroom* to get material and submit solution

Sign up: <https://classroom.github.com/classrooms/98274160-cs247-kaust-2023>

Tutorial: [https://www.youtube.com/watch?v=ObaFRGp\\_Eko](https://www.youtube.com/watch?v=ObaFRGp_Eko)

2. Assignment info and framework in git repository

3. Submit solution and report via git by submission deadline

4. Personal (online) presentation after submission



# Programming Assignments (2)



- Submit via *git* at the latest on day the assignment is due (code, libs, everything that is needed to run your program)
- Submission must include short report (2 pages, pdf), including short explanation of algorithms, your solution, problems, how to run it, screenshots
- Personal presentations:  
Present your program live and explain source code (10-15 min)
  - Sign up for presentation slot in advance (doodle)
  - Present via Zoom

# Programming Assignments (3)



## Grading

- Submission complete, code working for all the required features
- Documentation complete (report, but also source code comments!)
- Personal presentation
- Optional features, coding style, clean solution
- Every day of late submission reduces points by 10%
- No copies from the Internet (or anywhere else)!  
You have to do it yourself and understand what you program:  
your explanations during the presentations will be part of the grade!

# Programming Assignments (4)



General contact: `kaust.cs247@gmail.com`

Teaching Assistants:

- Alberto Jaspe (`alberto.jaspe@kaust.edu.sa`)  
– main contact for assignments; assignment presentations
- Reem Alghamdi (`reem.alghamdi@kaust.edu.sa`)  
– help with programming questions



Help in programming assignments (in this order!):

1. **Think about it, read about it, google it!**
2. **Discuss on Piazza:** `http://piazza.com/kaust.edu.sa/spring2023/cs247`
3. **Ask TAs:** `kaust.cs247@gmail.com` (Alberto, Reem)

# Programming Assignments Schedule (tentative)



Assignment 0:	Lab sign-up: setup piazza + github account, get git repo Basic OpenGL example [we will offer a tutorial!]	until	<b>Jan 29</b>
Assignment 1:	Volume slice viewer	until	<b>Feb 12</b>
Assignment 2:	Iso-contours (marching squares)	until	<b>Feb 26</b>
Assignment 3:	Iso-surface rendering (marching cubes)	until	<b>Mar 19</b>
Assignment 4:	Volume ray-casting, part 1	until	<b>Apr 2</b>
	Volume ray-casting, part 2	until	<b>Apr 9</b>
Assignment 5:	Flow vis, part 1 (hedgehog plots, streamlines, pathlines)	until	<b>Apr 30</b>
Assignment 6:	Flow vis, part 2 (LIC with color coding)	until	<b>May 10</b>

# Reading Assignment #1 (until Jan 29)



**Sign up for piazza!**

`http://piazza.com/kaust.edu.sa/spring2023/cs247`

Read (required):

- Data Visualization book, Chapter 1
- Data Visualization book, Chapter 2 until 2.3 (inclusive)
- Download and look at:  
NIH/NSF Visualization Research Challenges report

`http://gvi.seas.harvard.edu/sites/all/files/NIH-NSF-VRC-Report.pdf`

- Start familiarizing yourself with OpenGL if you do not know it !

# Thank you.

Thanks for material

- Helwig Hauser
- Eduard Gröller
- Daniel Weiskopf
- Torsten Möller
- Ronny Peikert
- Philipp Muigg
- Christof Rezk-Salama