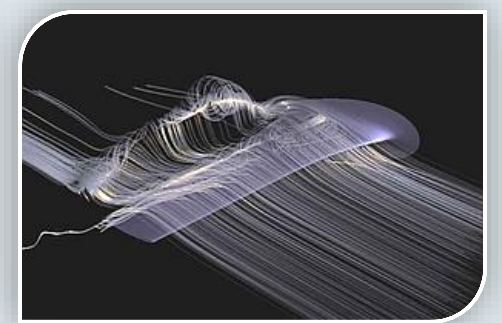



Visual Data Analytics

Introduction

Dr. Johannes Kehrer – Siemens AG, Munich

Disclaimer

These lecture slides and materials are for personal use by students only and are intended for educational purposes only. Distribution or reproduction of the material for any other purposes is prohibited.

The views and opinions expressed by any individual during the course of these lecturers are their own and do not necessarily represent the views, opinions, or positions of Siemens or the Technical University of Munich.

Although reasonable efforts have been made to ensure the information contained in these lecture slides is current and accurate, we do not assume any responsibility for the content of the posted material.

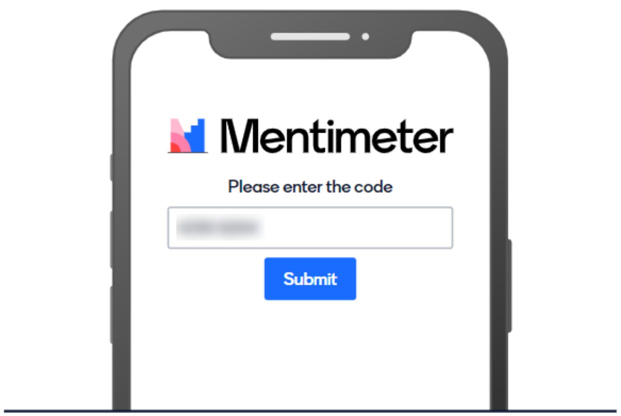
These slides also contain links to websites of third parties. As the content of these websites is not under our control, we cannot assume any liability for such external content.

Any comments/questions?

SIEMENS
Ingenuity for life

Go to

www.menti.com



Enter the code

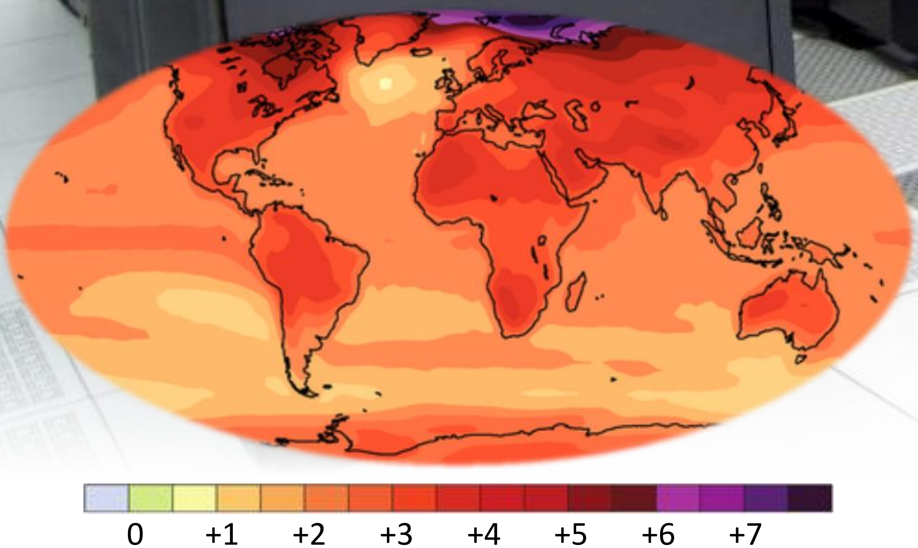
54 11 67 0

Or use QR code


LARGE AM0UNT5 0F DATA
ARE GENERATED EVERYWHERE!

Medical scanners

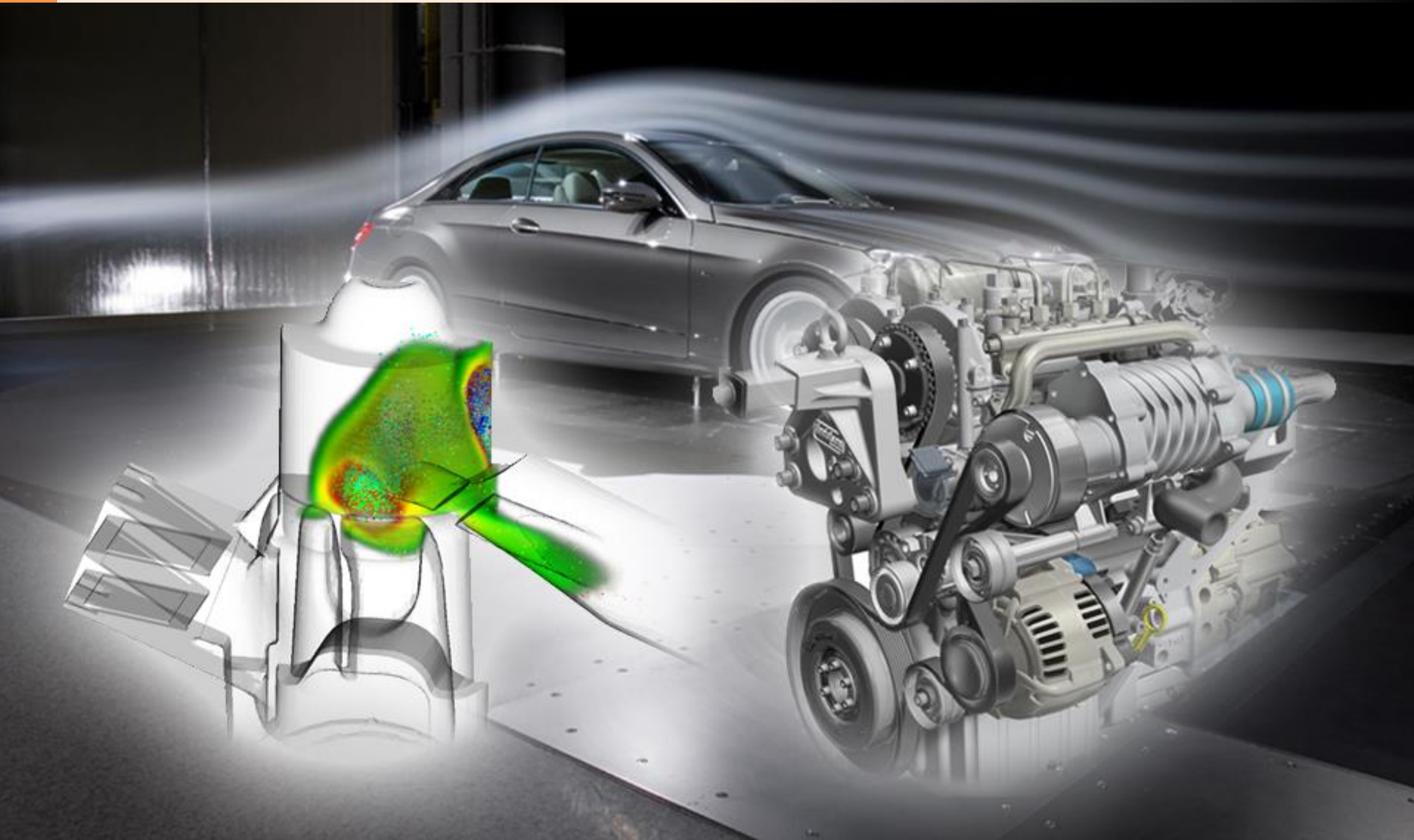
Weather & Climate Simulations



Changes in Temperature 2090-2099 (°C)

Digital Industries

Automotive Engineering



Social networks, emails, blogs, wikis, etc.

to vis·u·al·ize

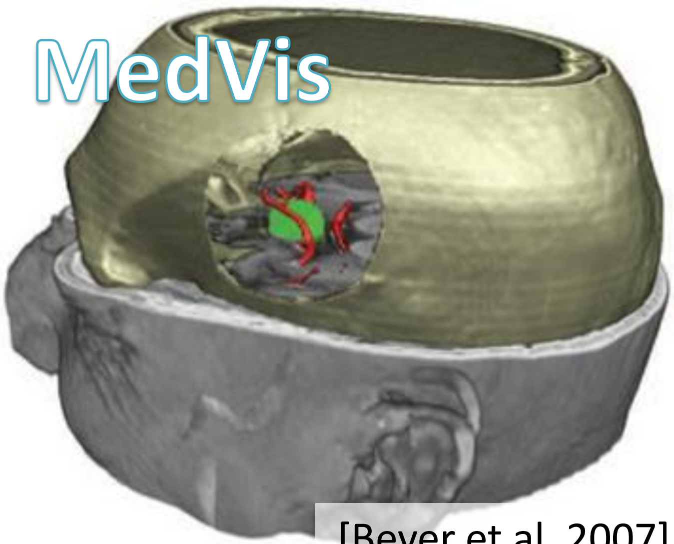
to form a **mental vision, image, or picture** of (something not visible or present to sight, or of an abstraction);

to make **visible to the mind or imagination**.

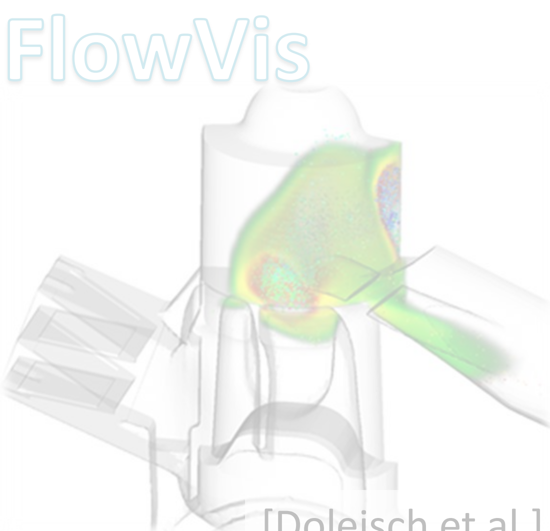
[Oxford English Dictionary]

Visualization

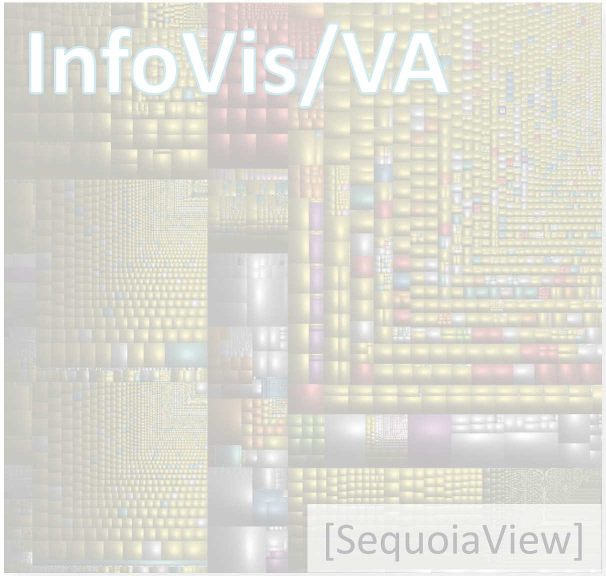
The use of computer-supported, interactive, visual representations of data to amplify cognition



[Beyer et al. 2007]



[Doleisch et al.]

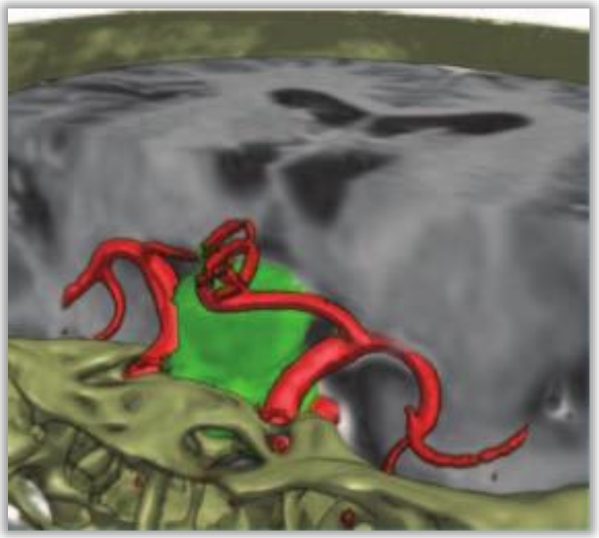


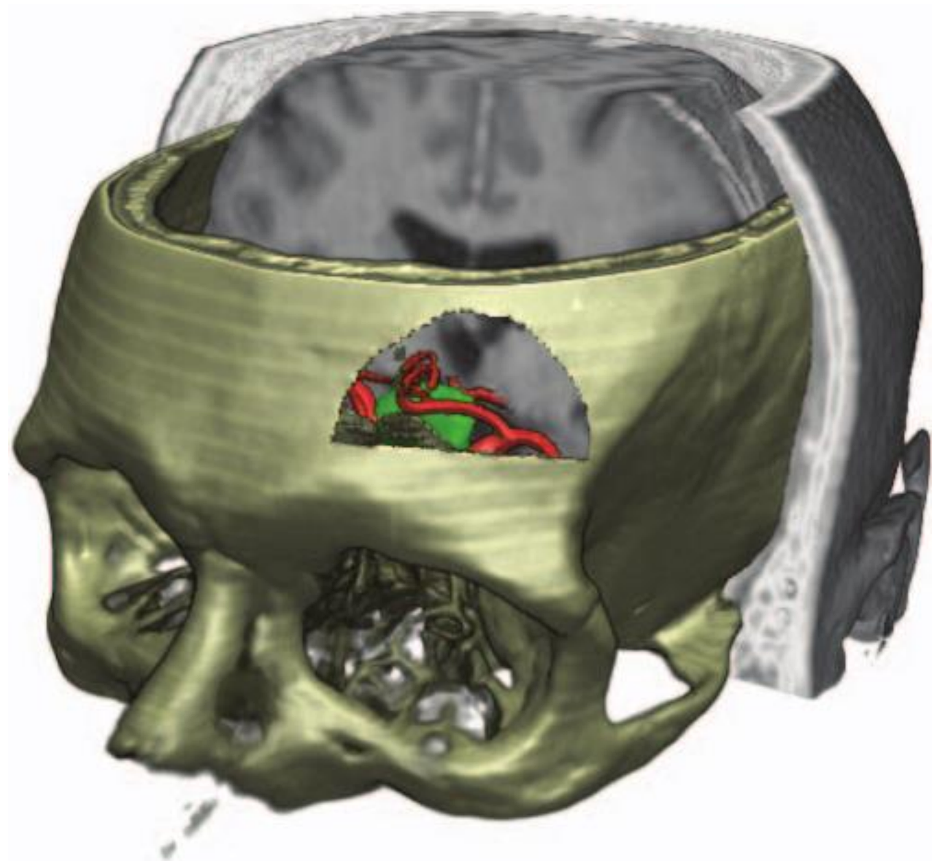
[SequoiaView]

Medical Visualization – Examples

SIEMENS
Ingenuity for life

- Preoperative planning of a tumor resection





[Beyer et al. 2007]

Black/white: brain – Magnetic Resonance Imaging (MRI)
Green: tumor – MRI
Red: vessels – Magnetic Resonance Angiogram (MRA)
Brown: skull – Computer Tomography (CT)

Medical Visualization – Examples

- Virtual fetoscopy (4D Ultrasound)

GE Healthcare Voluson HDlive

Medical Visualization – Examples

SIEMENS
Ingenuity for life

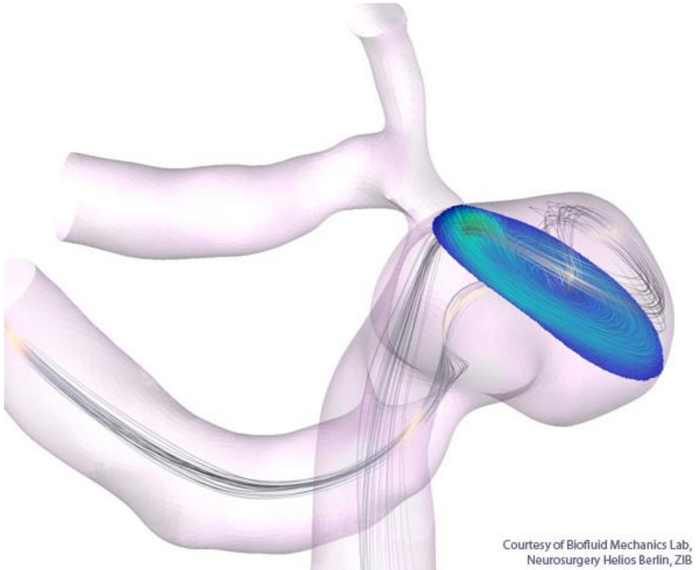
- Virtual fetoscopy (4D Ultrasound)
  - Enhance clinical confidence
  - User interest & involvement

GE Healthcare Voluson HDlive

https://youtu.be/BD7quHKgEuk

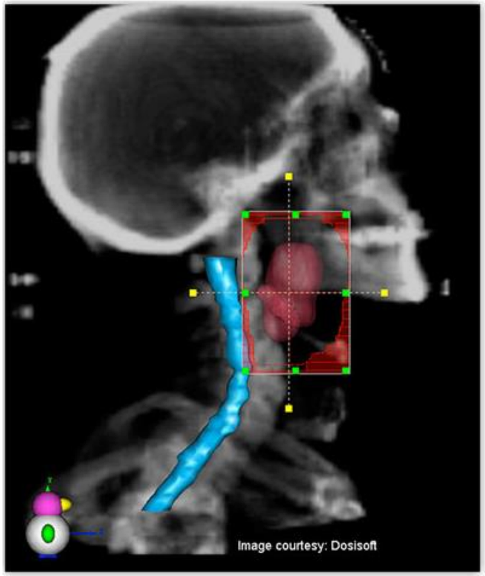
Medical Visualization – Examples

SIEMENS
Ingenuity for life

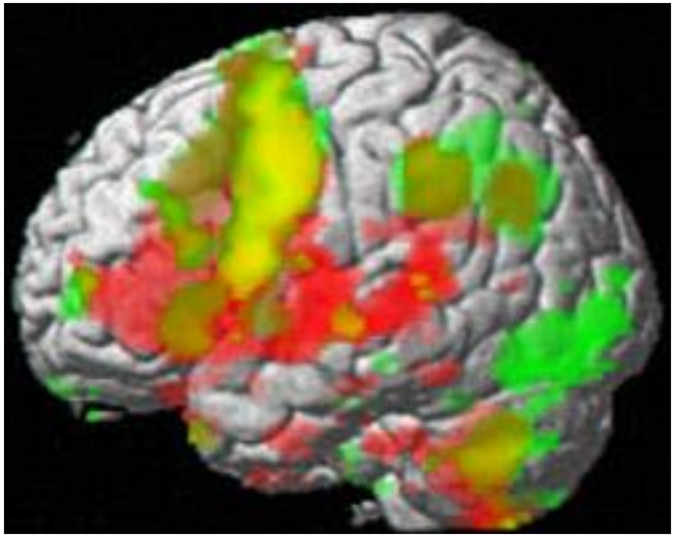


blood flow in aneurysm

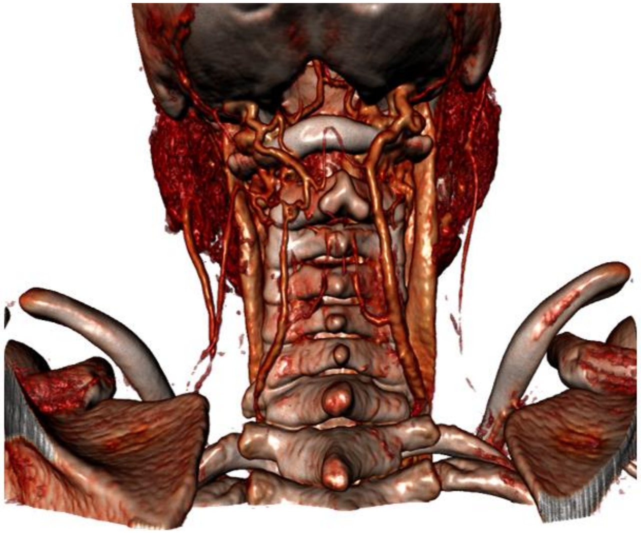
bone tissue density



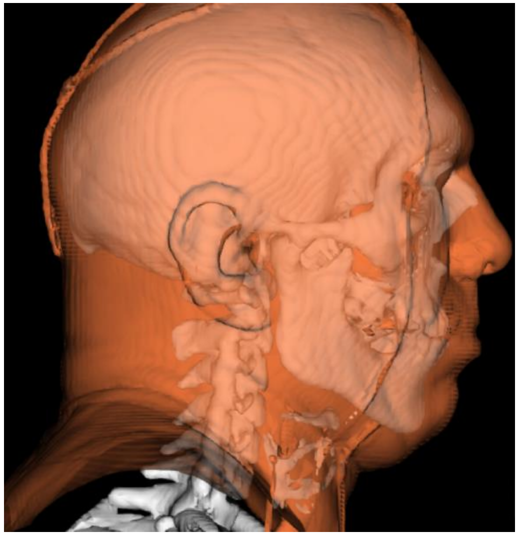
surgery planning



brain activity (fMRI)



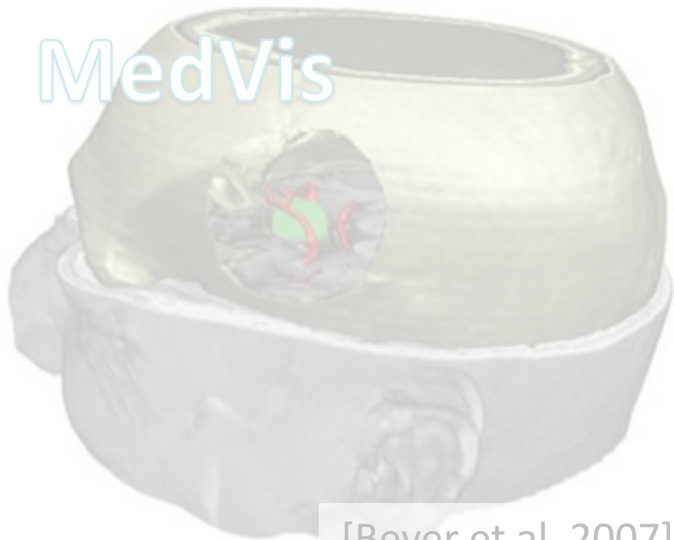
MRI scan - tissues



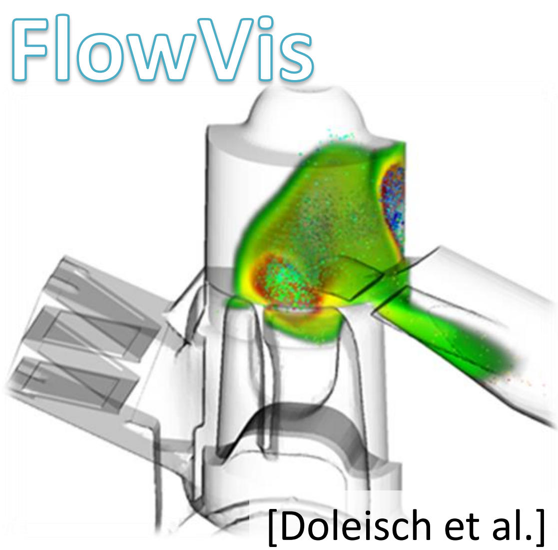
bone + skin surface

Visualization

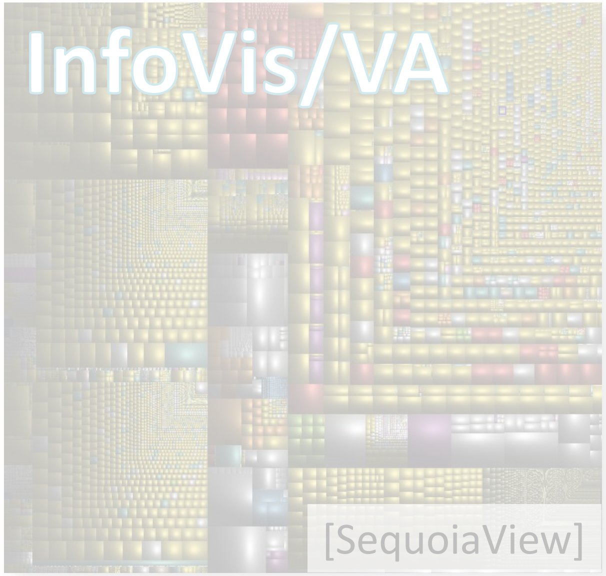
The use of computer-supported, interactive, visual representations of data to amplify cognition



[Beyer et al. 2007]



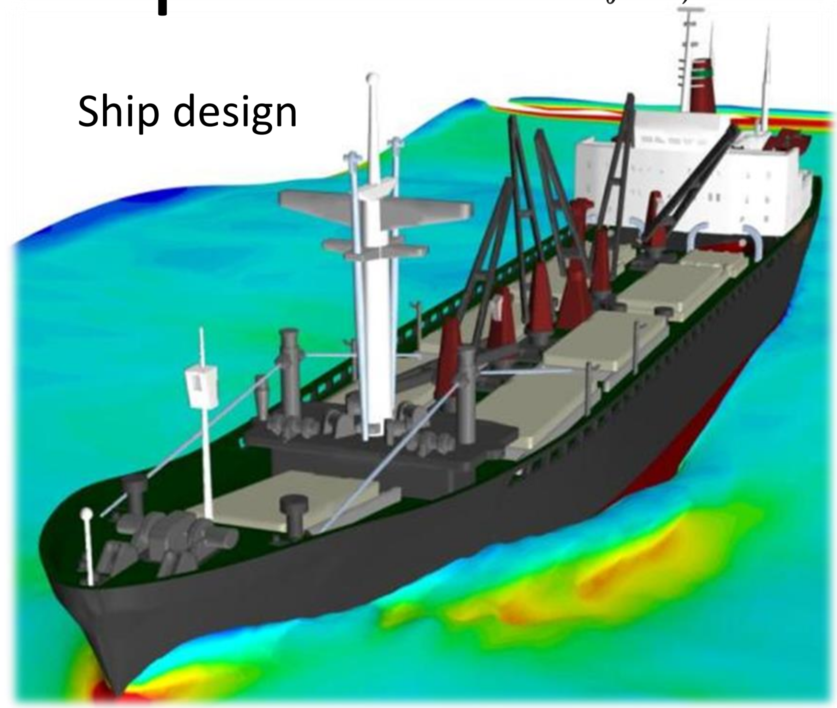
[Doleisch et al.]

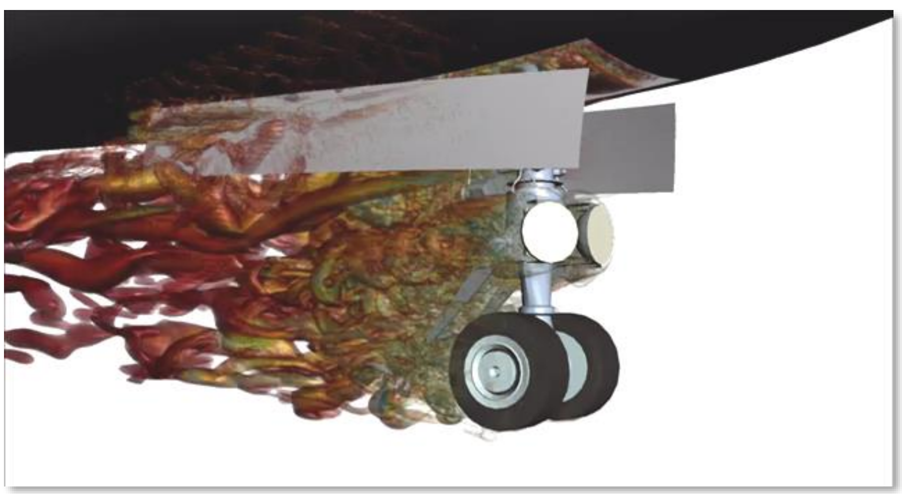


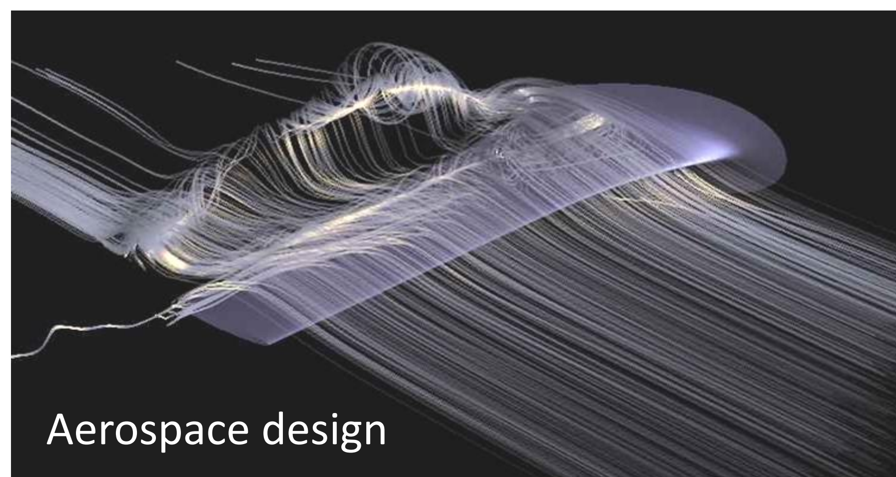
[SequoiaView]

Flow Visualization – Examples

SIEMENS
Ingenuity for life

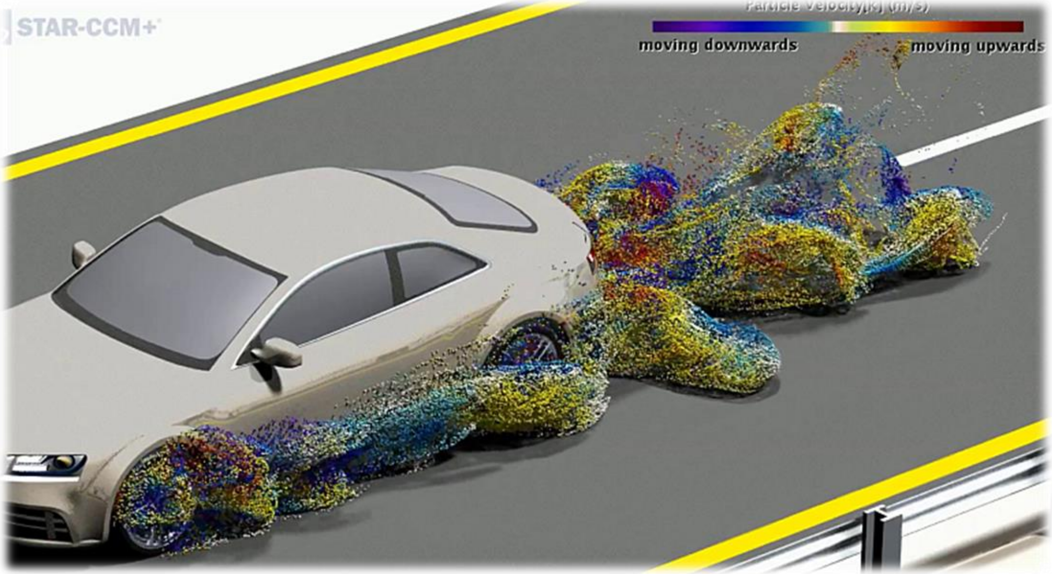


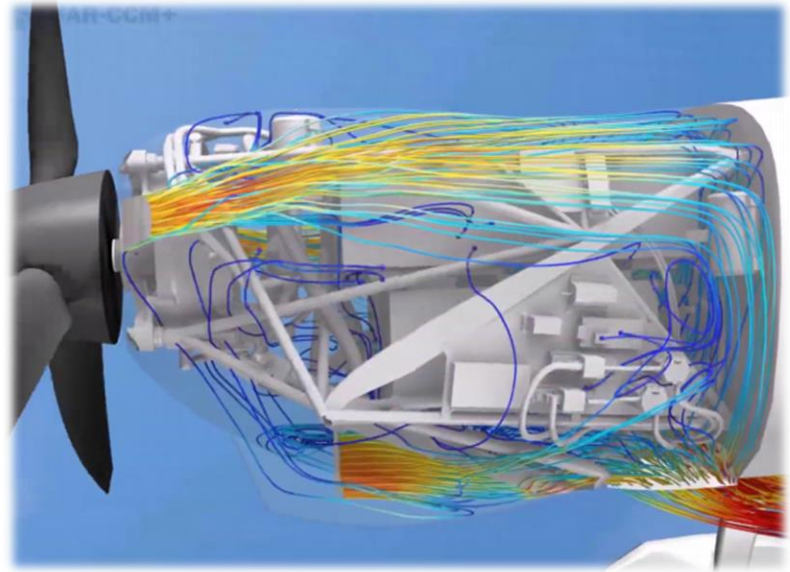


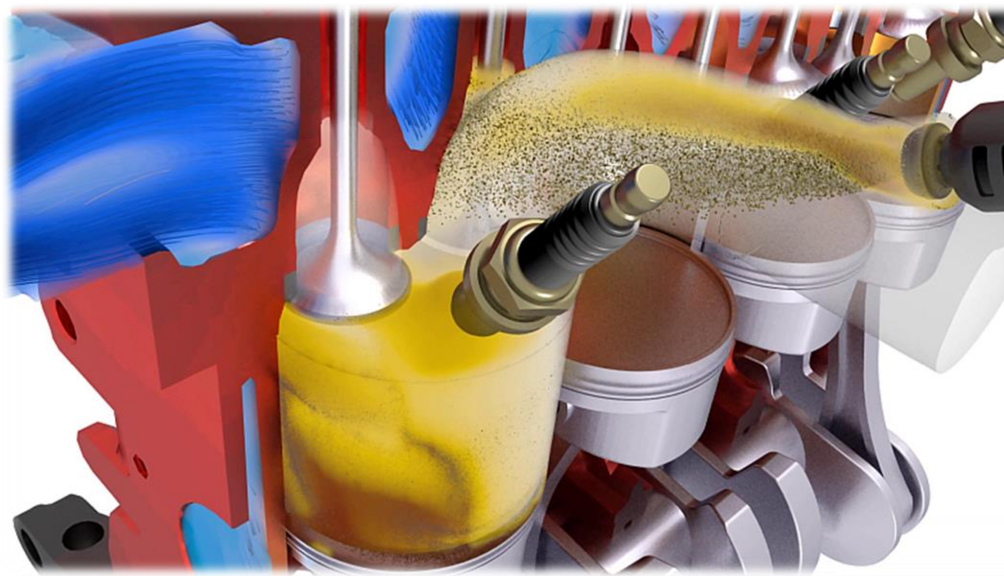


Flow Visualization – Examples

SIEMENS
Ingenuity for life





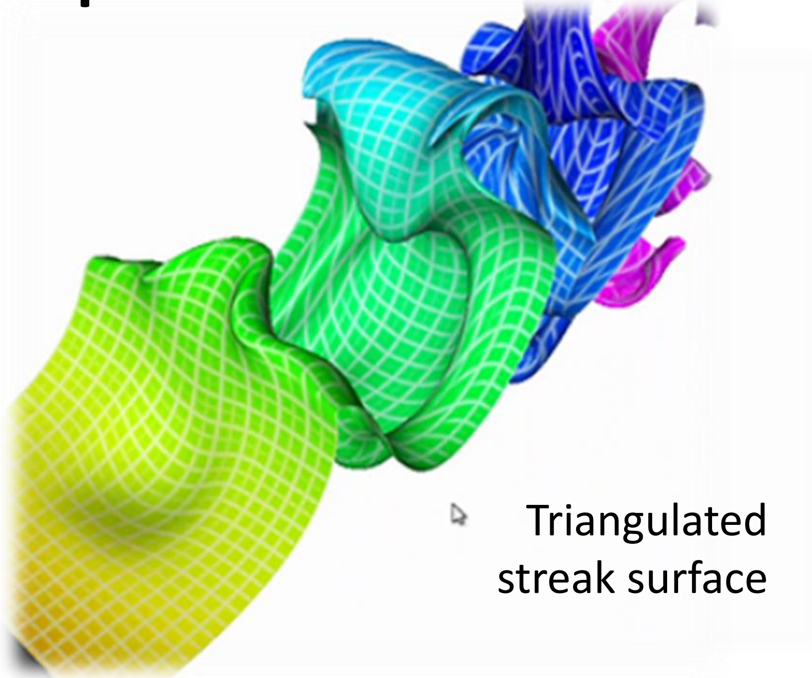


Simcenter STAR-CCM+
https://youtu.be/443kbDFPjUo

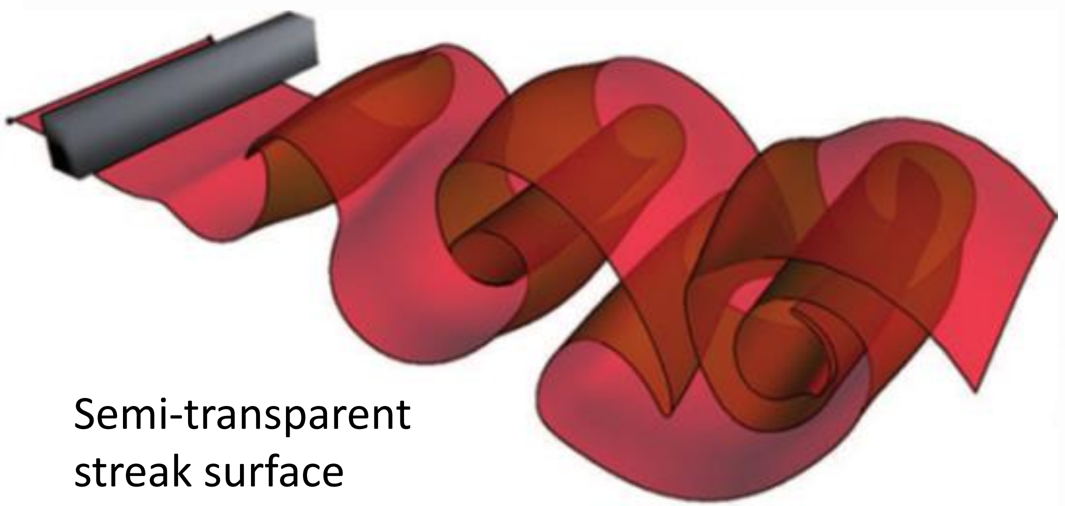
Flow Visualization – Examples

SIEMENS
Ingenuity for life

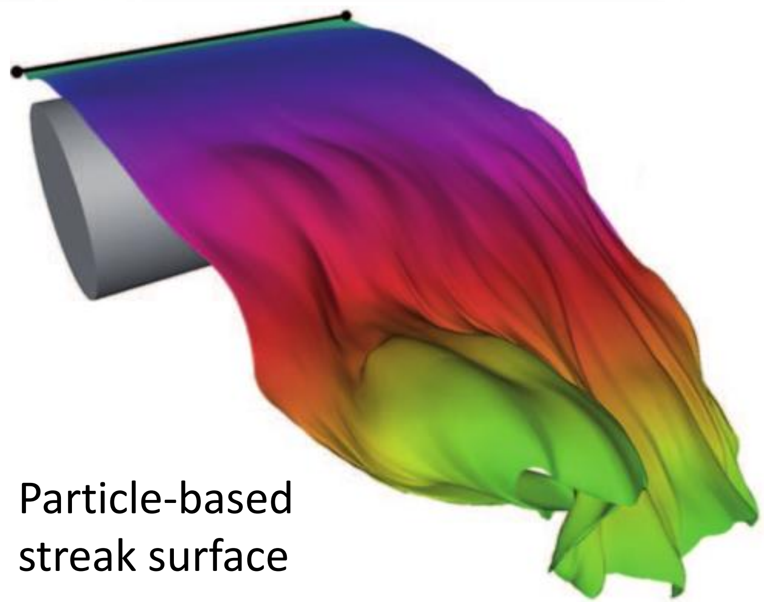
- Visualization of complex flows



Triangulated streak surface



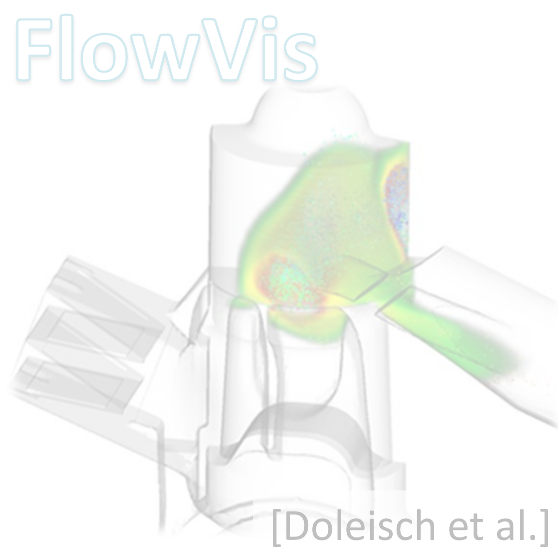
Semi-transparent streak surface

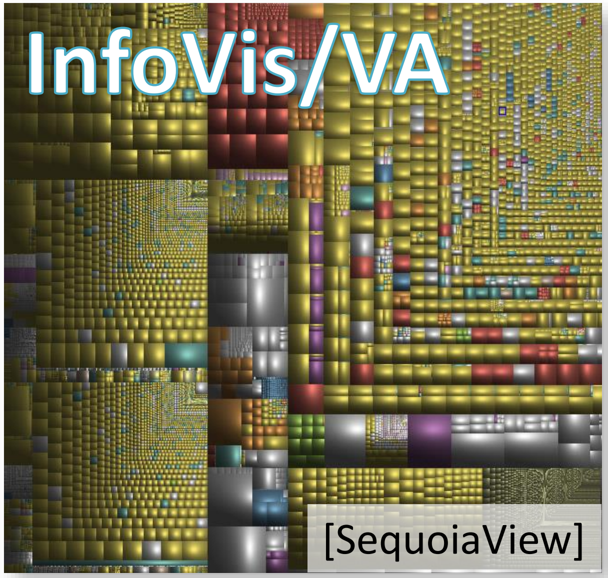


Particle-based streak surface

Visualization

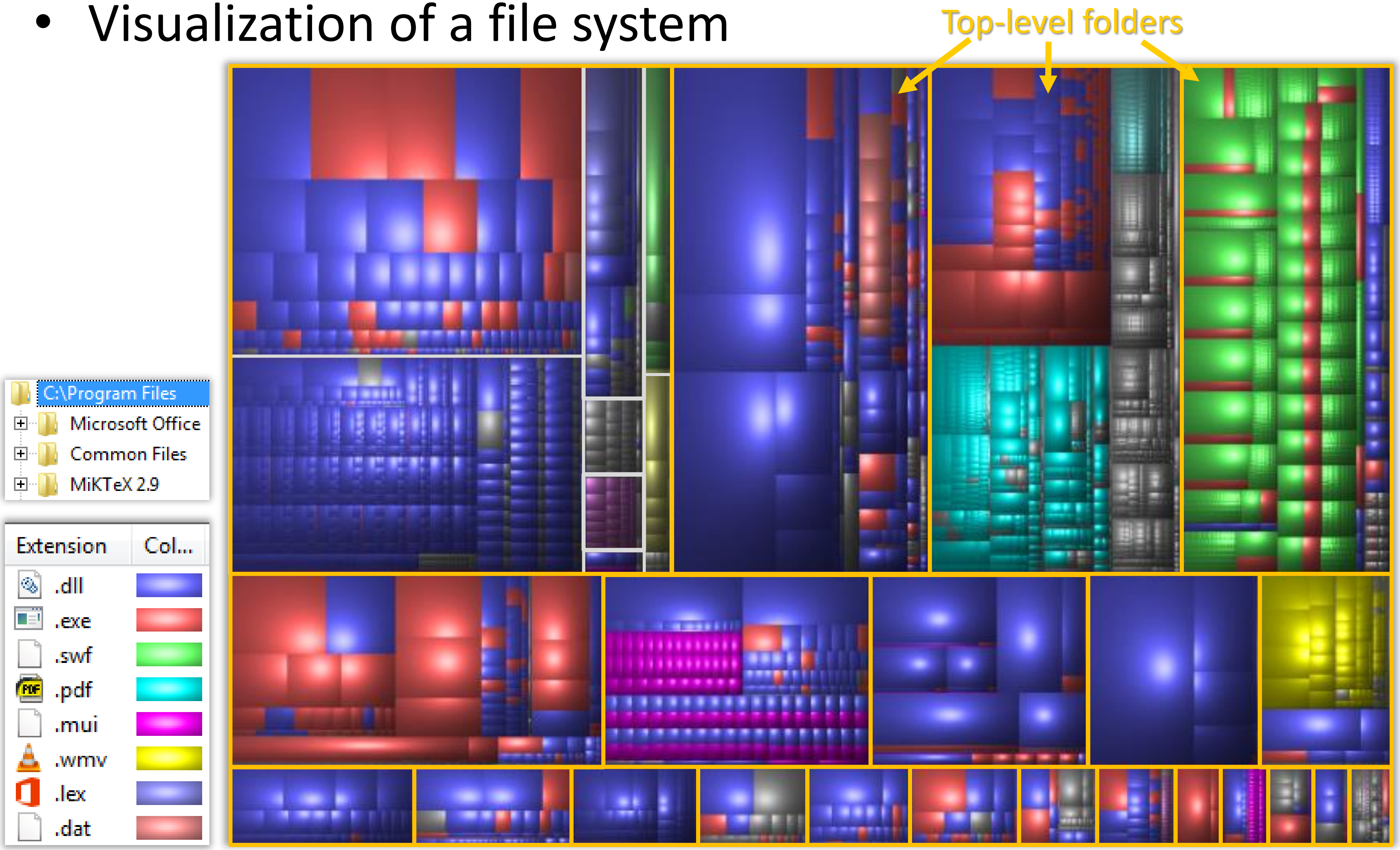
The use of computer-supported, interactive, visual representations of data to amplify cognition





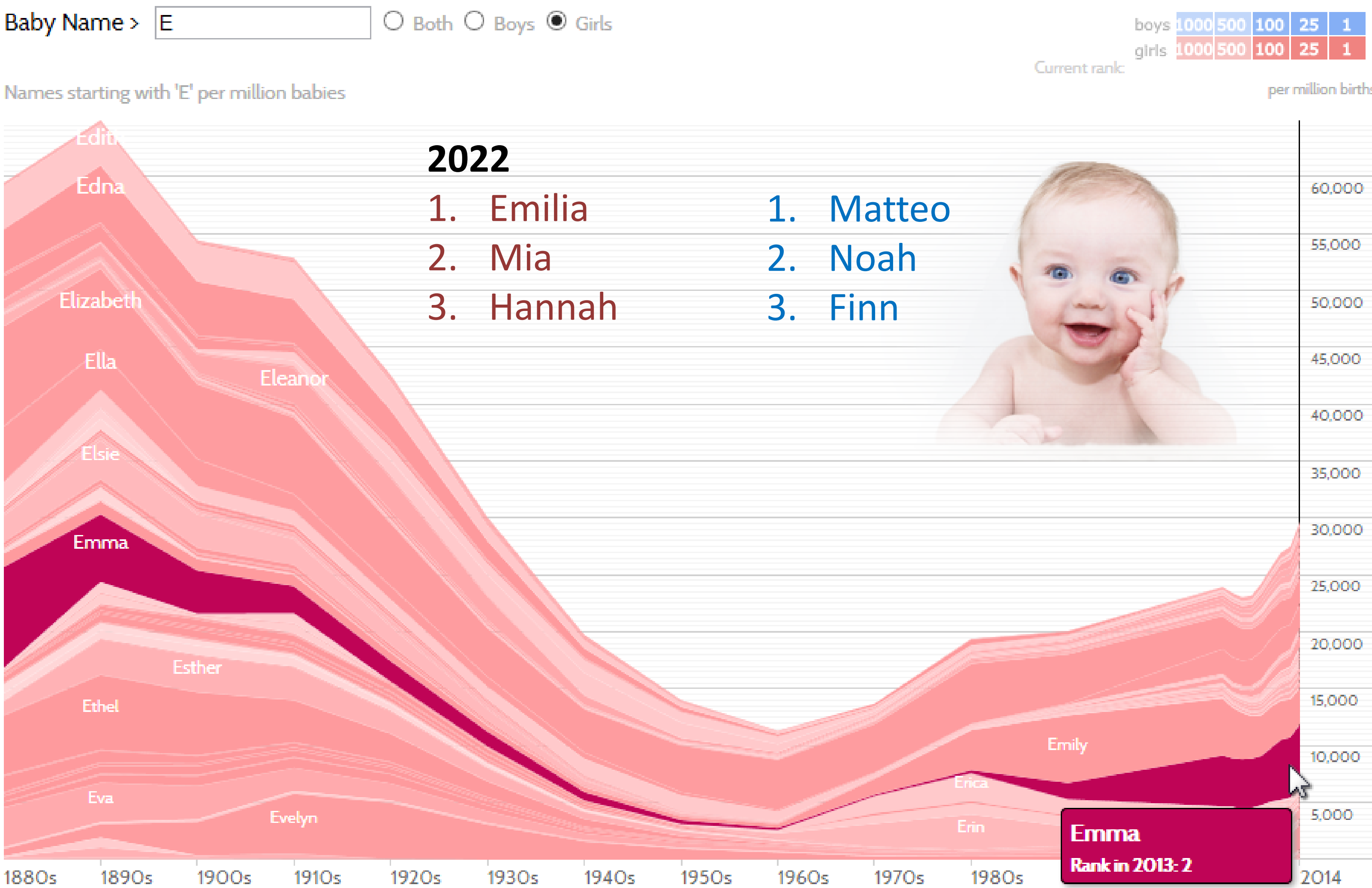
InfoVis/VA – Examples

- Visualization of a file system



InfoVis/VA – Examples

- NameGrapher



2022

1. Emilia
2. Mia
3. Hannah

1. Matteo
2. Noah
3. Finn

M. Wattenberg, namerology.com/baby-name-grapher/

Preventive Maintenance using AI

SIEMENS
Ingenuity for life

- Sensor data are collected in many fields to monitor operation, e.g., car manufacturing, oil & gas production, heavy industries, automation systems
- AI methods can help to detect abnormal behavior, prevent failures, and optimize operation
- Technical interpretation by domain experts needed

Why Visualization?

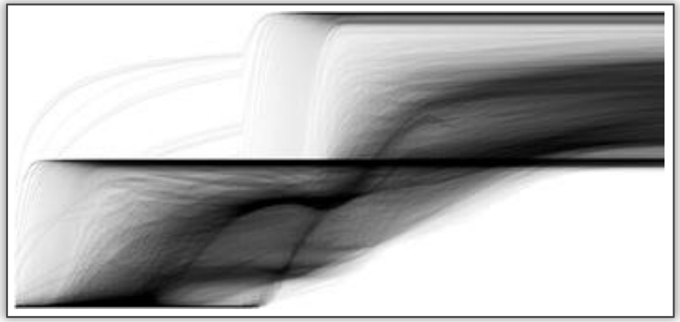
“A picture is worth a thousand words”

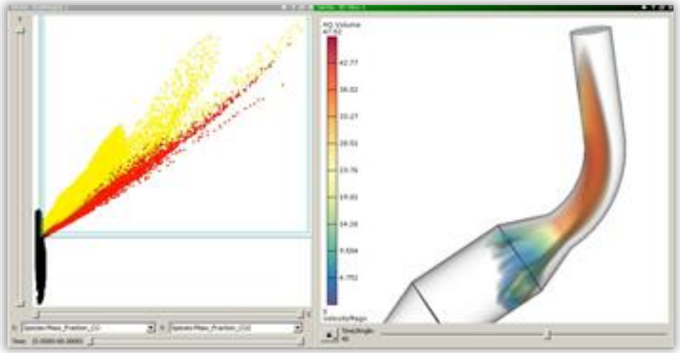
“The purpose of visualization is **insight**, not pictures.”
[B. Shneiderman]

- Vision is the dominant sense for acquiring information from our world
- Nearly $^1/_3$ of our brain is devoted to processing visual information (8% for touch, 3% for hearing)

Why visualization matters?

- Visualization …
  - lets you see things that would rather go unnoticed (data trends, outliers, dependencies, etc.)
  - gives answers faster
  - lets you interact with your data, study causes and effects, etc.
  - helps to deal with increasing size and diversity of data
  - produces pretty, informative, & interactive pictures





www.targetprocess.com/articles/information-visualization/

Big Data and Visualization

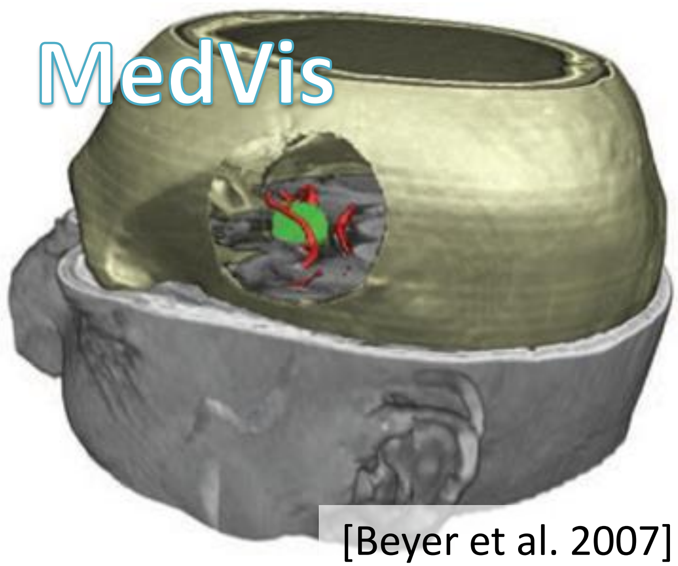
SIEMENS
Ingenuity for life

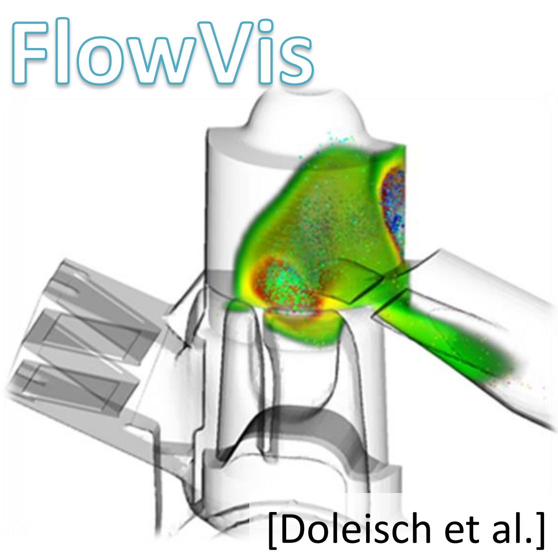
"The ability to take data—to be able to **understand** it, to **process** it, to **extract value** from it, to **visualize** it, to **communicate** it—that's going to be a hugely important skill in the next decades, … because now we really do have **essentially free and ubiquitous data**."

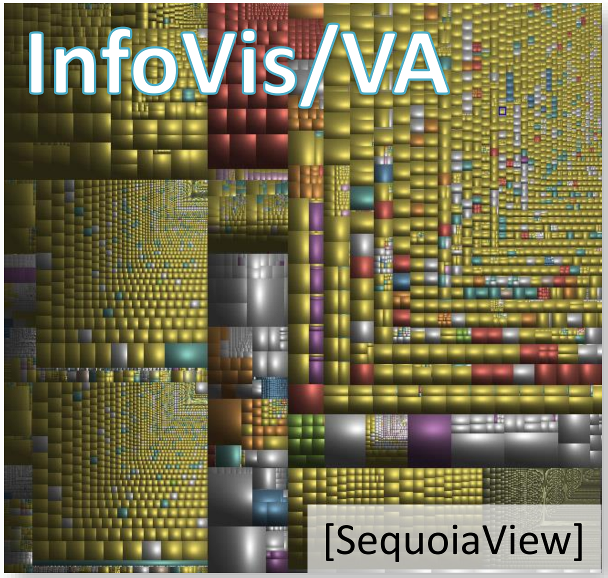
Hal Varian, Google's Chief Economist
The McKinsey Quarterly, Jan 2009

This course – goals

- Understand basic visualization algorithms in different domains (flow/volume visualization, visual analytics)
- Learn which techniques to use for which type of data
- Practical exercise: experiment with visualization software systems on your own initiative

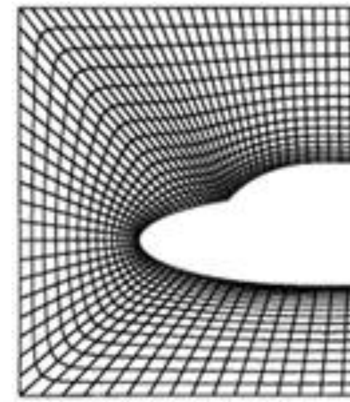


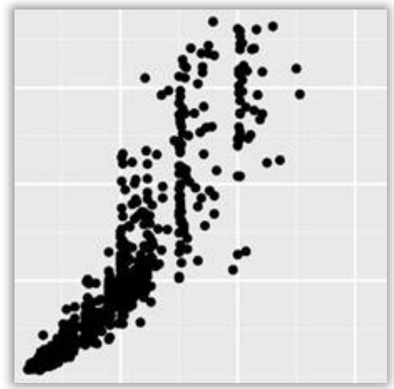


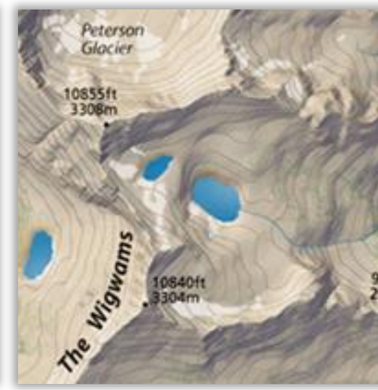


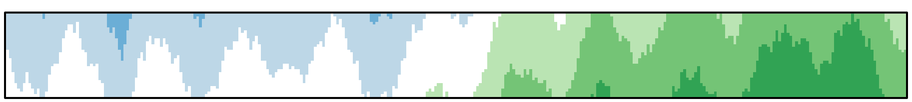
This course – overview part I

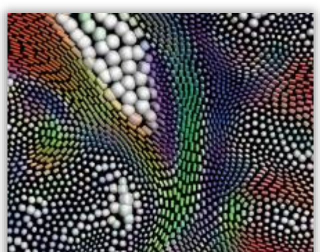
- Fundamentals of data visualization
  - Visualization examples, data sources, vis pipeline, data representation and classification
- Data reconstruction
  - Scattered data interpolation, triangulation, grids, cell-wise interpolation
- Basic data mapping techniques
  - Color coding, scatter plots, glyphs, diagrams, contour lines, etc.

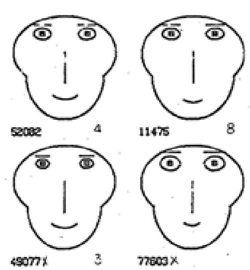


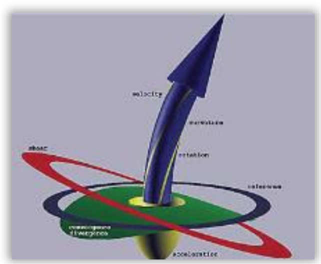


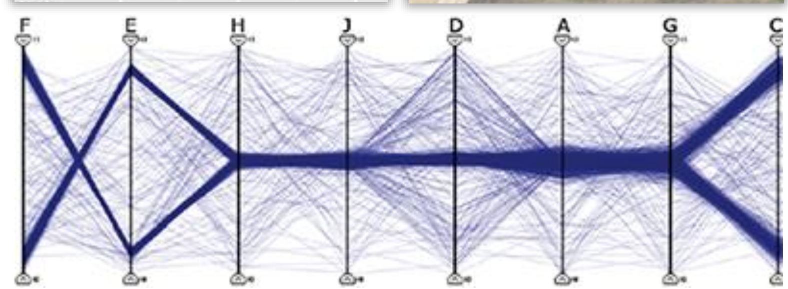




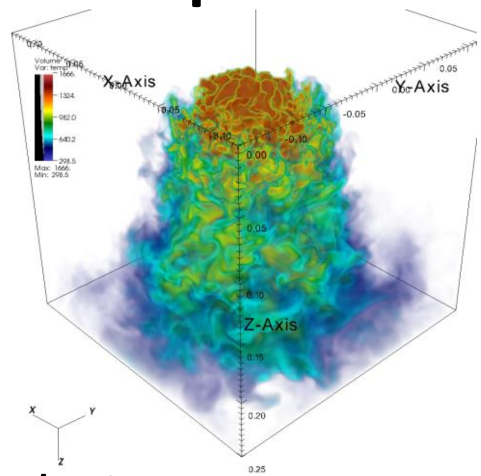


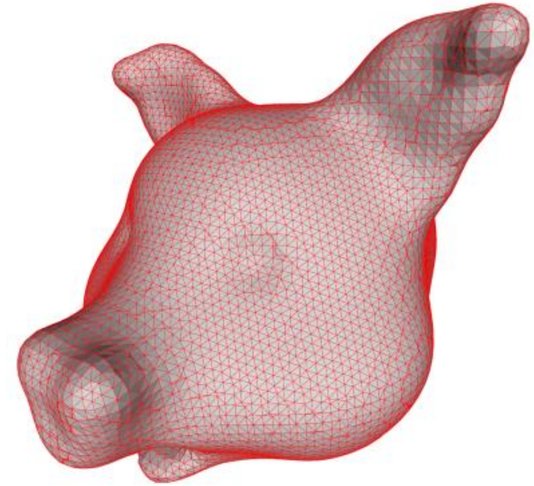




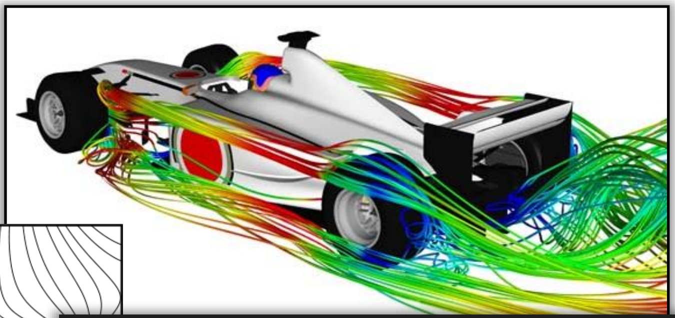


This course – overview part II

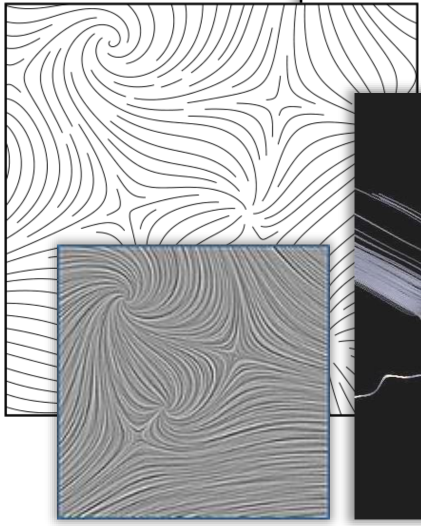


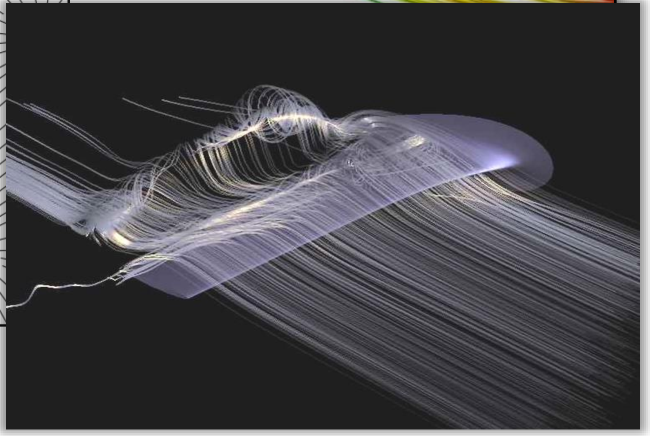


- Volume visualization
  - Data acquisition
  - Iso-surface rendering
  - Lighting and shading
  - Direct volume rendering techniques
  - Classification via transfer functions



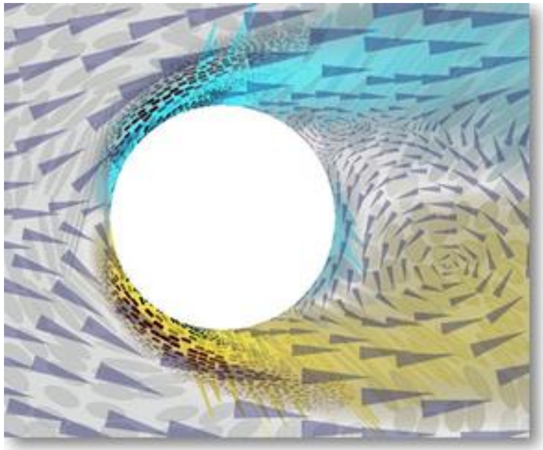
- Flow visualization
  - Characteristic lines
  - Particle tracing
  - Vector field topology
  - Line integral convolution

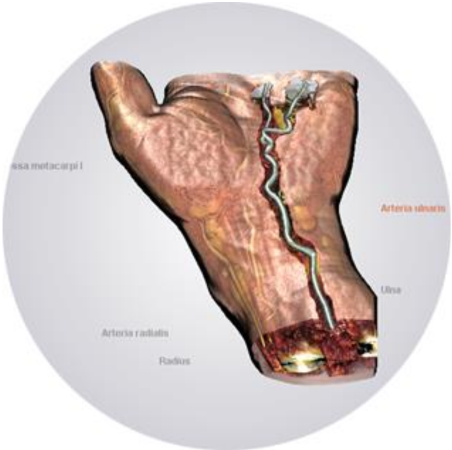


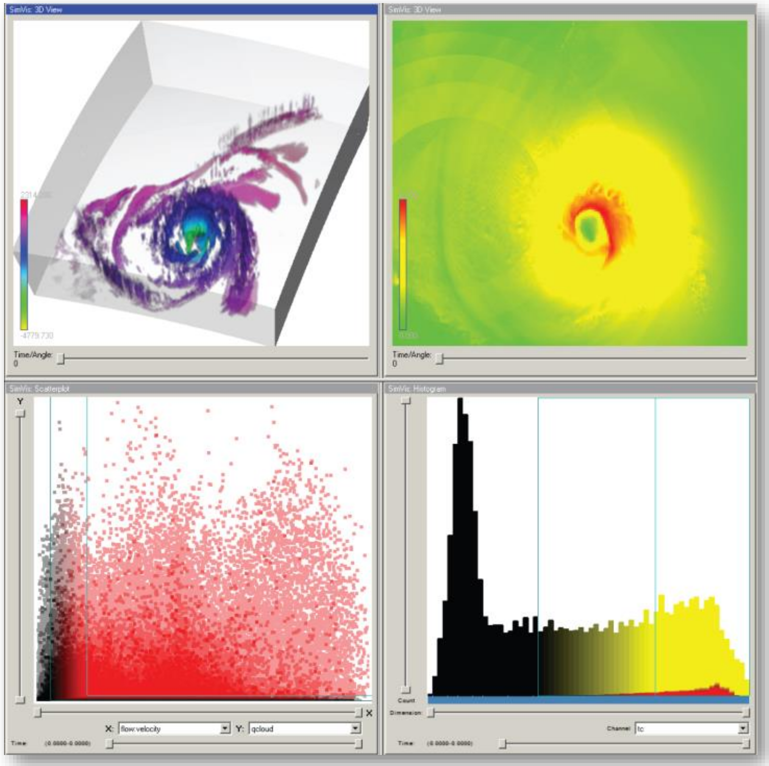


This course – overview part III

- Visual Analysis/Analytics of Scientific Data
  - Combination of visual and computational analysis methods
  - Coordinated multiple views, linking & brushing, focus+context visualization, etc.
  - Spatiotemporal, multi-variate, multi-modal, & multi-run data







This course

- Announcements, slides, exercise via Moodle: https://www.moodle.tum.de/course/view.php?id=80170
- Lectures will be both **on-campus** and **streamed online** via TUM-Live (livestream & video on demand)
- You can ask questions regarding the current lecture via Menti or the Moodle question forum

The course is self-contained

This course – prerequisites

- Analysis, Linear Algebra
- No programming required
- Willingness to learn new software & tools
  - This can be time consuming
  - You will need to build skills by yourself

This course – practical exercise

SIEMENS
Ingenuity for life

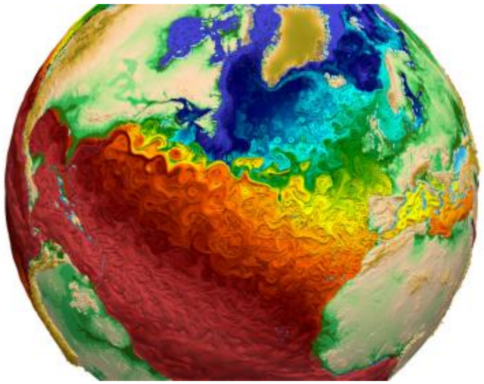
- Experiment with vis. software on your own initiative
- You get several data sets
- You use different techniques to visualize these data sets in ParaView, ImageVis3D and/or Tableau
- Until the end of the semester, you wrap up your experience in a 3-5 page summary, including images and explanations of what you have done

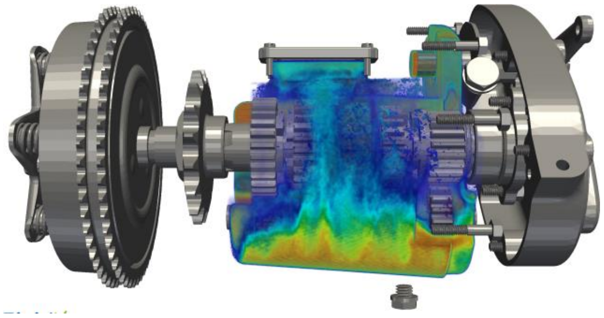
- Practical exercise not mandatory, **but**
  - you can obtain a bonus of 0.3 on a successfully passed exam, if the practical exercise has been passed
  - there can be questions about ParaView, ImageVis3D & Tableau at the exam!

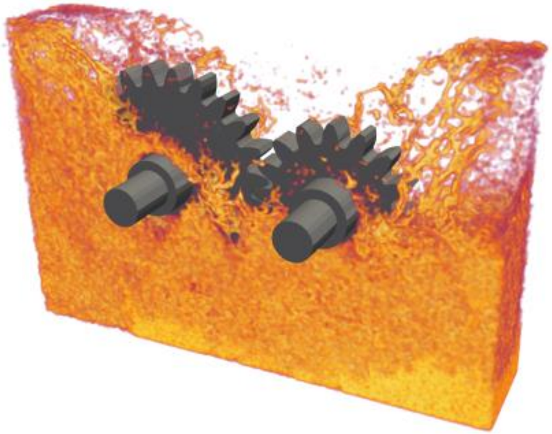
ParaView

http://www.paraview.org/

- History
  - 2000 collaborative project between Kitware Inc. and Los Alamos National Laboratory
  - 2002 first release
- Supports many platforms (Windows, Linux, etc.)
- Supports parallel computer architectures
- Supports many different visualization techniques







ImageVis3D

http://en.wikipedia.org/wiki/ImageVis3D

- History
  - Project at the Scientific Computing and Imaging Institute at the University of Utah
  - Roots at the Chair for Computer Graphics and Visualization at TUM
- Supports many platforms (Windows, Linux, etc.)
- Graphics hardware acceleration for volume visualization
- Supports large data sets

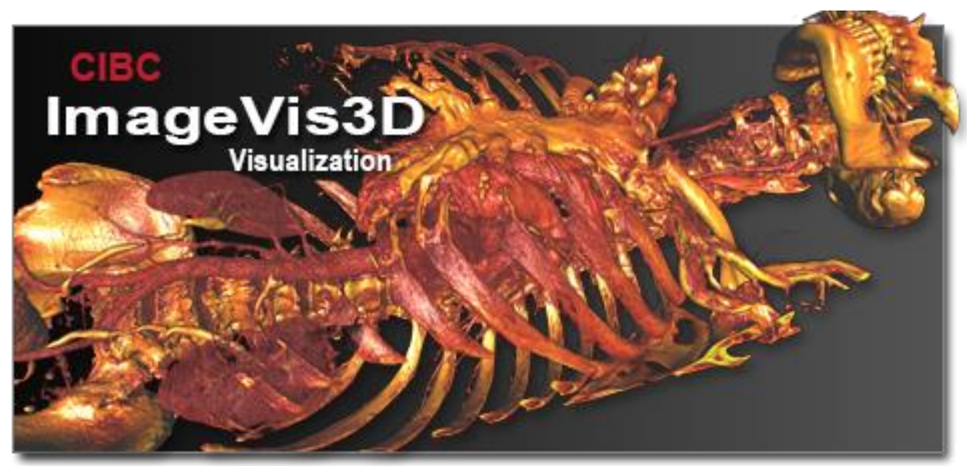
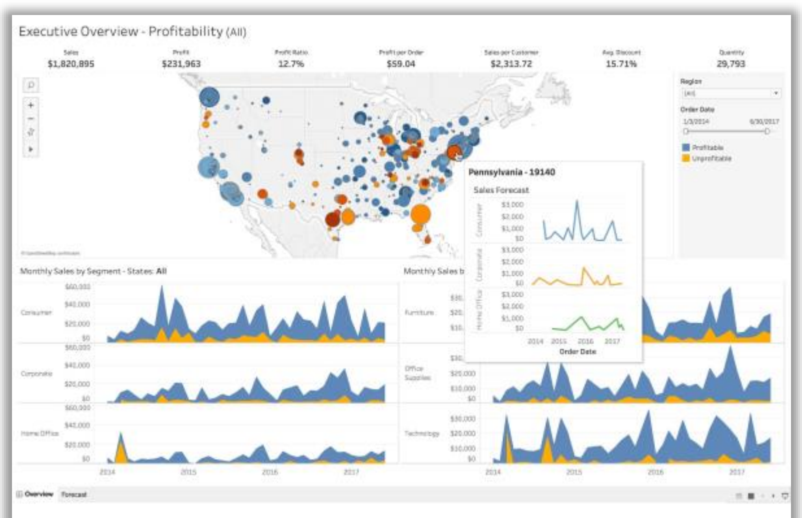


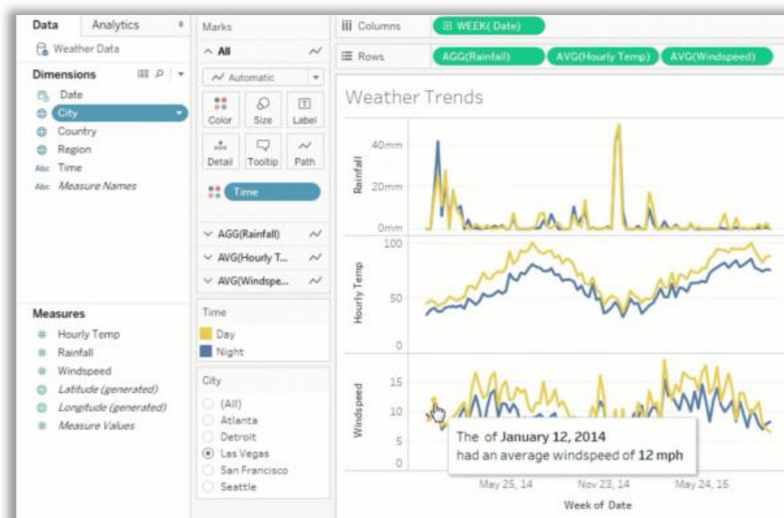
Tableau Desktop

www.tableau.com

SIEMENS
Ingenuity for life

- History
  - Roots at Stanford University (Polaris System)
  - Founded in 2003
- Commercial visual analytics platform for business intelligence
- Explore and understand data using visual dashboards





This course – exams

- Written exam:
  currently planned on-campus,
  no materials allowed
  date and time to be announced

  Registration via tumonline (mandatory)

- If you fail …
  - Oral or written repetition in the semester break

This course – literature

Computer graphics

- Foley, Van Dam, Feiner, Hughes:
  Computer Graphics: Principles and Practice,
  Addison-Wesley, 3rd edition
- Watt, Watt:
  Computer Graphics, Addison-Wesley
- Glassner:
  Principles of digital image synthesis, Morgan Kaufman
- Encarnaçao, Klein, Strasser:
  Graphische Datenverarbeitung, Oldenburg Verlag, 4. Auflage
- Griebel, Bungartz, Zenger:
  Computer Graphik, Vieweg Verlag

This course – literature

Books on Visualization

- T. Munzner: *Visualization Analysis & Design*, CRC Press
- C. Hansen, C. Johnson (Ed.): *The Visualization Handbook*, Elsevier
- H. Schumann, W. Müller: *Visualisierung - Grundlagen und allgemeine Methoden*, Springer-Verlag
- G.M. Nielson, H. Hagen, H. Müller: *Scientific Visualization*, IEEE CS Press
- R.S. Gallagher (Ed.): *Computer Visualization: Graphics Techniques for Scientific and Engineering Analysis*, CRC Press
- R.A. Earnshaw, N. Wiseman (Eds.): *An Introductory Guide to Scientific Visualization*, Springer-Verlag
- K.W. Brodlie et al. (Eds.): *Scientific Visualization - Techniques and Applications*, Springer-Verlag
- E. Tufte: *The visual display of quantitative information*, Graphics Press

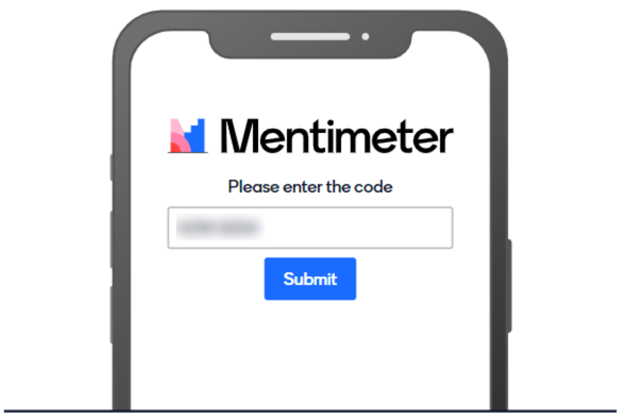
Check the library and the web for literature!

What is your field of study?

SIEMENS
Ingenuity for life

Go to

www.menti.com



Enter the code

54 11 67 0

Or use QR code

This course – lecturer

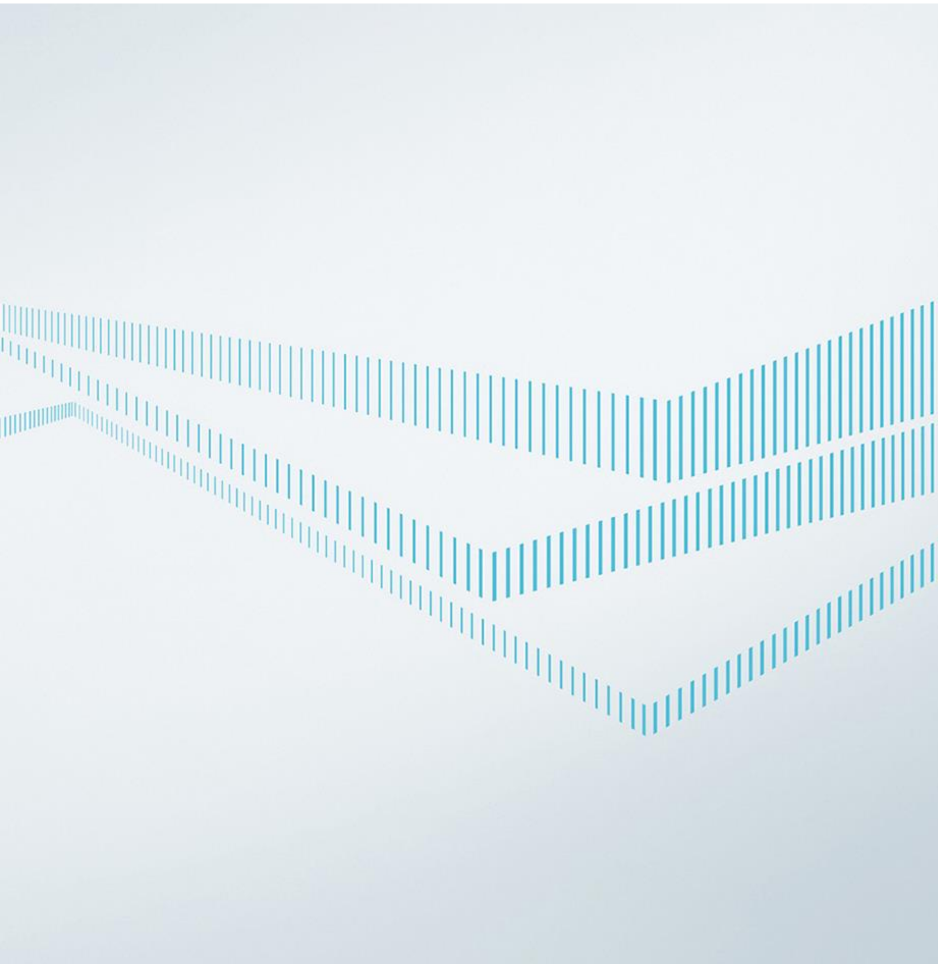
SIEMENS
Ingenuity for life

- Dr. Johannes Kehrer
  - Siemens Technology
  - **`kehrer.johannes@siemens.com`**
  - Meeting by appointment

- Diploma theses and project work
  - Many topics are available
  - Check websites
    wwwcg.in.tum.de
  - Come and talk to us!

Contact information

Siemens
Ingenuity for life

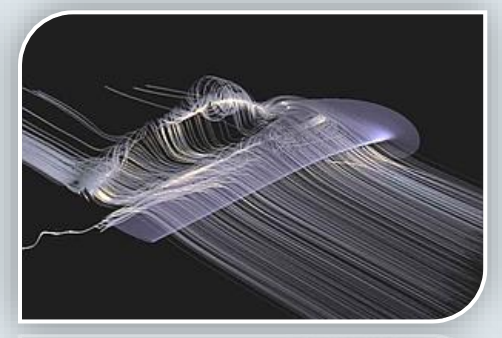
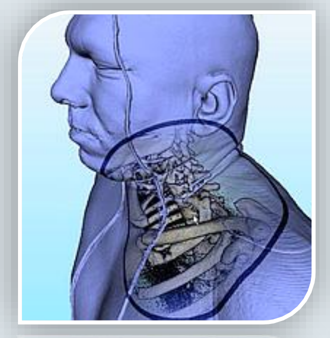


Dr. Johannes Kehrer

Siemens Technology
T DAI HCA-DE
Otto-Hahn-Ring 6
81739 München, Deutschland

E-mail:
kehrer.johannes@siemens.com

Internet
siemens.com/innovation


Visual Data Analytics
Visualization

Dr. Johannes Kehrer – Siemens Technology, Munich

Disclaimer

These lecture slides and materials are for personal use by students only and are intended for educational purposes only. Distribution or reproduction of the material for any other purposes is prohibited.

The views and opinions expressed by any individual during the course of these lecturers are their own and do not necessarily represent the views, opinions, or positions of Siemens or the Technical University of Munich.

Although reasonable efforts have been made to ensure the information contained in these lecture slides is current and accurate, we do not assume any responsibility for the content of the posted material.

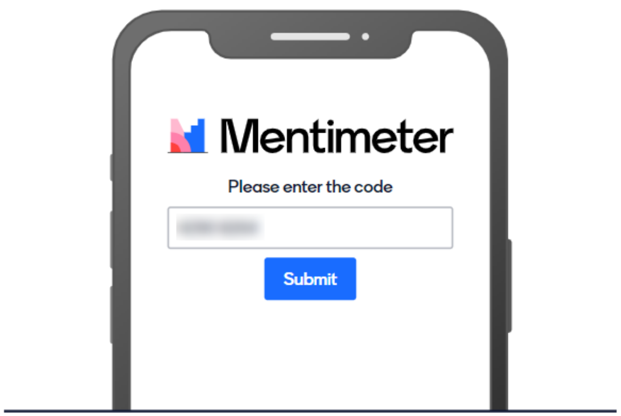
These slides also contain links to websites of third parties. As the content of these websites is not under our control, we cannot assume any liability for such external content.

Any comments/questions?

SIEMENS
Ingenuity for life

Go to

www.menti.com



Enter the code

54 11 67 0

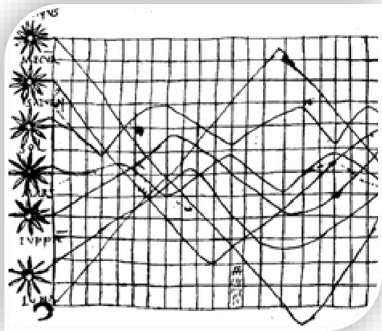
Or use QR code

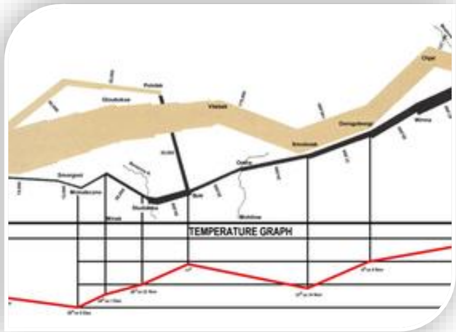
Today’s lecture

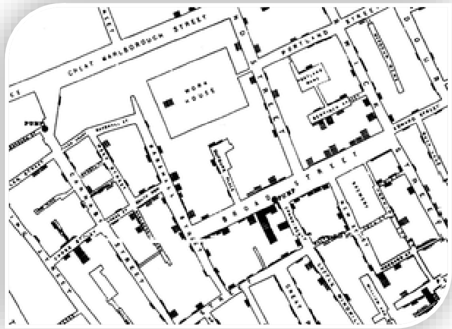
- History of visualization
- Definitions, goals and major areas

History

Techniques for finding visual representations of (abstract) data are not new!

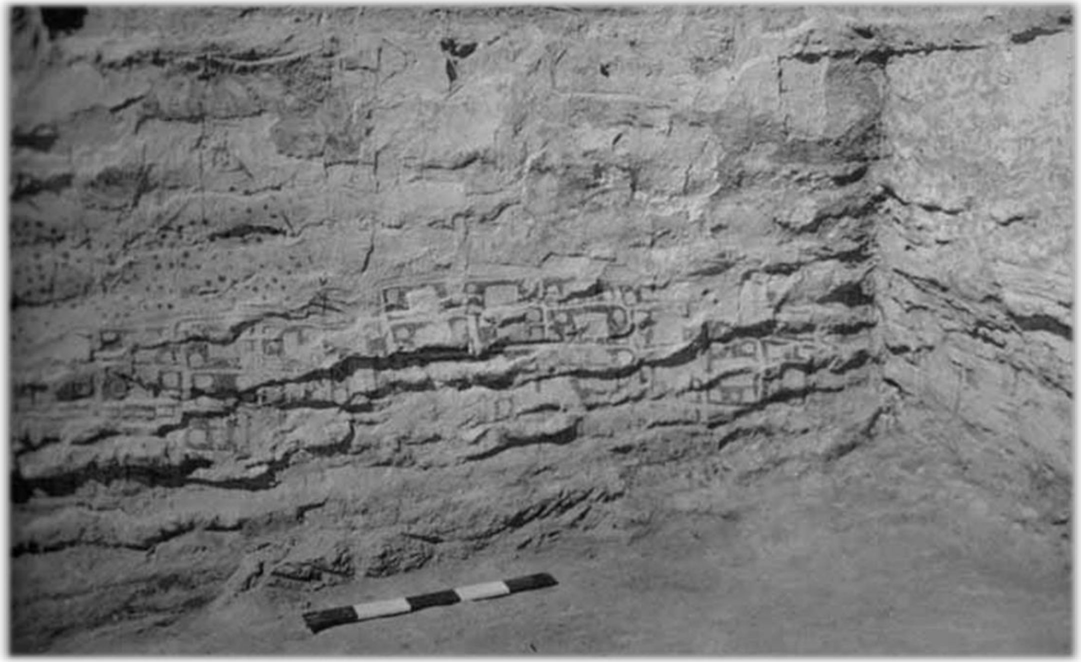






History

SIEMENS
Ingenuity for life

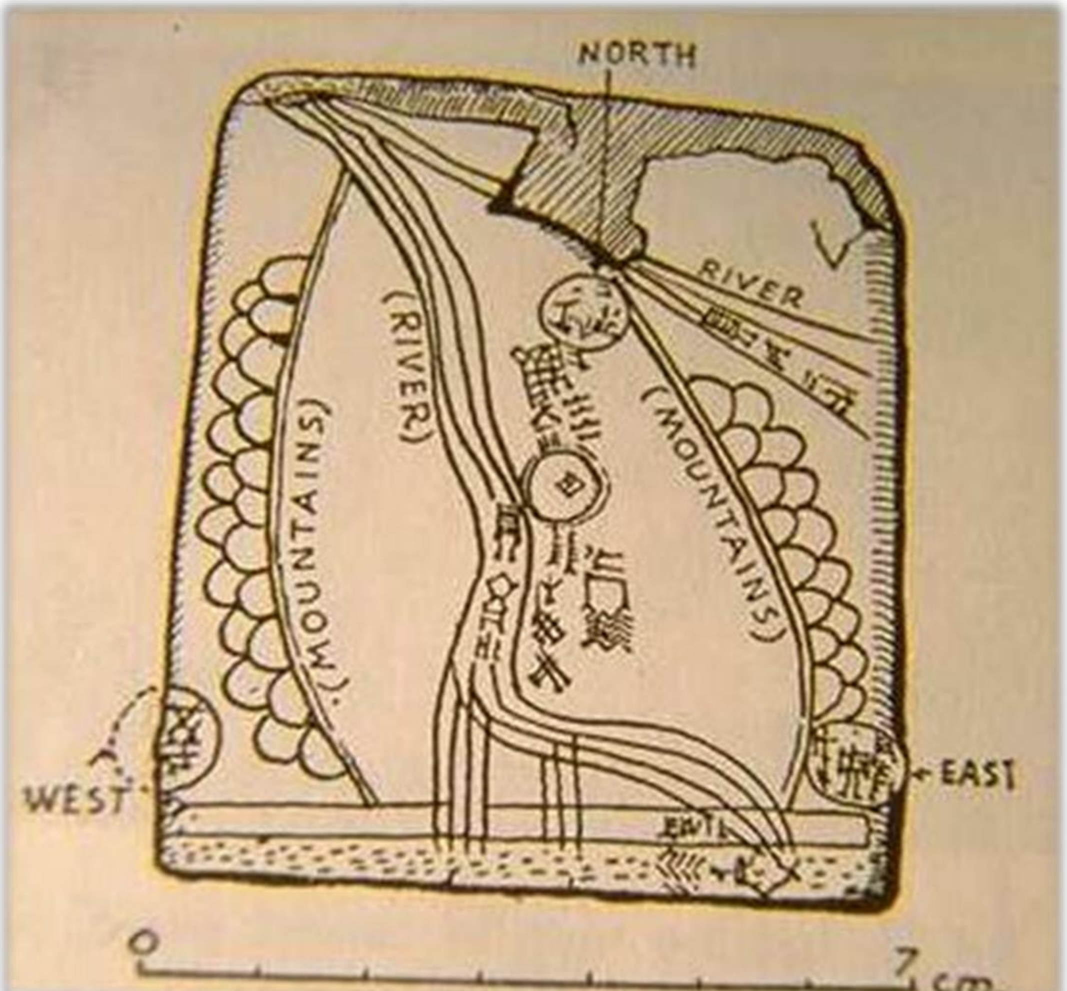


(Oldest) Map of Catal Hyük, Turkey 6200 BC

Excavation

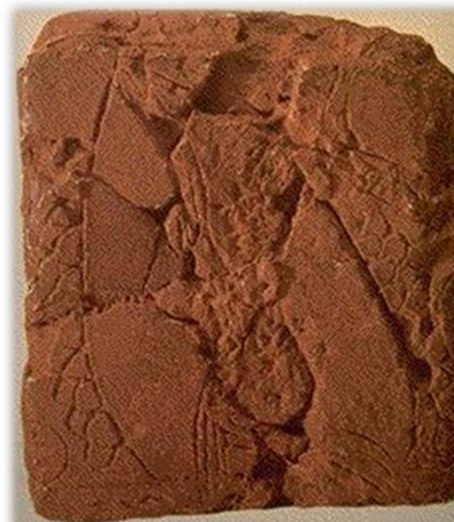
Reconstruction

History

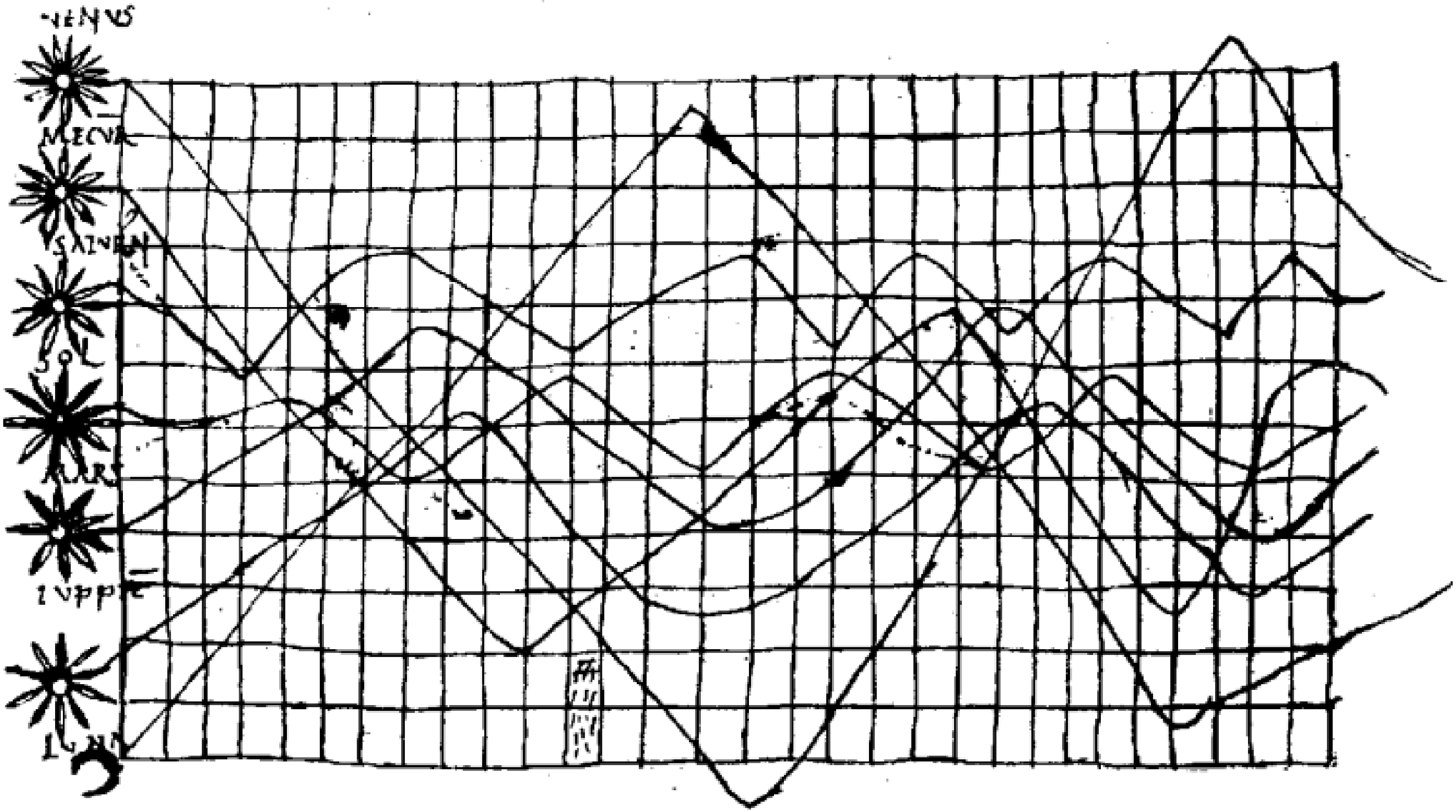


Map on clay of Ga-Sur, Iraq
2500 BC

Map on clay
Reconstruction



History

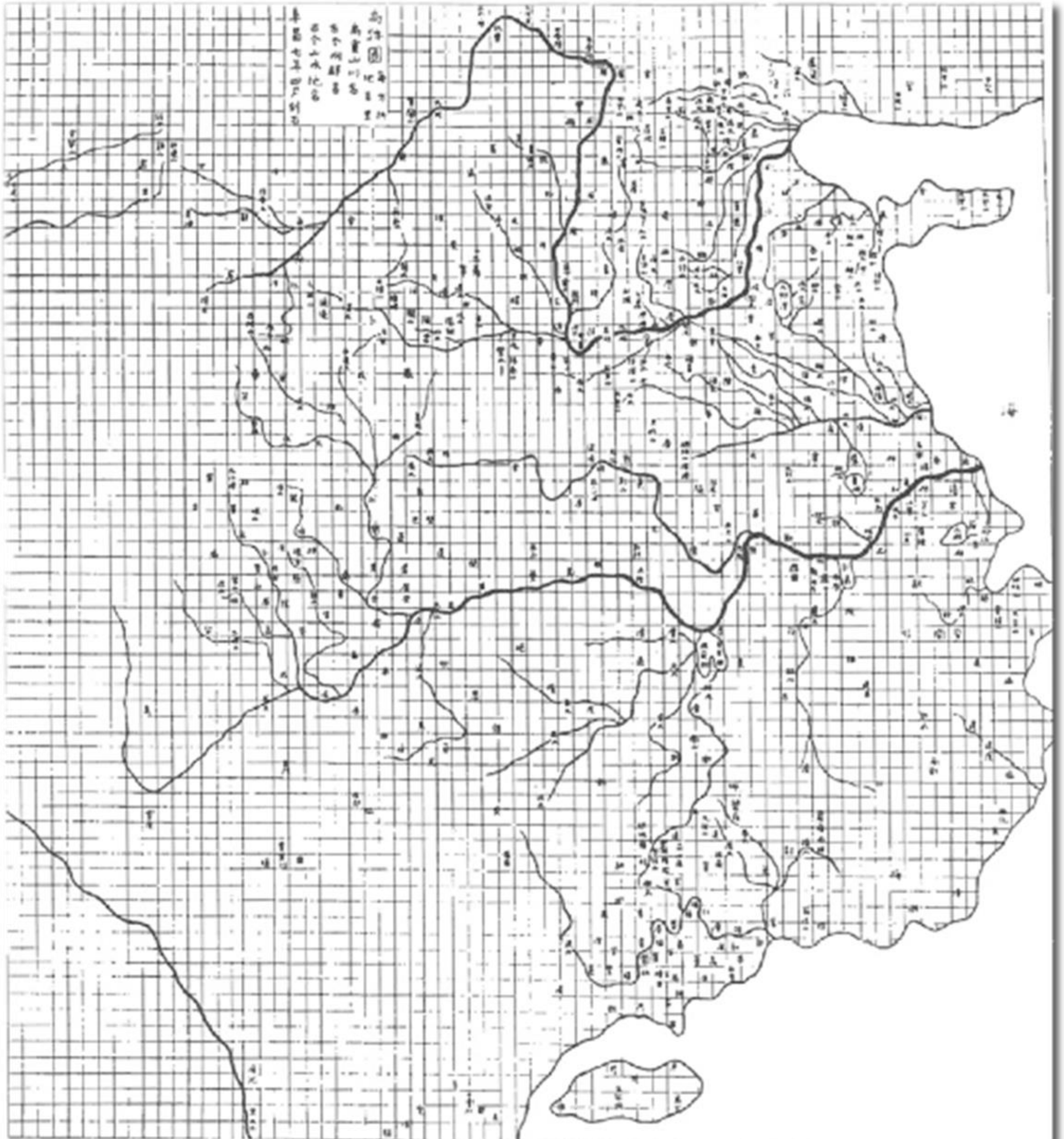


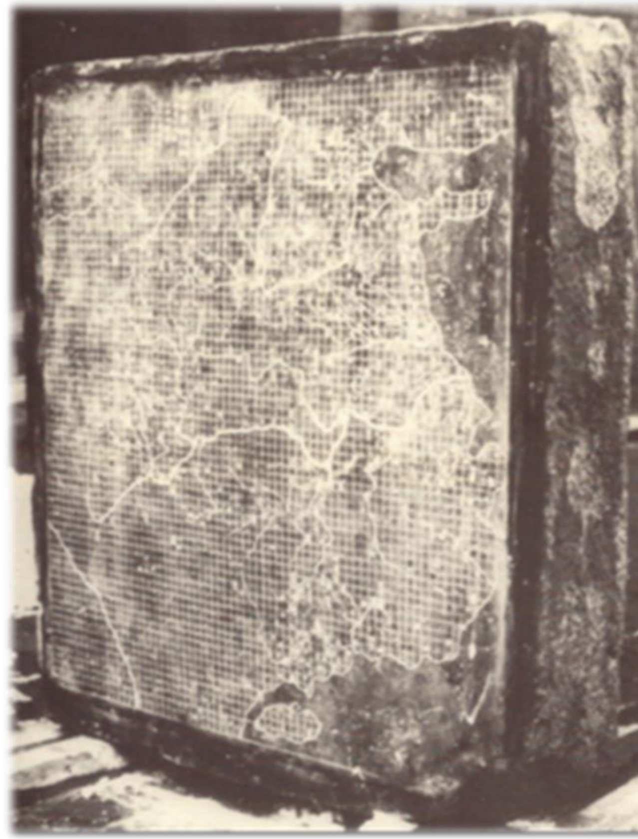
Inclinations of planetary orbits as function of time

10th century

History

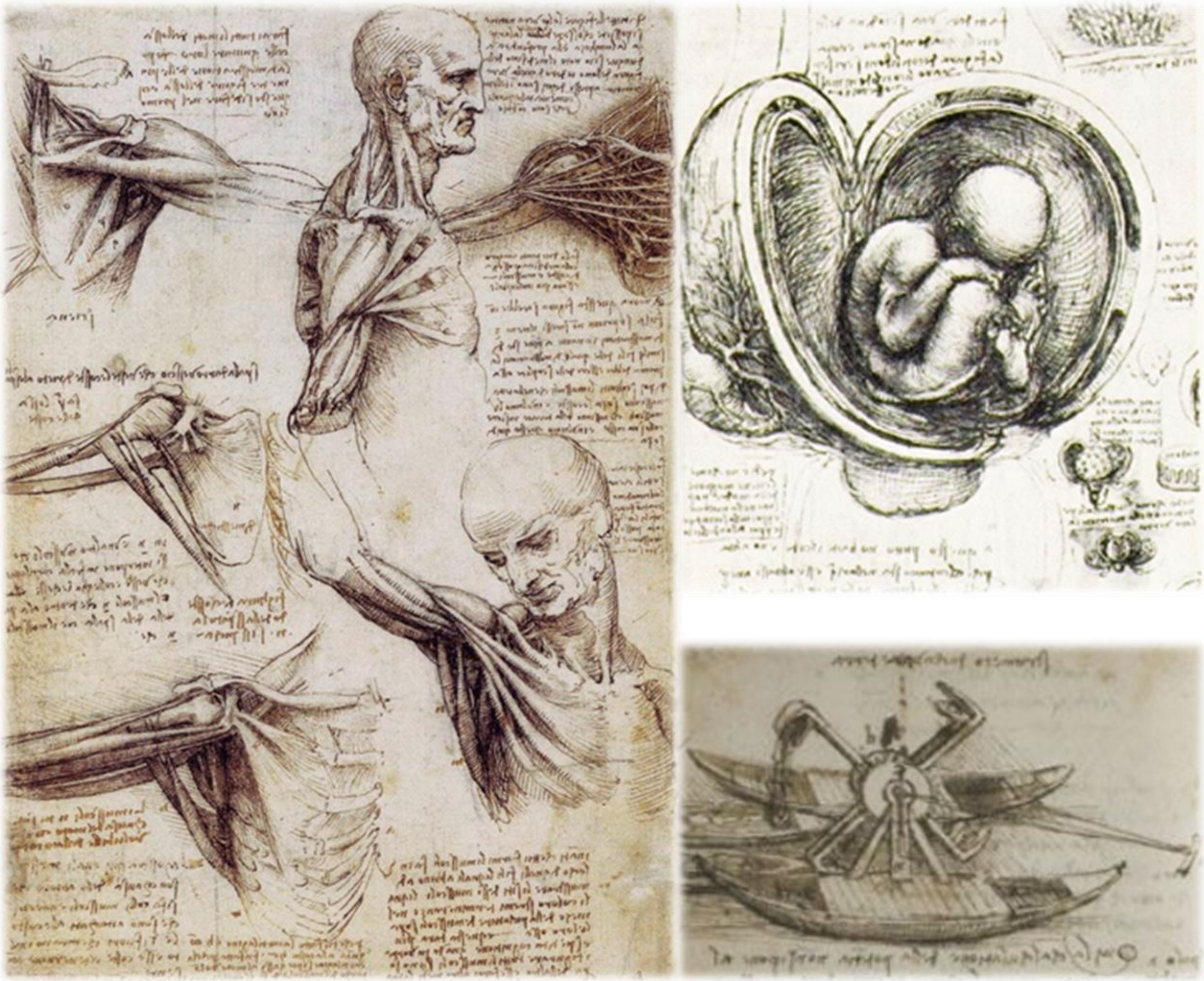
SIEMENS
Ingenuity for life





Map of the Tracks of
Yü the Great, China (11th century)

History

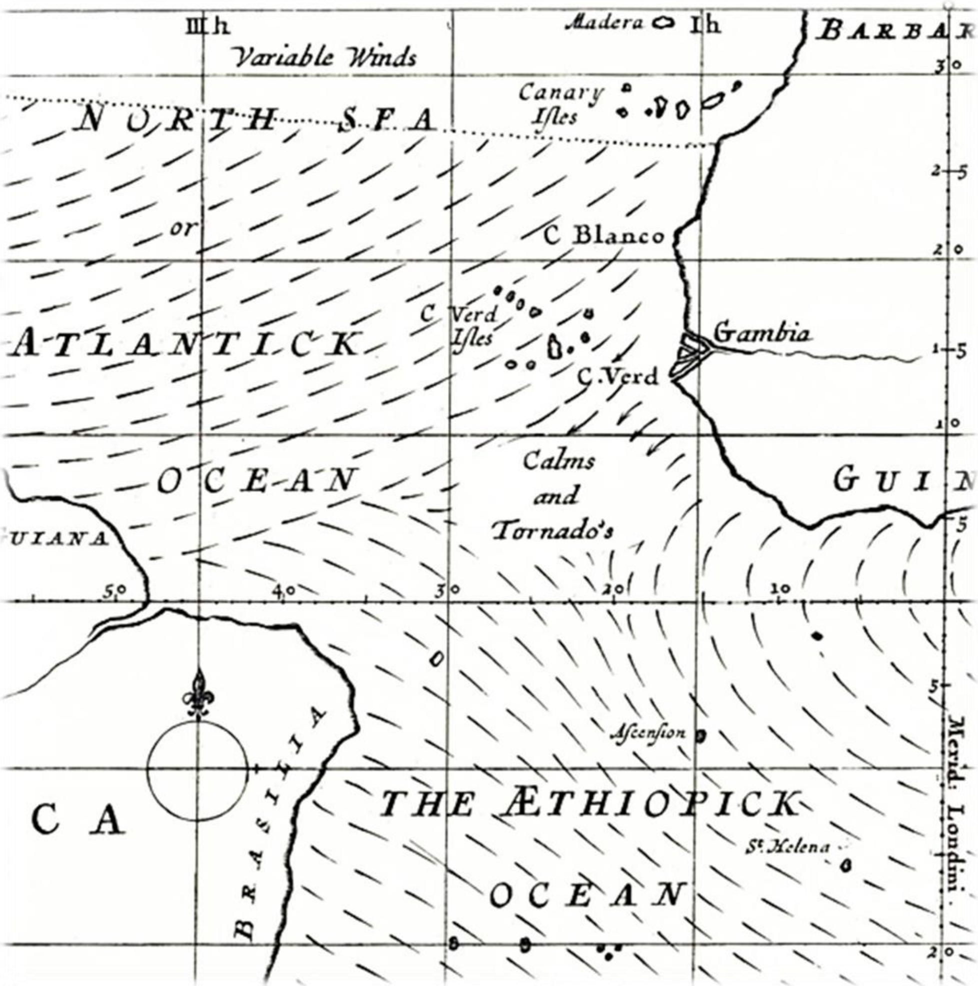


Illustrations by Leonardo DaVinci (1452-1519)

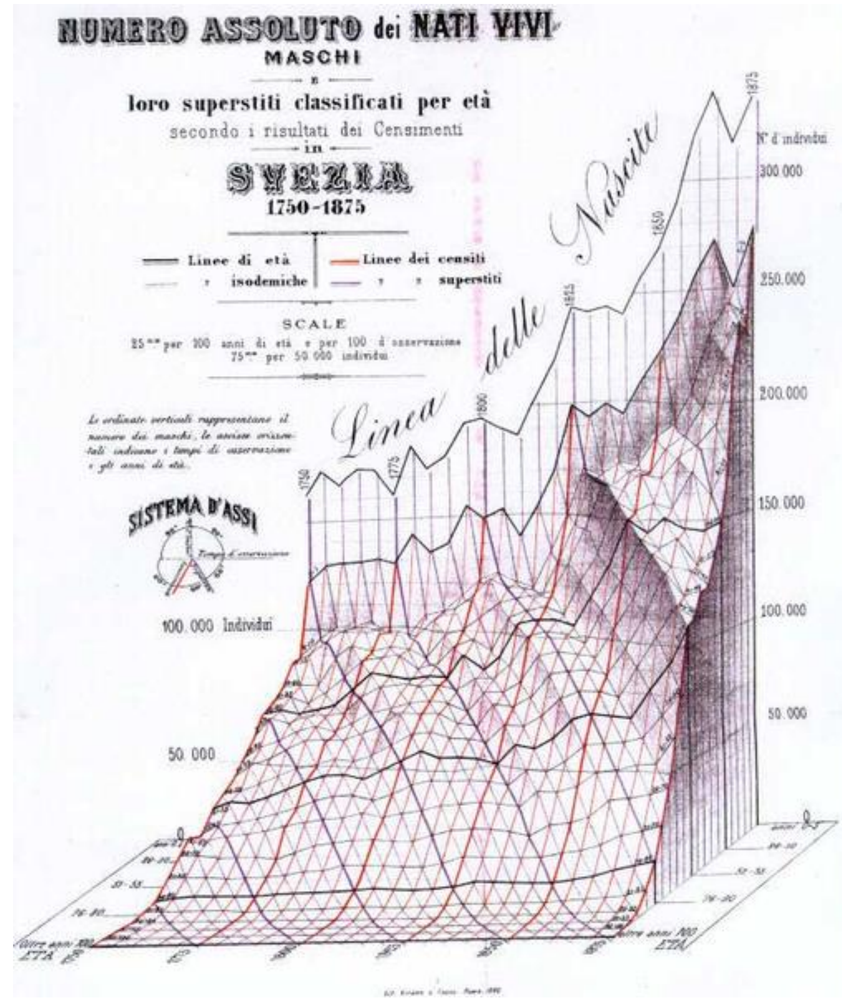
William Curtis (1746-1799)

History

SIEMENS
Ingenuity for life

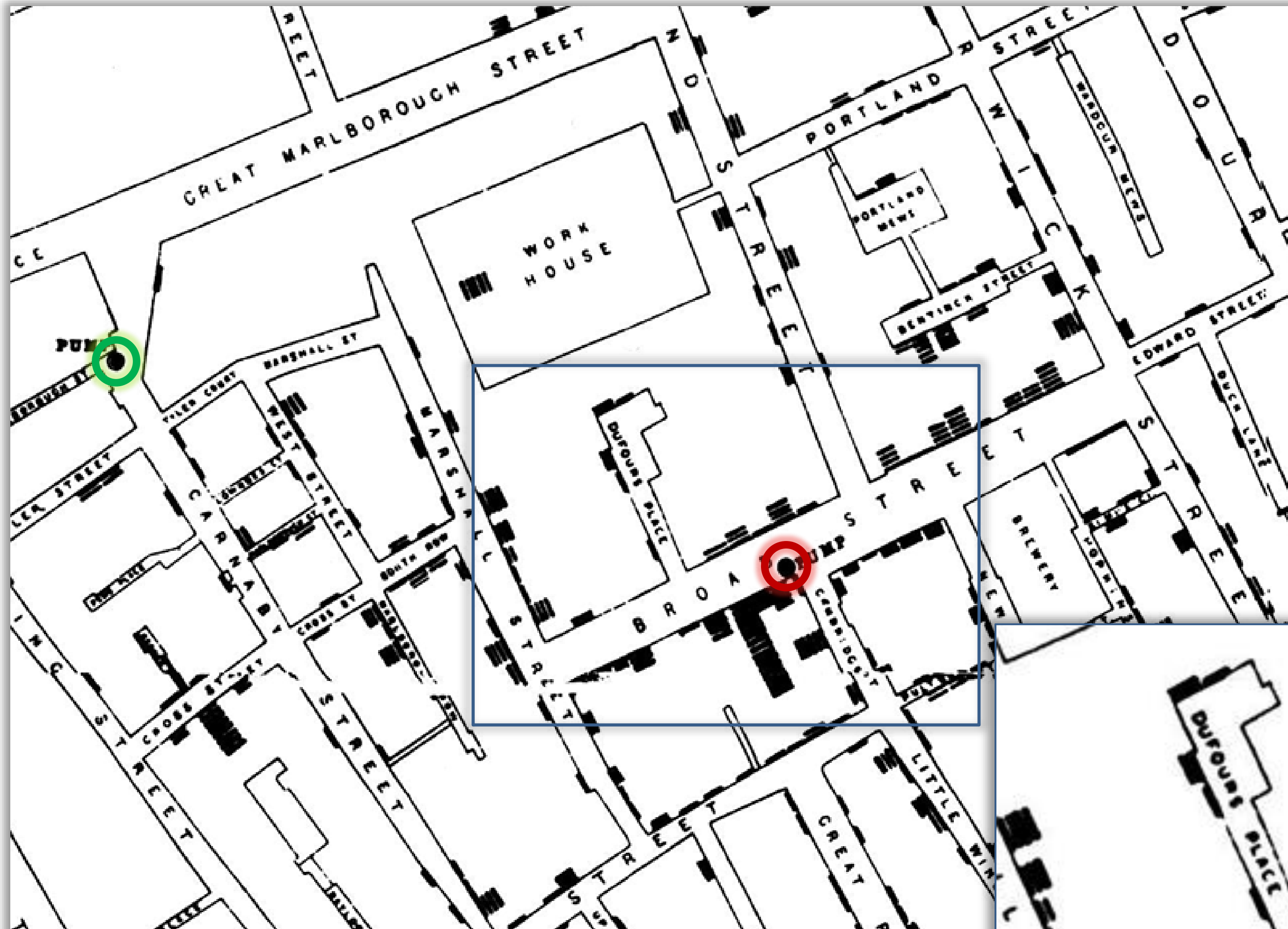


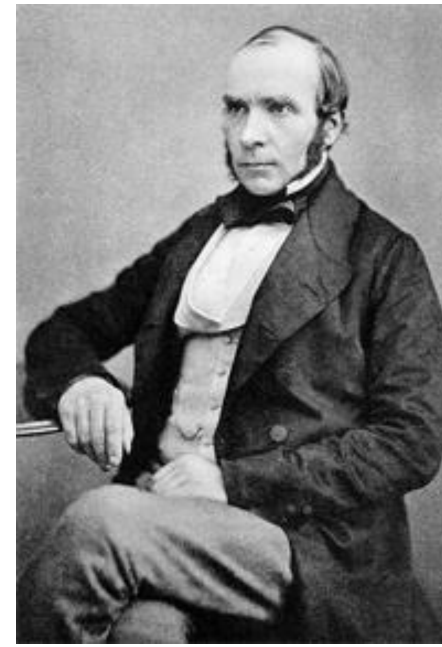
Vector visualization (1686)



Height field (1879)

History



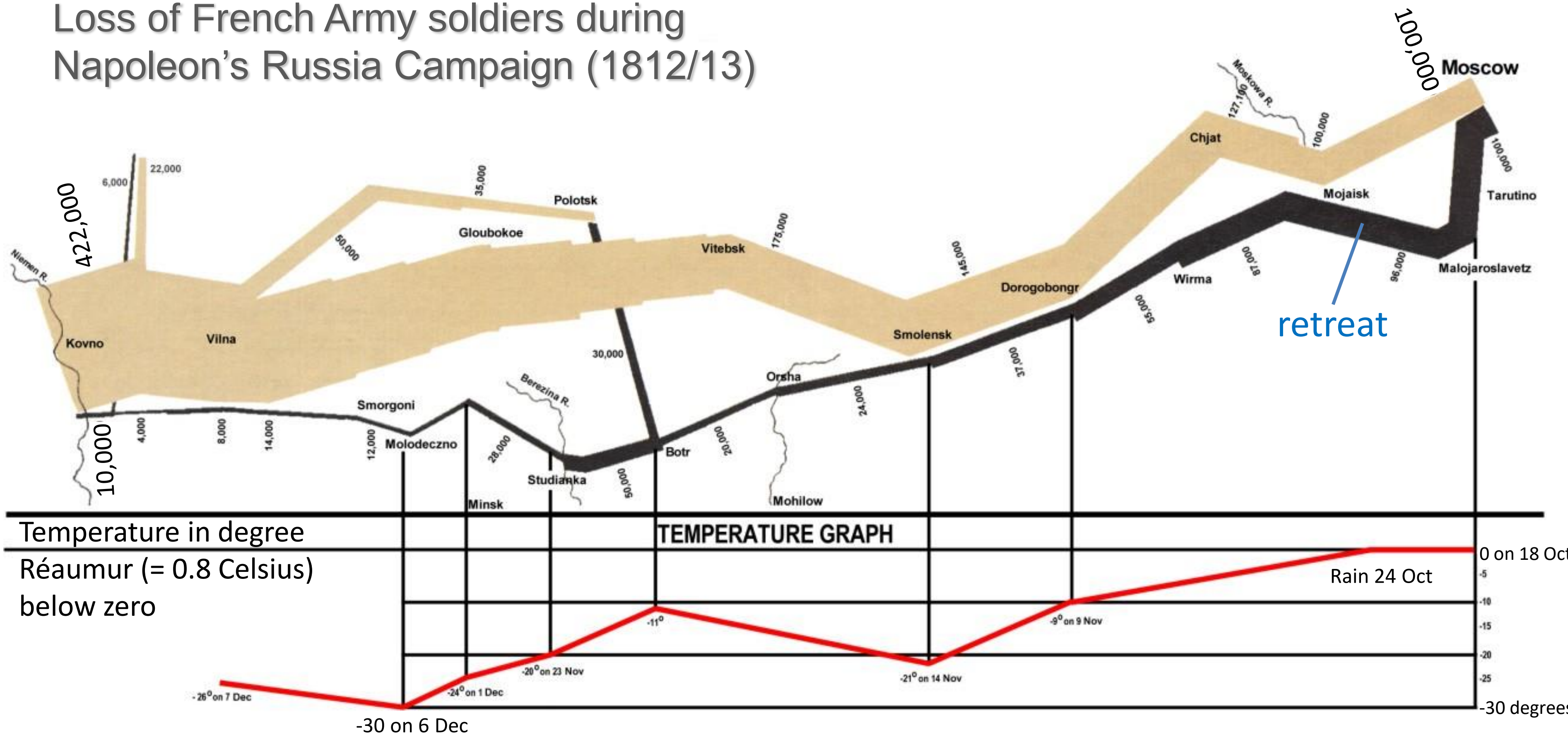


Cholera outbreak in Soho, London
John Snow (1854)

History

SIEMENS
Ingenuity for life

Loss of French Army soldiers during Napoleon's Russia Campaign (1812/13)



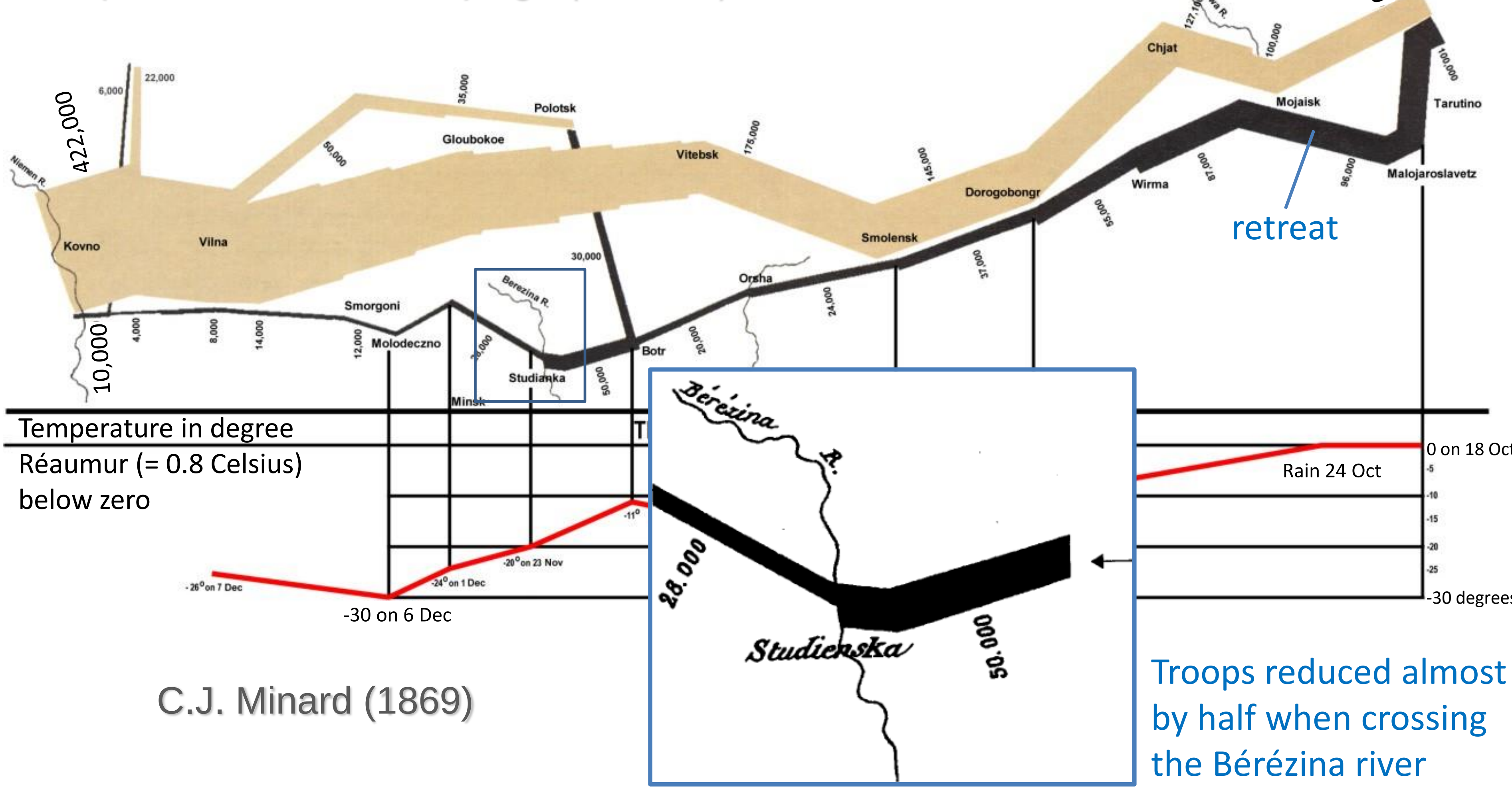
C.J. Minard (1869)

Map encodes size of army (width of band), location & direction of movement, split and reunion of troops, and temperature during retreat

History

SIEMENS
Ingenuity for life

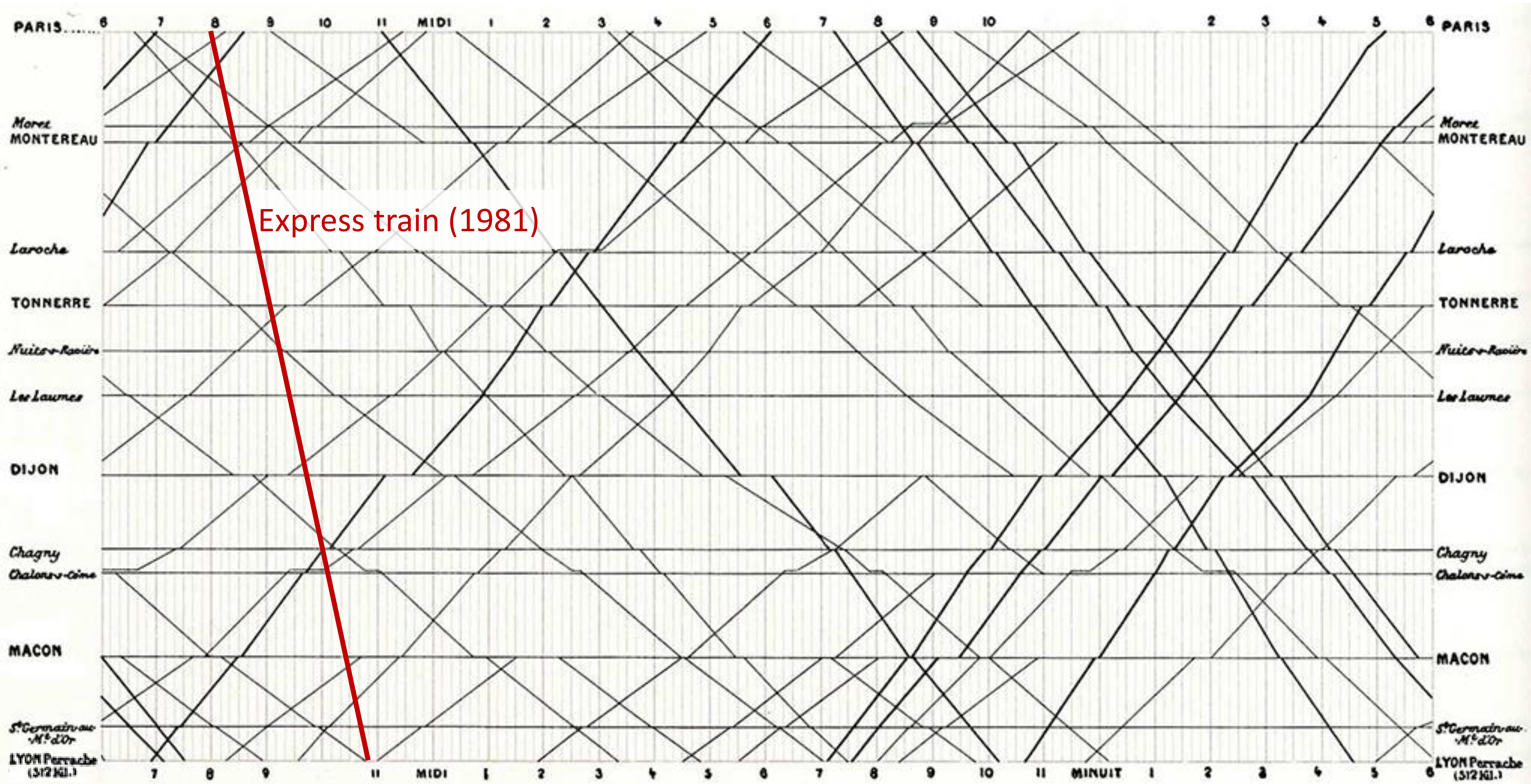
Loss of French Army soldiers during Napoleon's Russia Campaign (1812/13)



C.J. Minard (1869)

Troops reduced almost by half when crossing the Bérézina river

History



Train schedule between Paris and Lyon
E.J. Marey (1880)

History of Modern Visualization

“The purpose of computing is **insight**, not numbers”
(Hamming 1962)

- 1987 US NSF Advisory Panel on Graphics and Image Processing
  - Computer experiments allow access to new worlds
    - Real experiments are too expensive, too dangerous, etc.
    - Arbitrary large, small time scales, and spatial dimensions

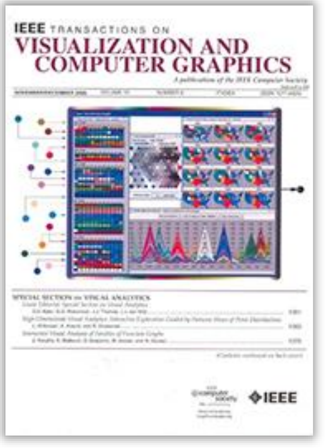
History of Modern Visualization

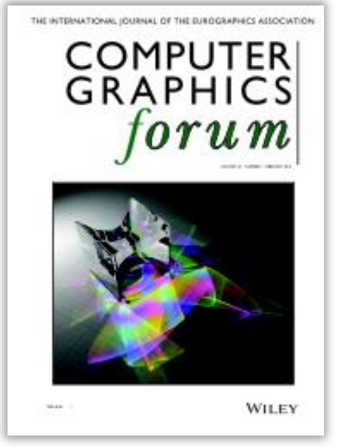
SIEMENS
Ingenuity for life

- Direct implications
  - The data flood from supercomputer simulations can only be dealt with visually
  - Needs a visualization specialist and an interdisciplinary team
  - New developments in hard/software, nets, etc. are necessary
- Advantages in the long term will be
  - Faster insight
  - Faster product – development cycles
  - Stronger position in global competition
- Suggestion: spend lots of money to support scientific visualization

History of Modern Visualization

- First visualization conference: 1990
- Conferences:
  - IEEE SciVis,
    IEEE InfoVis, IEEE VAST
  - IEEE PacificVis, EuroVis, etc.
- Journals
  - IEEE Transactions on Visualization and Computer Graphics
  - Computer Graphics Forum
  - Computers & Graphics, etc.

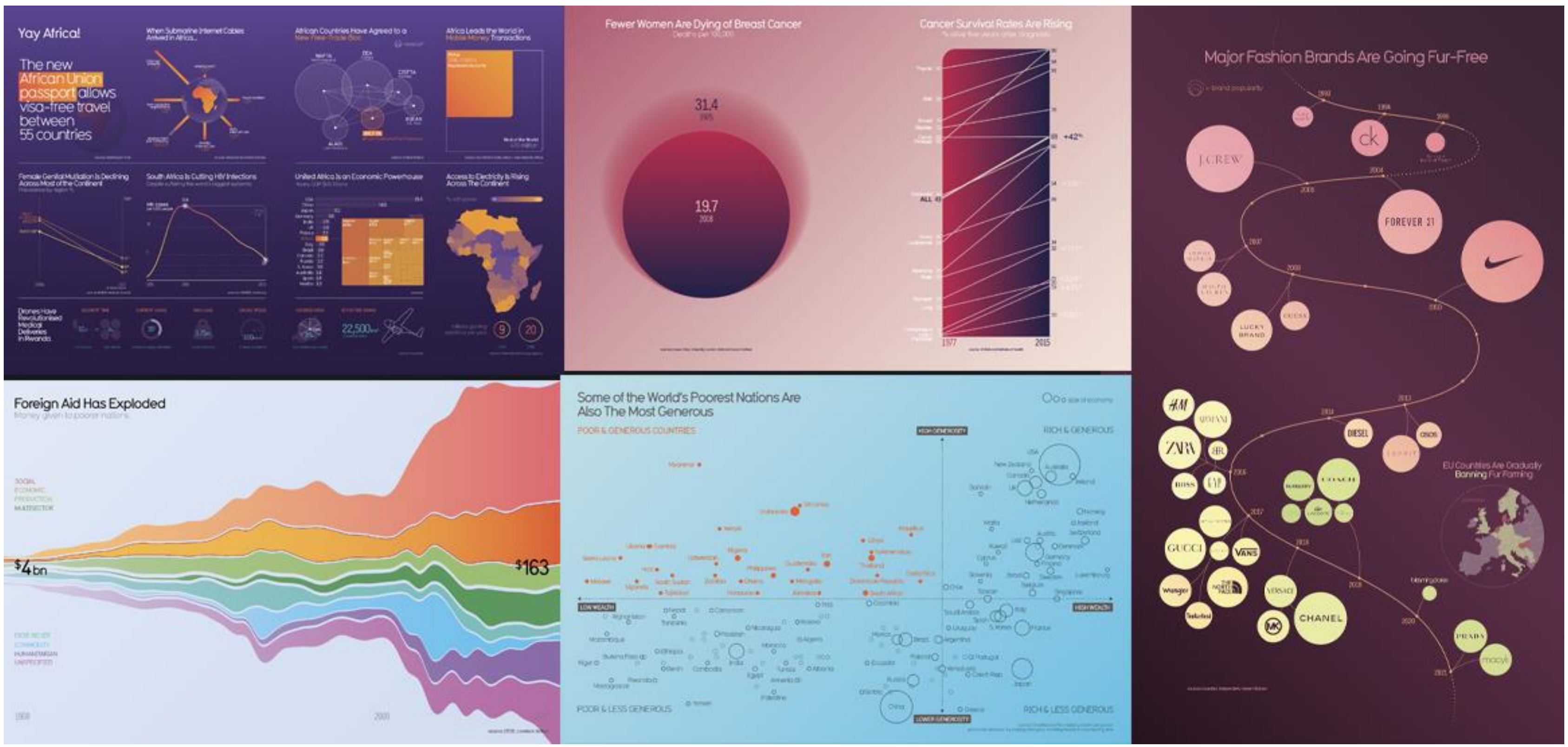




History of Modern Visualization

SIEMENS
Ingenuity for life

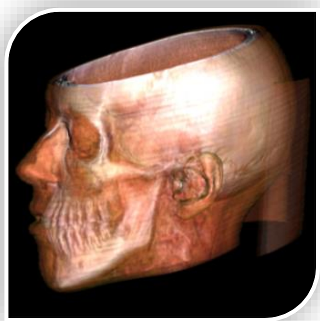
- Infographics
- Data journalism (NY Times, Die Zeit, etc.)
- Web-based visualizations (D3.js, Vega, ChartJS, etc.)

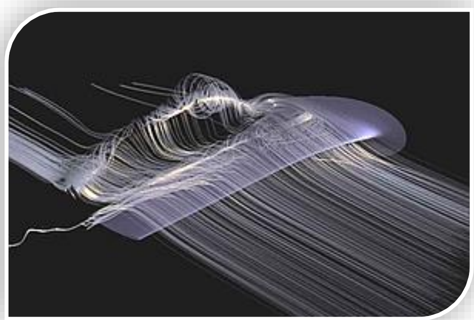


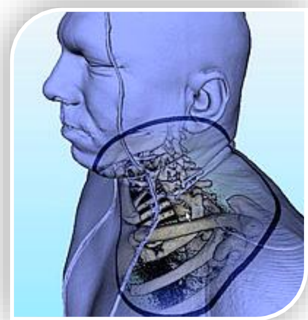
informationisbeautiful.net

Visualization

Definitions, goals, and major areas



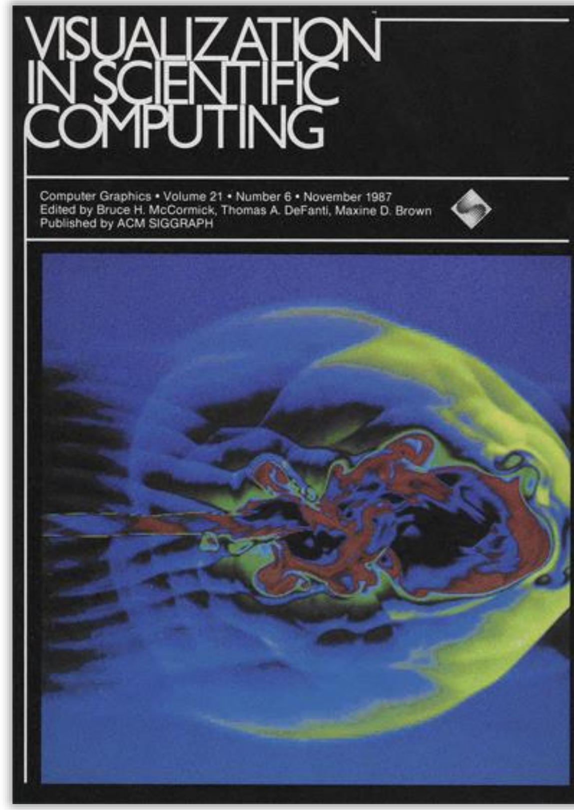




Visualization – Definitions

- McCormick, DeFanti, Brown 1987:

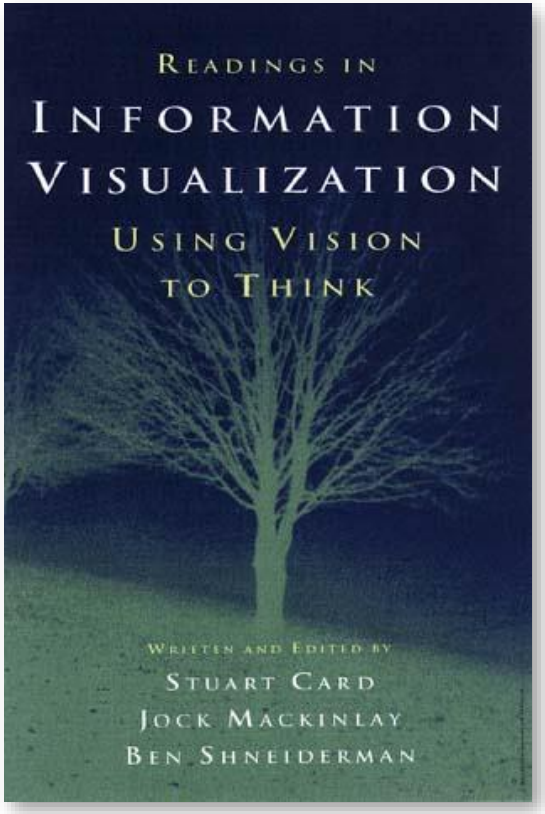
"Visualization is a method of computing. It **transforms the symbolic into the geometric**, enabling researchers to observe their simulations and computations. Visualization offers a method for **seeing the unseen**. It enriches the process of scientific discovery and fosters profound and unexpected **insights**.
[…] It studies those mechanisms in **humans and computers** which allow them in concert to perceive, use and communicate visual information."



Visualization – Definitions

- Card, MacKinlay, Shneiderman 1999:

"[Information / Scientific] Visualization ...
The use of **computer-supported, interactive, visual representations** of [abstract/scientific] **data** to **amplify cognition**."



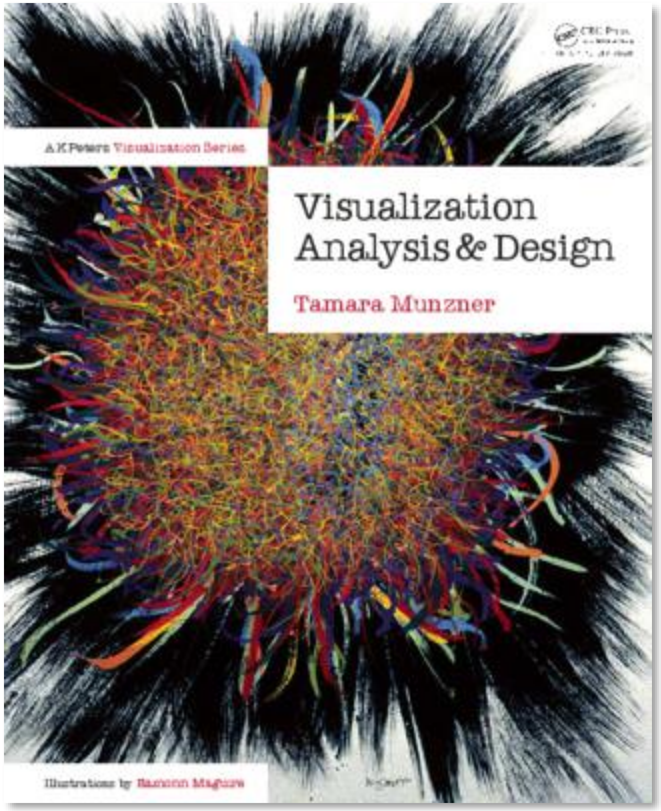
"The purpose of visualization is **insight**, not pictures." [B. Shneiderman]

Visualization – Definitions

SIEMENS
Ingenuity for life

- Munzner 2014:

  “Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively”

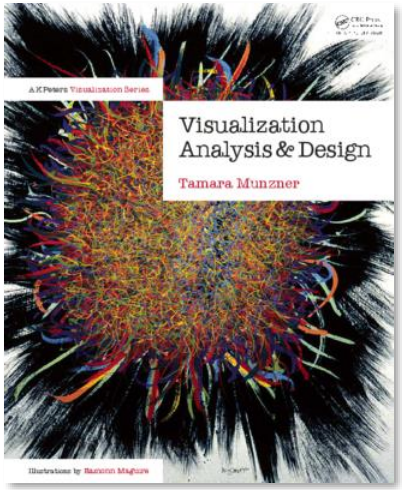


Visualization – Definitions

SIEMENS
Ingenuity for life

- Munzner 2014:

  "Computer-based visualization systems provide visual representations of **datasets** designed to help **people** carry out tasks more effectively"

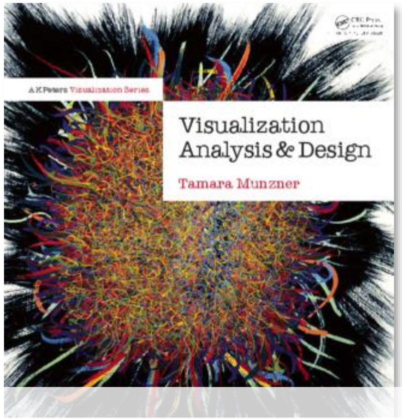


- Why have a human in the loop?
  - No need for vis when fully automatic solution exists that can be trusted, e.g.,
    - if question can be answered by a compact, precise query
    - if a decision can be automated (e.g., stock market trading)

Visualization – Definitions

- Munzner 2014:

I		II		III		IV	
x	y	x	y	x	y	x	y
10.0	8.04	10.0	9.14	10.0	7.46	8.0	6.58
8.0	6.95	8.0	8.14	8.0	6.77	8.0	5.76
13.0	7.58	13.0	8.74	13.0	12.74	8.0	7.71
9.0	8.81	9.0	8.77	9.0	7.11	8.0	8.84
11.0	8.33	11.0	9.26	11.0	7.81	8.0	8.47
14.0	9.96	14.0	8.10	14.0	8.84	8.0	7.04
6.0	7.24	6.0	6.13	6.0	6.08	8.0	5.25
4.0	4.26	4.0	3.10	4.0	5.39	19.0	12.50
12.0	10.84	12.0	9.13	12.0	8.15	8.0	5.56
7.0	4.82	7.0	7.26	7.0	6.42	8.0	7.91
5.0	5.68	5.0	4.74	5.0	5.73	8.0	6.89



Identical statistics	
x mean	9.0
y mean	7.5
x variance	10.0
y variance	3.75
x/y correlation	0.82
regression line	y = 3+0.5x

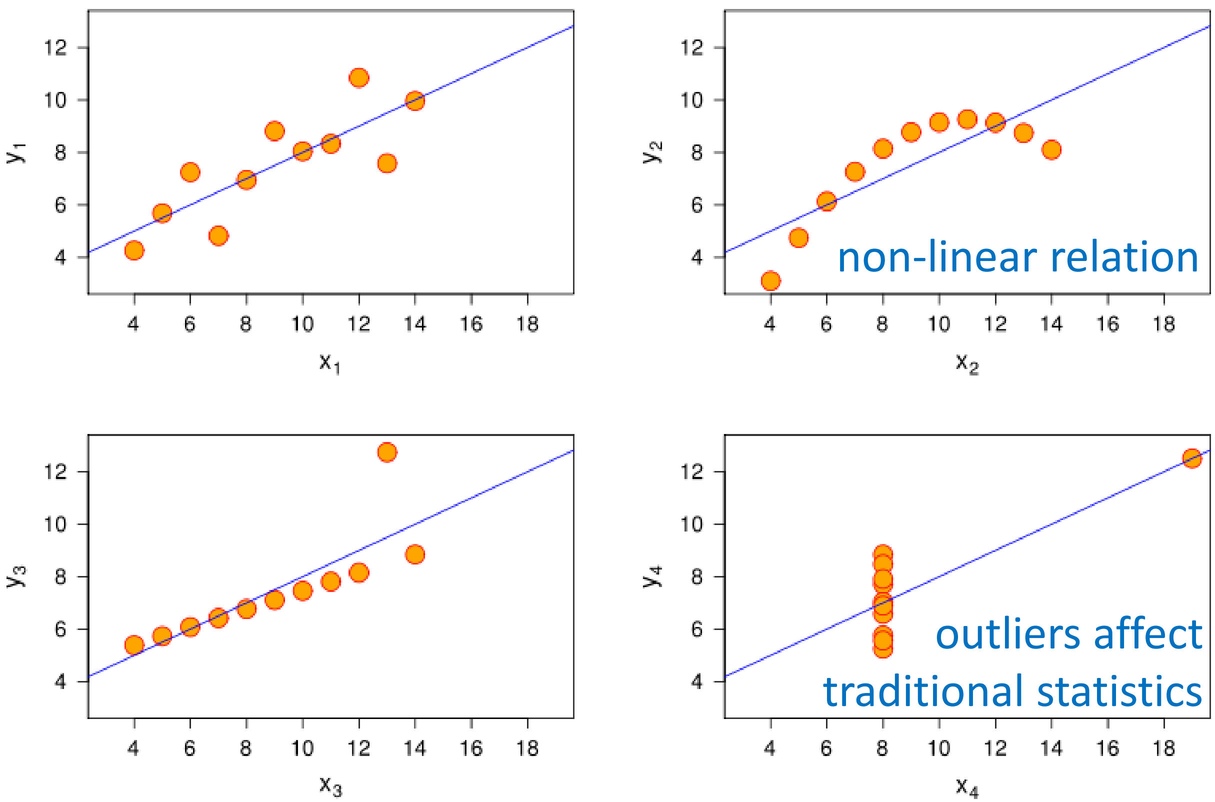
4 datasets with identical statistics

[Anscombe, 1973]

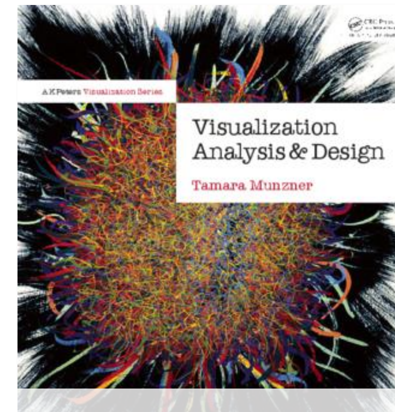
Visualization – Definitions

- Munzner 2014:

Statistics don't always tell you the full story!



4 datasets with identical statistics



Identical statistics	
x mean	9.0
y mean	7.5
x variance	10.0
y variance	3.75
x/y correlation	0.82
regression line	y = 3+0.5x

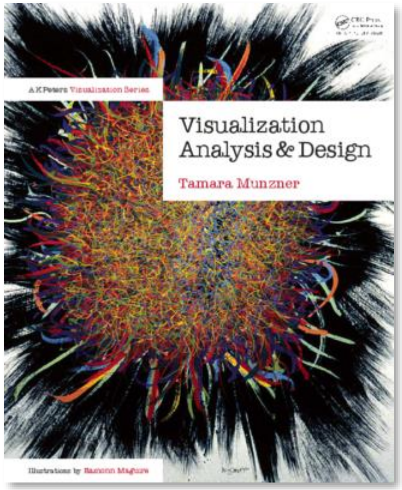
[Anscombe, 1973]

Visualization – Definitions

SIEMENS
Ingenuity for life

- Munzner 2014:

  “Computer-based visualization systems provide visual representations of **datasets** designed to help **people** carry out tasks more effectively”



- Why have a human in the loop?
  - Statistics only work for well-specified problems
    - Many analysis problems are ill-defined
    - Don’t know exactly what questions to ask in advance
  - Detect the expected & discover the unexpected [Thomas & Cook]

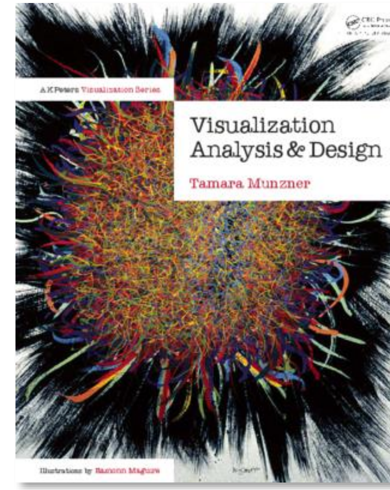
“Visualization is suitable when there is a need to **augment** human capabilities **rather than replace** people with computational decision-making methods.”

Visualization – Definitions

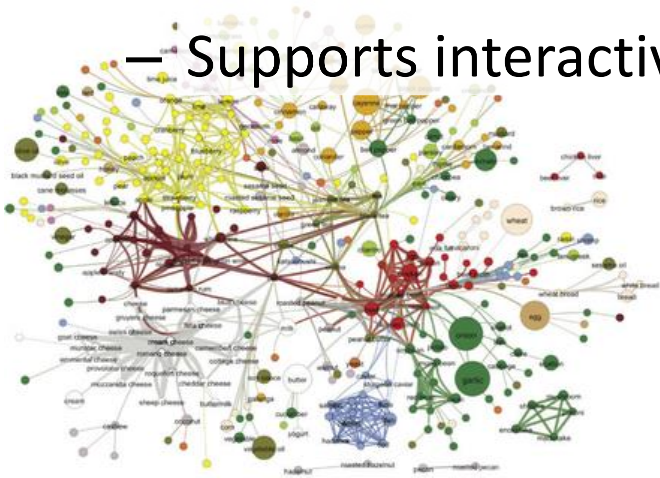
SIEMENS
Ingenuity for life

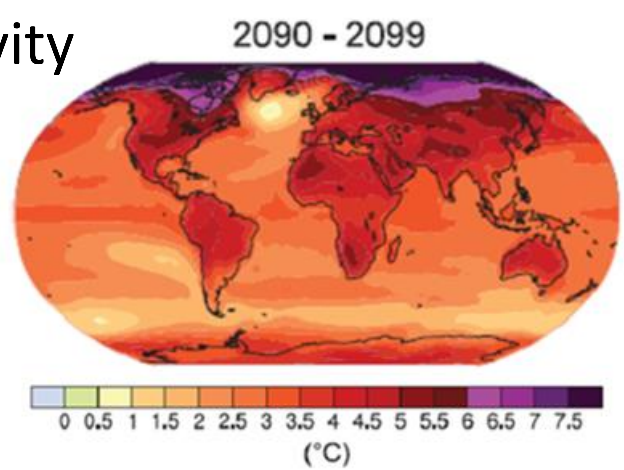
- Munzner 2014:

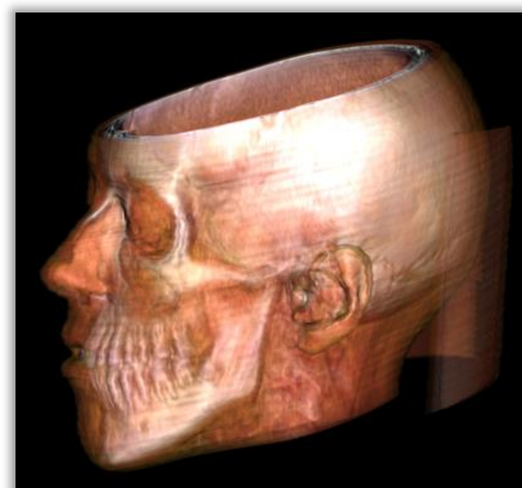
  “**Computer-based** visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively”



- Why have a computer in the loop?
  - Large datasets are infeasible to draw by hand
  - Goes beyond human capacities / patience
  - Supports interactivity








Search Task

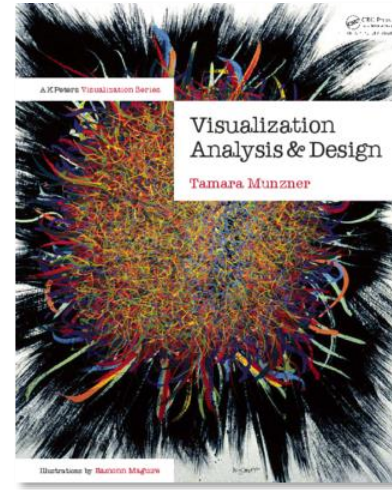
- Spot the differences

Comparison Task

Visualization – Definitions

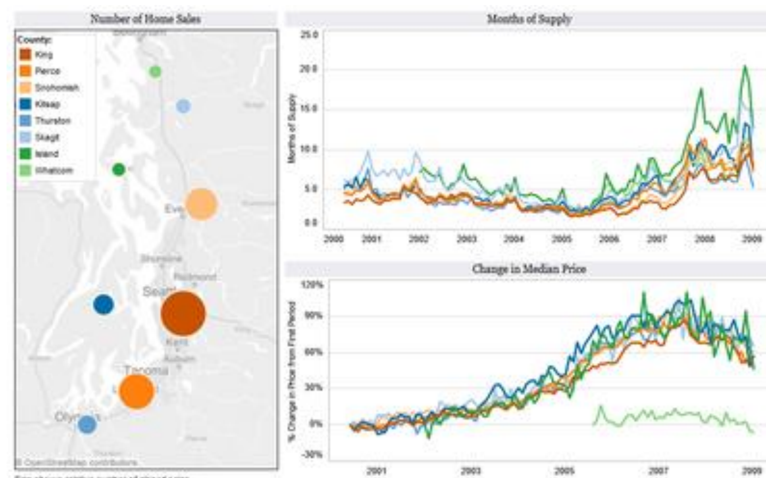
- Munzner 2014:

  “**Computer-based** visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively”



- Why use interactivity?
  - Handle data complexity
  - A single static view can show only one aspect of data

> “Overview first, zoom and filter, then details-on-demand.”
> [B. Shneiderman]



www.tableau.com/stories/gallery/real-estate-prices

Visualization – Definitions

SIEMENS
Ingenuity for life

Interaction

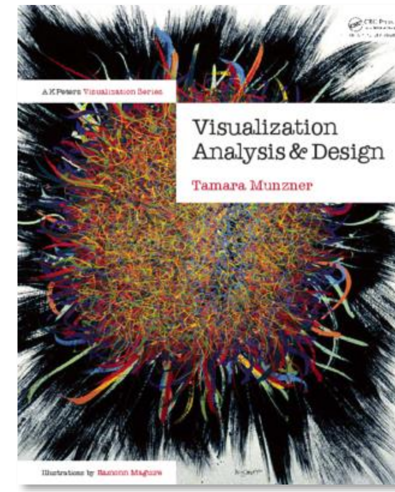
[Bruckner & Gröller 07]

Visualization – Definitions

SIEMENS
Ingenuity for life

- Munzner 2014:

  > “Computer-based visualization systems provide **visual** representations of datasets designed to help people carry out tasks more effectively”

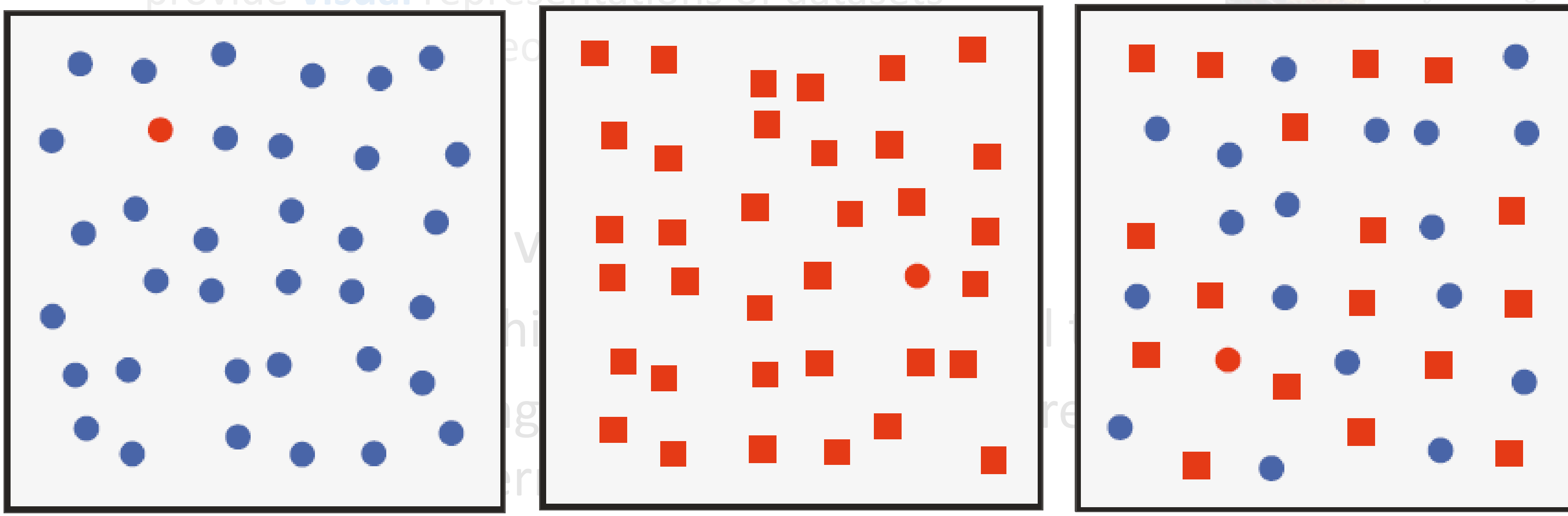


- Why depend on vision?
  - Visual system is high-bandwidth channel to brain
  - Detect interesting visual structures and relationships (e.g., anomalies, patterns, or trends)
  - Sequential vs. parallel processing (popout)

Visualization – Definitions

- Munzner 2014:

“Computer-based visualization systems
provide **visual** representations of datasets



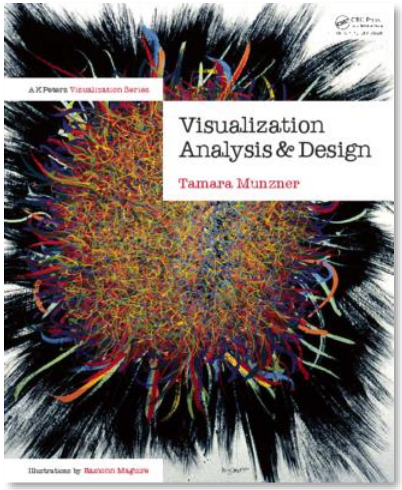
  - Sequential vs. parallel processing (popout)

Visualization – Definitions

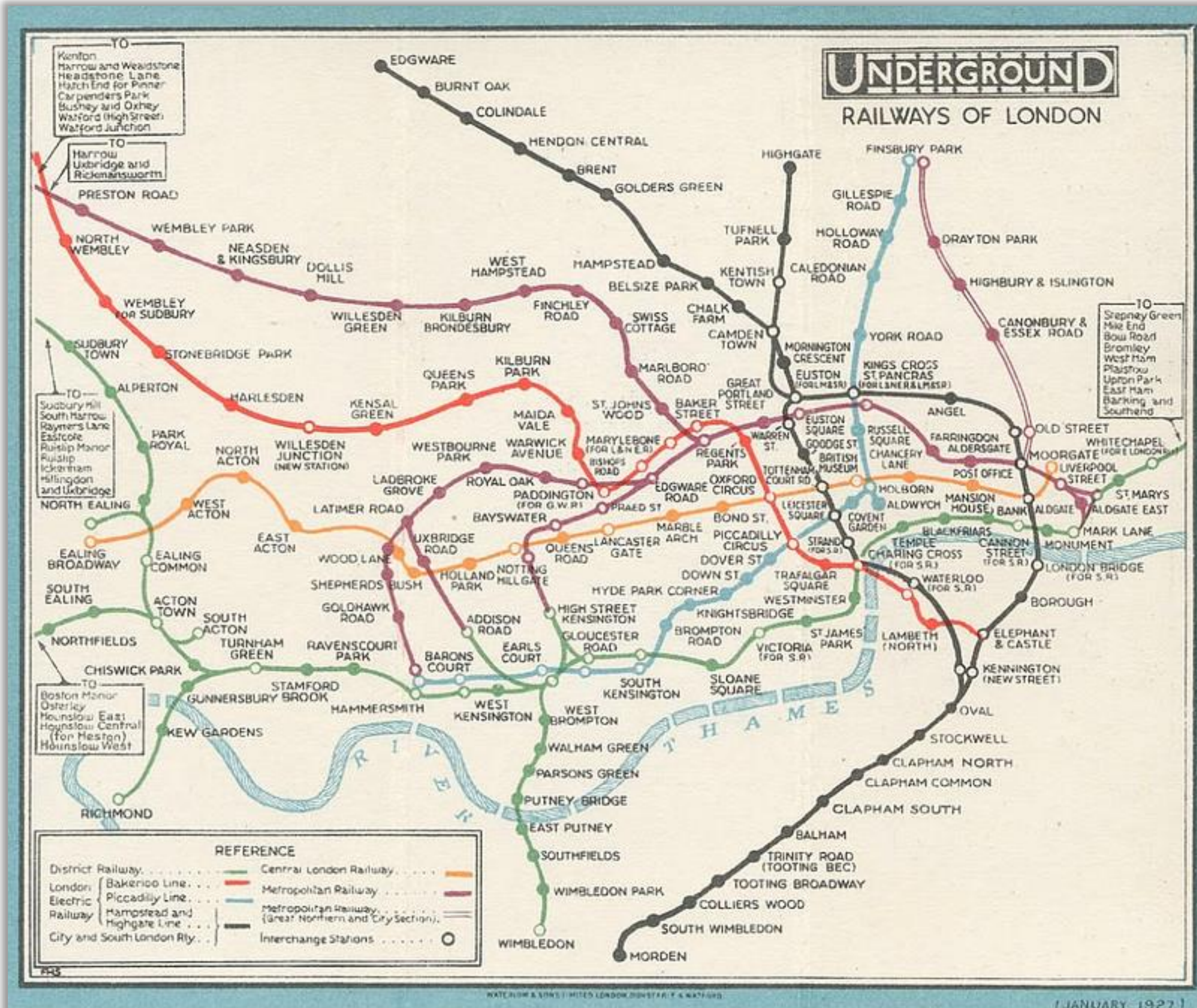
SIEMENS
Ingenuity for life

- Munzner 2014:

  "Computer-based visualization systems provide visual representations of datasets designed to help people carry out **tasks** more **effectively**"



- Why focus on tasks & effectiveness?
  – What problem do we want to solve?

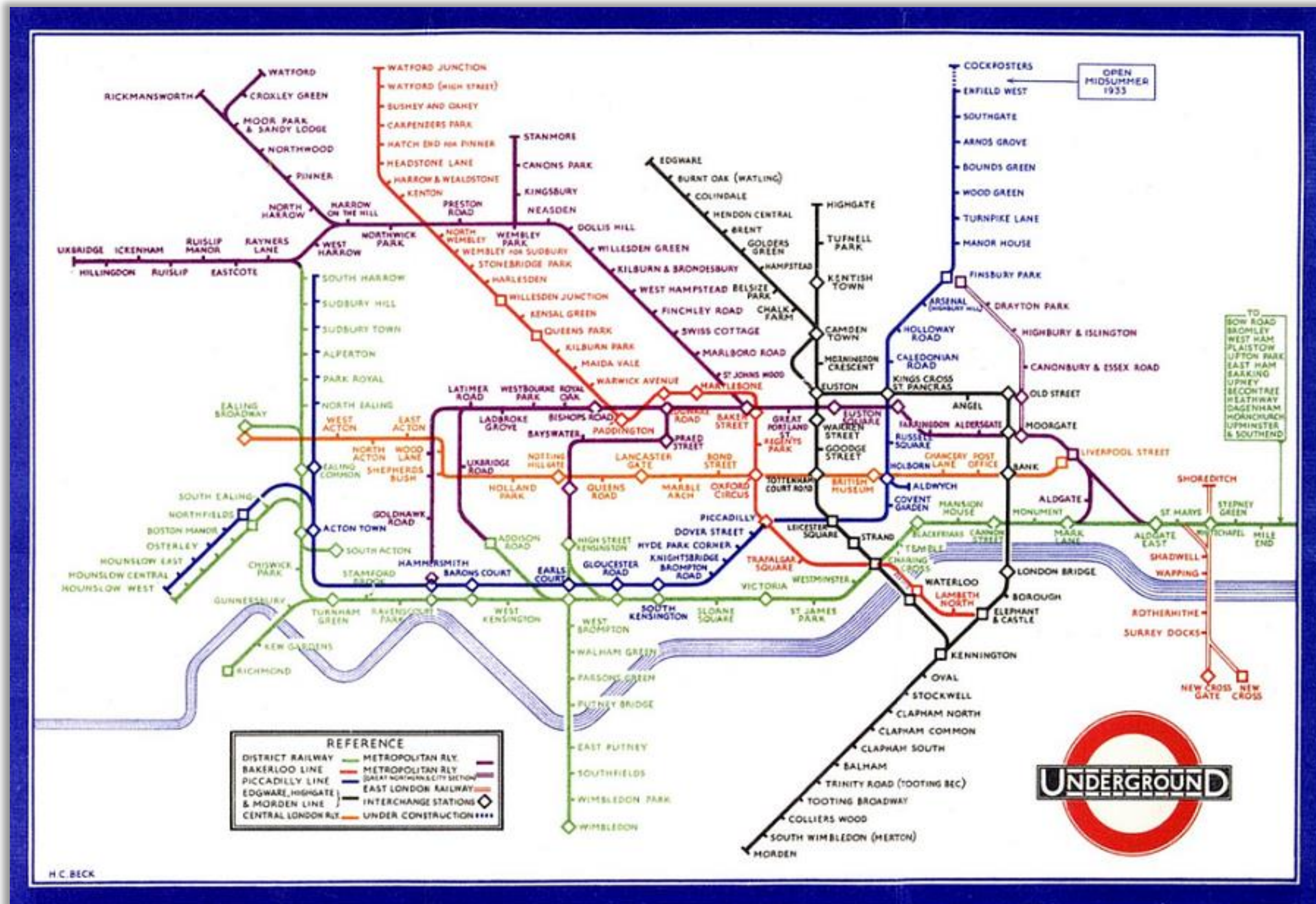


London underground map
Fred Stingemore (1927)

Which station is closest to me?

Visualization – Definitions

SIEMENS
Ingenuity for life



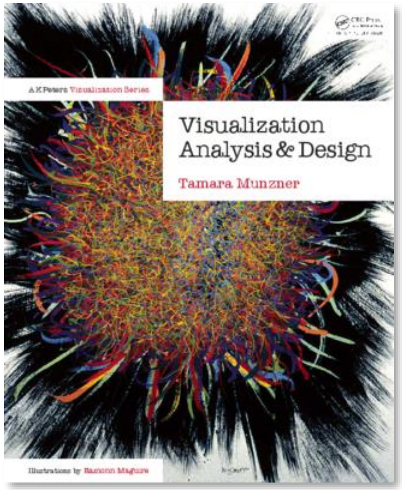
Redesign by Harry Beck (1933)

What's the fastest path from A to B?

Visualization – Definitions

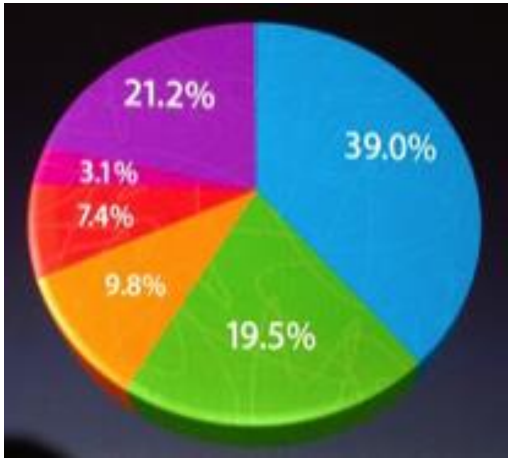
- Munzner 2014:

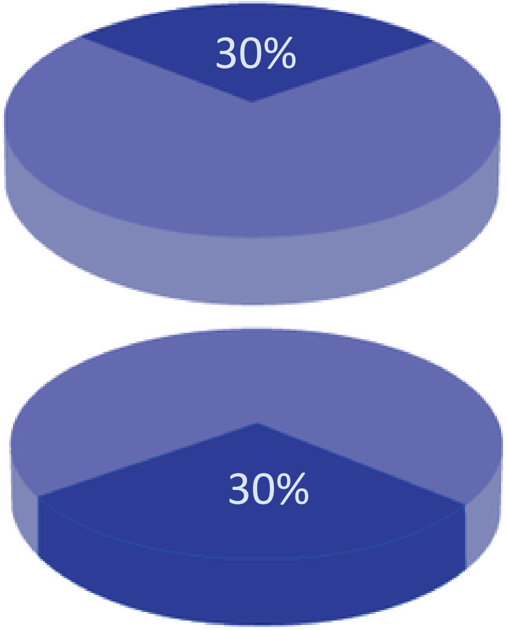
  "Computer-based visualization systems provide visual representations of datasets designed to help people carry out **tasks** more **effectively**"



- Why focus on tasks & effectiveness?
  - What problem do we want to solve?
  - A tool serving one task can be poorly suited for another one
  - Most possibilities are ineffective
  - Representation should be correct, accurate, and truthful

Visualization – Definitions



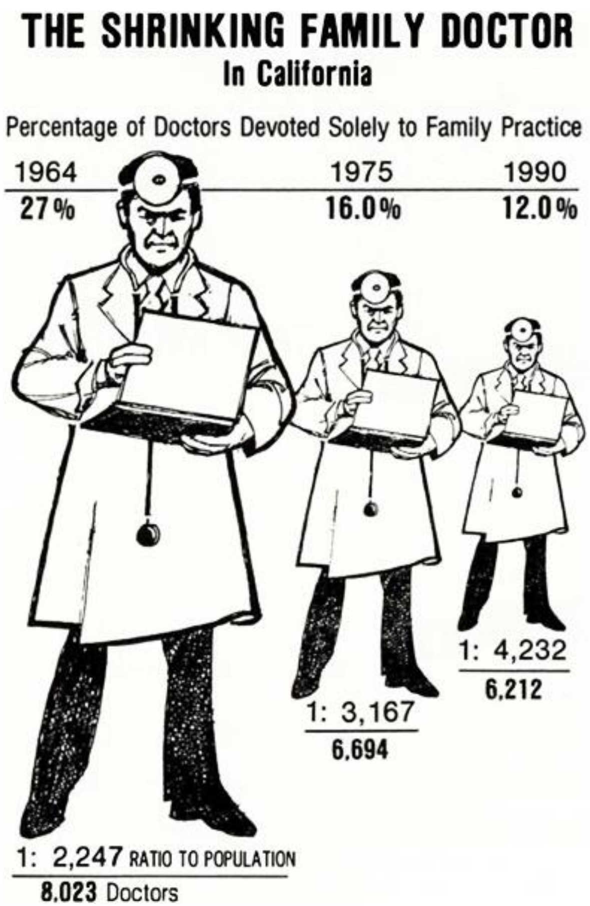


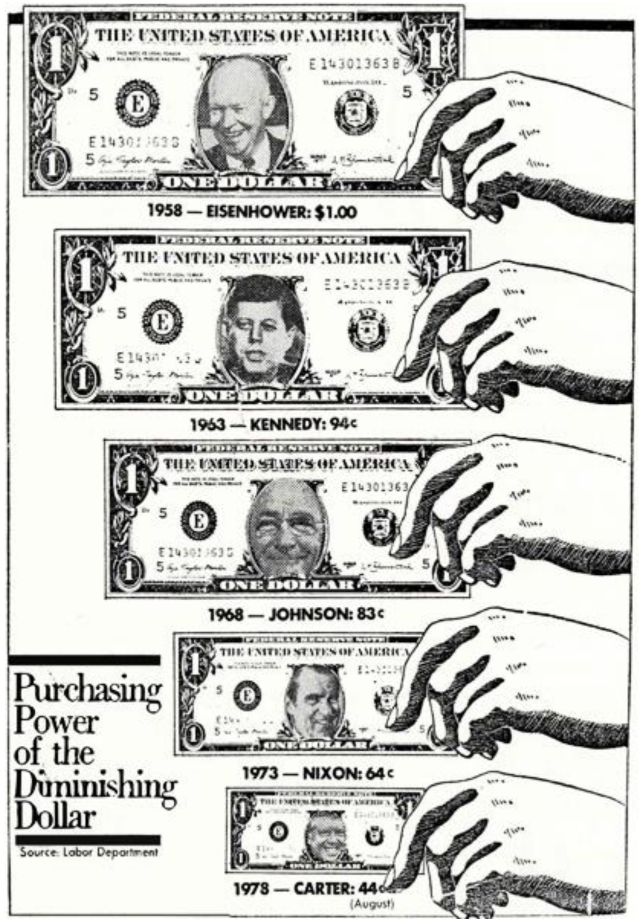
Since when is 19.5% bigger than 21.2%?

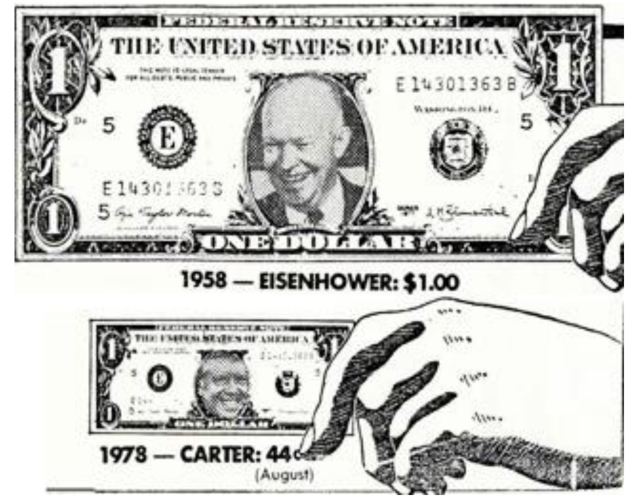
– Representation should be correct, accurate, and **truthful**

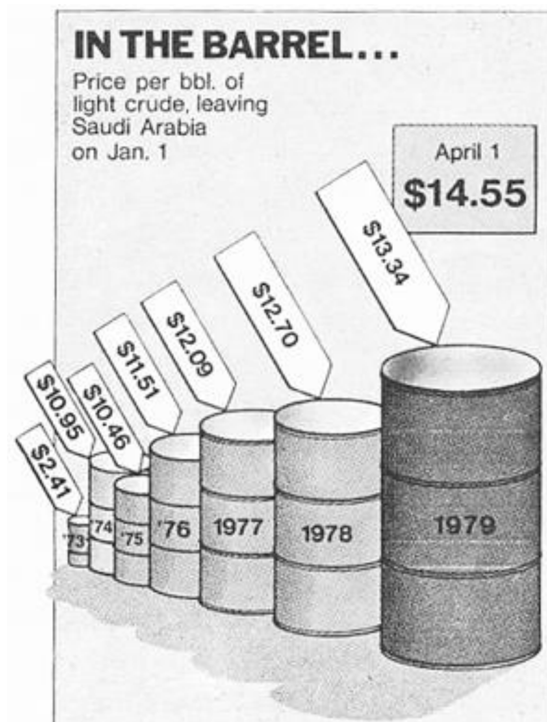
Visualization – Definitions

SIEMENS
Ingenuity for life









– Representation should be correct, accurate, and **truthful**

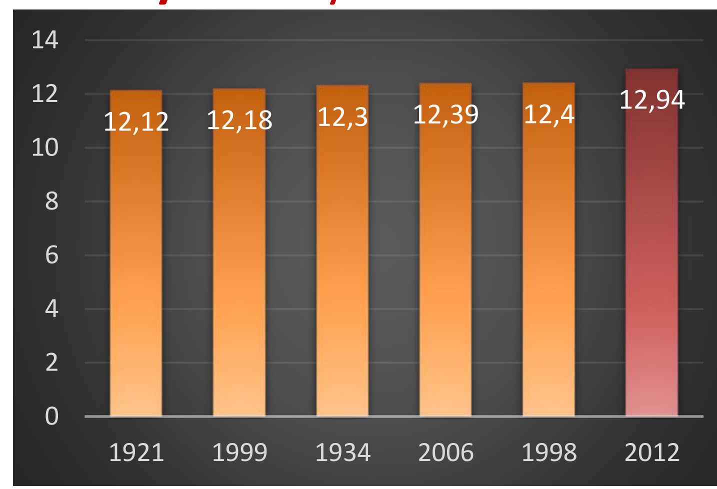
$$\text{Lie factor} = \frac{\text{Size of effect shown in graphic}}{\text{Size of effect in data}}$$

Edward Tufte

Visualization – Definitions

SIEMENS
Ingenuity for life

Always start your bars at zero!



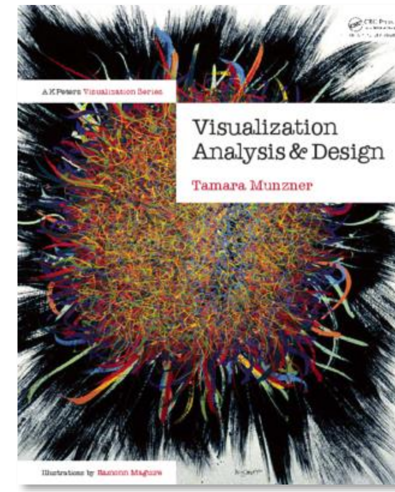
– Representation should be correct, accurate, and **truthful**

www.climatecentral.org/news/noaa-2012-was-warmest-and-second-most-extreme-year-on-record-15436

Visualization – Definitions

- Munzner 2014:

  > “Visualization designers must take into account three very different kinds of resource limitations: those of **computers**, of **humans**, and of **displays**”



- Computational limits
  - Processing time / system memory
- Display limits
  - Number of pixels
  - Information density: ratio of used space vs. unused whitespace
- Human limits
  - Perception, attention and memory (e.g., change blindness)

Visualization – Definitions

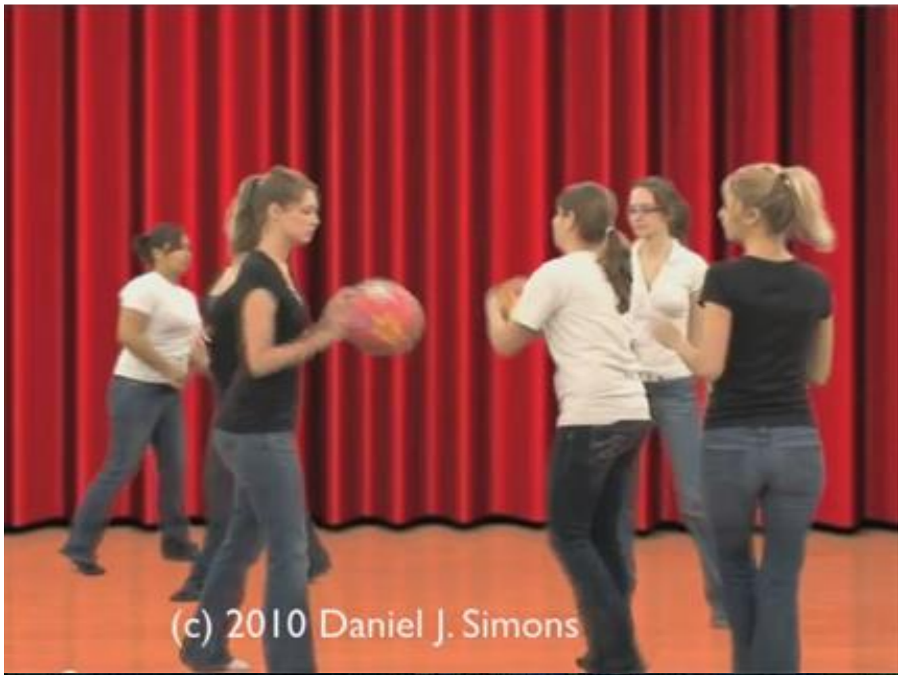
- Human limits
  - Perception, attention and memory (e.g., change blindness)

Visualization – Definitions

- Human limits
  - Perception, attention and memory (e.g., change blindness)

Visualization – Definitions

SIEMENS
Ingenuity for life



- Human limits
  - Perception, attention and memory (e.g., change blindness)

Visualization – Goals

SIEMENS
Ingenuity for life

Visualization is good for

– **Visual exploration**
  - find unknown/unexpected
  - generate new hypotheses

Nothing is known about the data

– **Visual analysis** (confirmative vis.)
  - confirm or reject hypotheses
  - information drill-down

There are hypotheses

– **Presentation**
  - effective/efficient communication of results

"Everything" is known

Visualization – Major Areas

SIEMENS
Ingenuity for life

- Major areas

  – Volume Visualization
  – Flow Visualization

Scientific Visualization

Inherent spatial reference

3D

  – Information Visualization
  – Visual Analytics

nD

Usually no spatial reference

Visualization – Major Areas

- Nowadays, borders between SciVis / InfoVis / Visual Analytics are not that well defined any more
  - Information visualization may also include spatial data (e.g., geographic data)
  - Need for integration of abstract data in spatial data

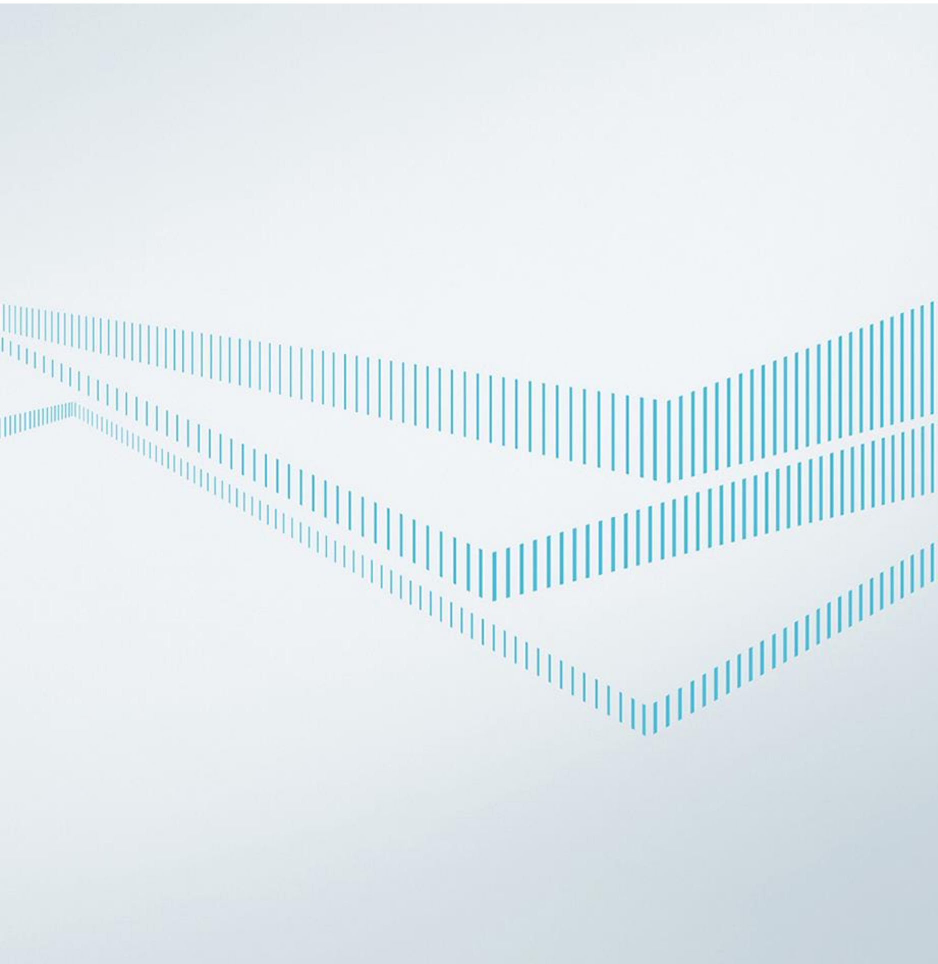
Acknowledgements

SIEMENS
Ingenuity for life

- M. Eduard Gröller
- Helwig Hauser
- Torsten Möller
- Tamara Munzner
- many more

Contact information

SIEMENS
Ingenuity for life

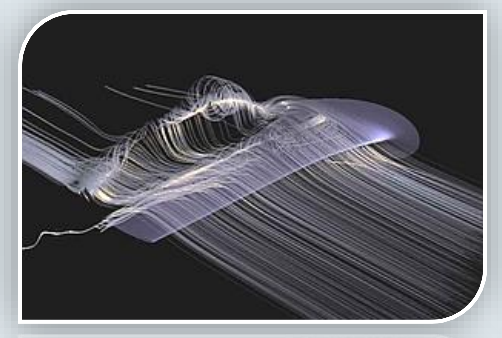
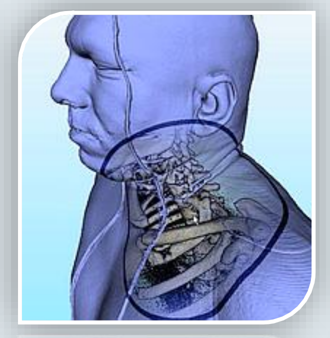


Dr. Johannes Kehrer

Siemens Technology
T RTA BAM IBI-DE
Otto-Hahn-Ring 6
81739 München, Deutschland

E-mail:
kehrer.johannes@siemens.com

Internet
siemens.com/innovation


Visual Data Analytics

Visualization

Dr. Johannes Kehrer – Siemens Technology, Munich

Disclaimer

These lecture slides and materials are for personal use by students only and are intended for educational purposes only. Distribution or reproduction of the material for any other purposes is prohibited.

The views and opinions expressed by any individual during the course of these lecturers are their own and do not necessarily represent the views, opinions, or positions of Siemens or the Technical University of Munich.

Although reasonable efforts have been made to ensure the information contained in these lecture slides is current and accurate, we do not assume any responsibility for the content of the posted material.

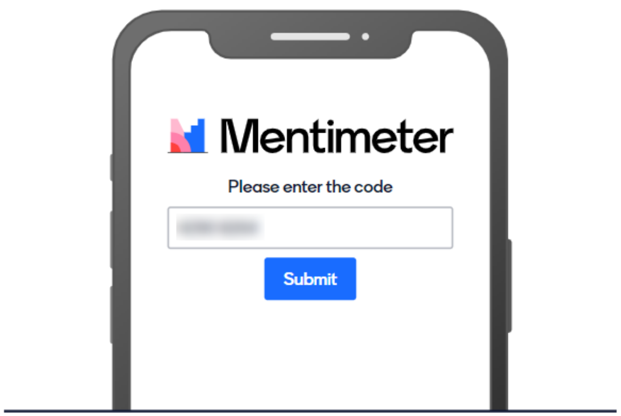
These slides also contain links to websites of third parties. As the content of these websites is not under our control, we cannot assume any liability for such external content.

Any comments/questions?

SIEMENS
Ingenuity for life

Go to

www.menti.com



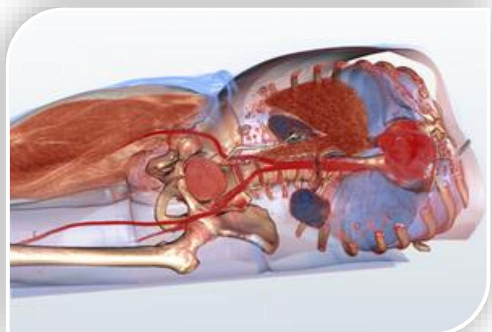
Enter the code

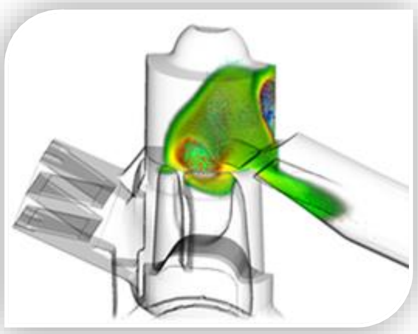
7582 8825

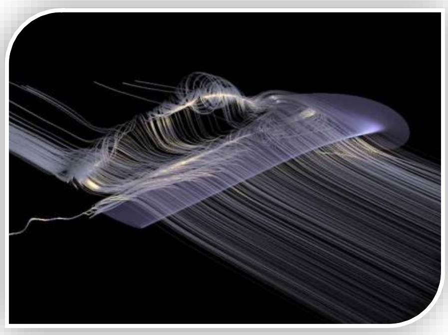
Or use QR code

Visualization basics

Data sources & examples





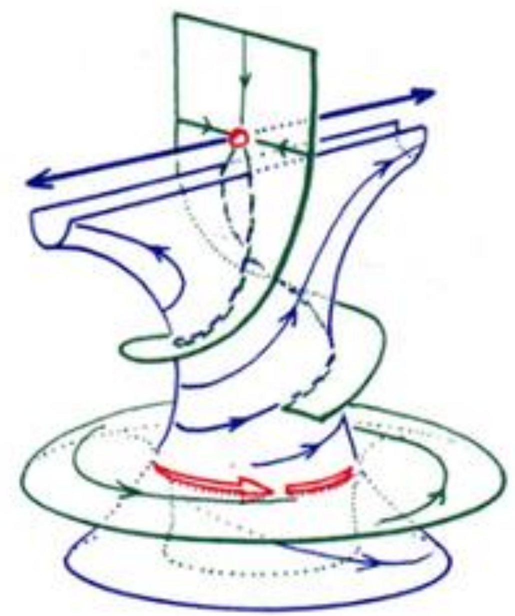


Data sources

SIEMENS
Ingenuity for life

- Data may come from any source with arbitrary size
  - **Challenge**: efficiently visualize large-scale data sets and new data types

- Real world
  - Measurements and observation
- Theoretical world
  - Mathematical and technical models



- Artificial world
  - Data that is designed

Data sources

SIEMENS
Ingenuity for life

- Real-world measurements
  - Medical Imaging (MRI, CT, PET)
  - Geographical information systems (GIS)
  - Electron microscopy

    TB
  - Meteorology and environmental sciences (satellites)
  - Seismic data
  - Crystallography

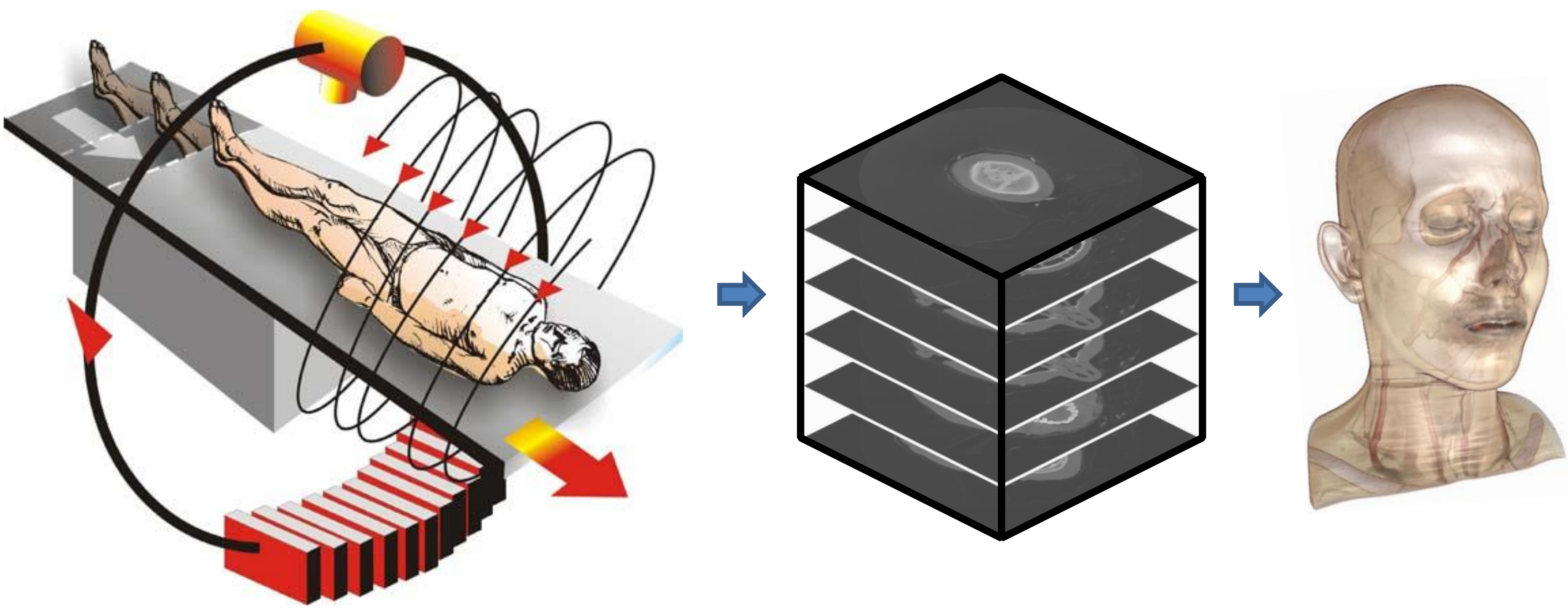
    PB
  - High energy physic
  - Astronomy (e.g. Hubble Space Telescope 100MB/day)

    PB

Data sources – medical imaging

SIEMENS
Ingenuity for life

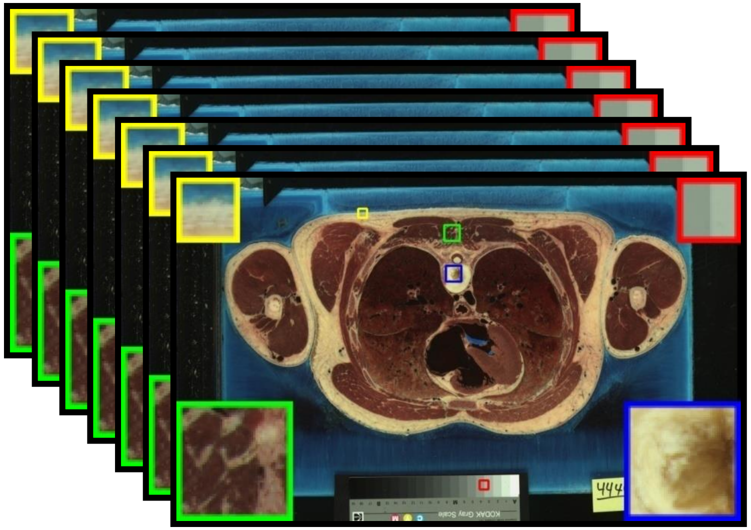
- Visualization of medical data sets
  - Provide insight into 3D scans



Data sources – medical imaging

SIEMENS
Ingenuity for life

- Visualization of medical data sets
  - The Visible Human Project
    - www.nlm.nih.gov/research/visible/visible_human.html
    - Bodies were frozen in a special material to preserve tissues and organs
    - Sections were 'shaved' off the frozen block in micro-thin layers to expose underlying tissues
    - A picture is taken
    - A ´stack´ of 2D images is obtained

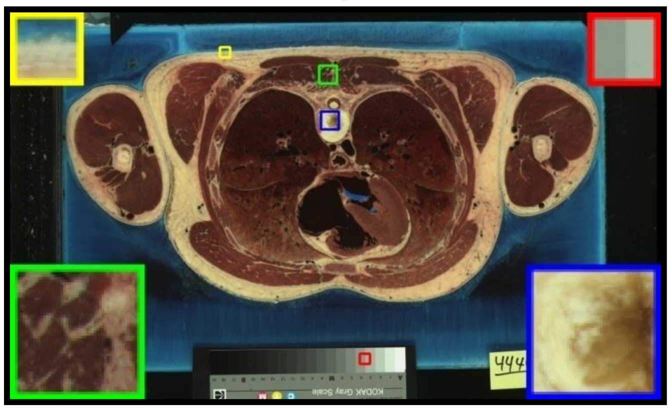


Data sources – medical imaging

SIEMENS
Ingenuity for life

- Visualization of medical data sets
  - The Visible Human Project – different imaging modalities

Color Cryosections

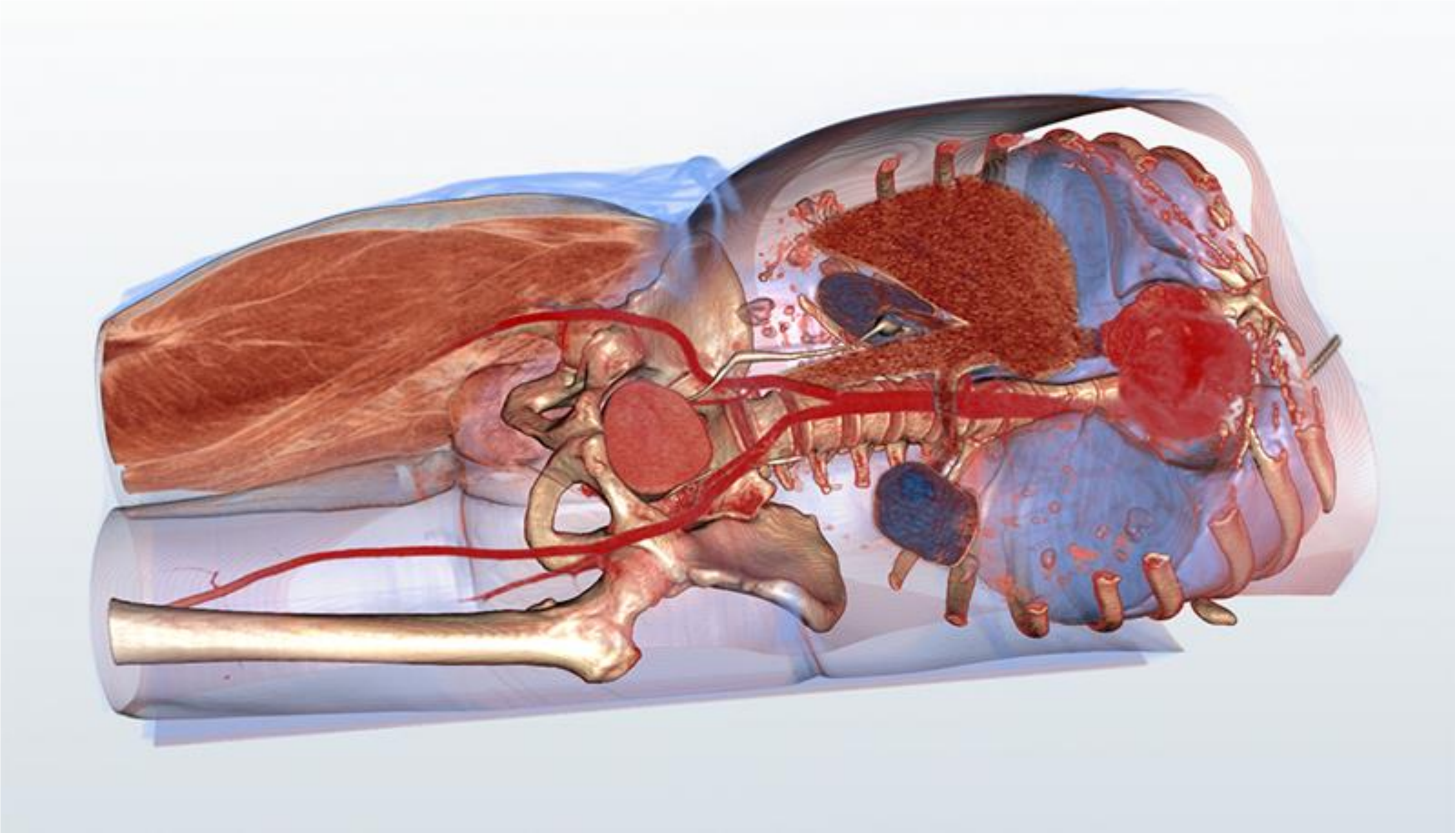


CT scan

MRI scan

Data sources – medical imaging

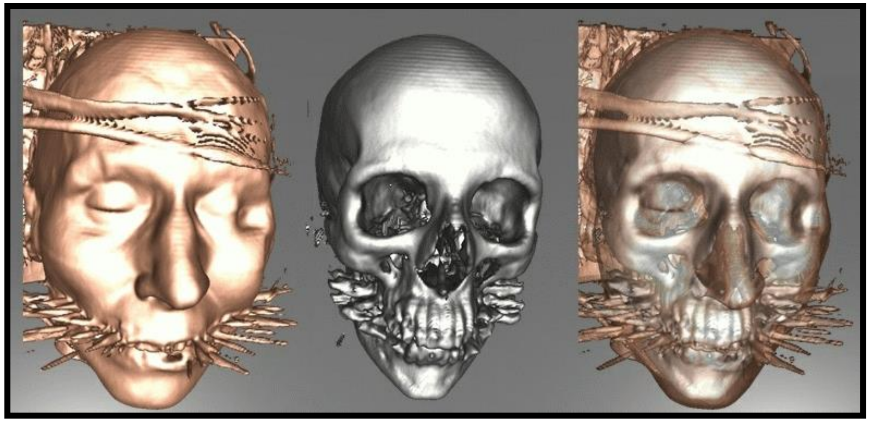
- The Visible Human Project – from 2D to 3D

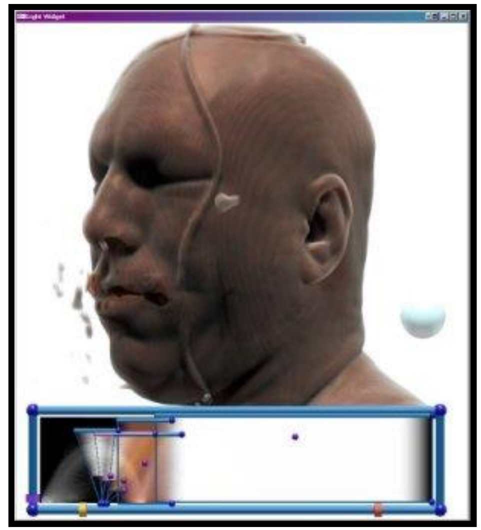


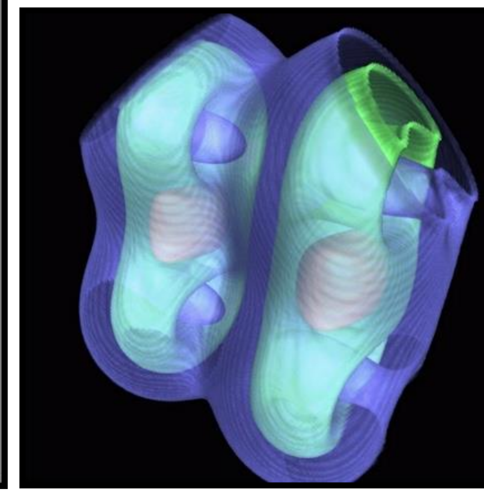
Data sources – medical imaging

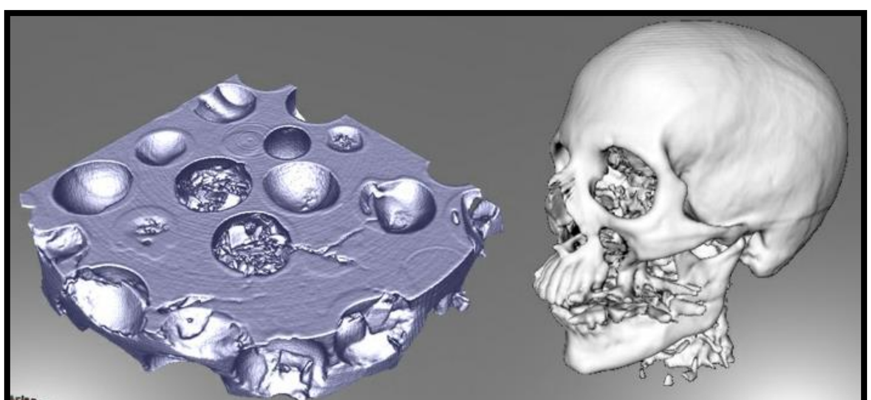
SIEMENS
Ingenuity for life

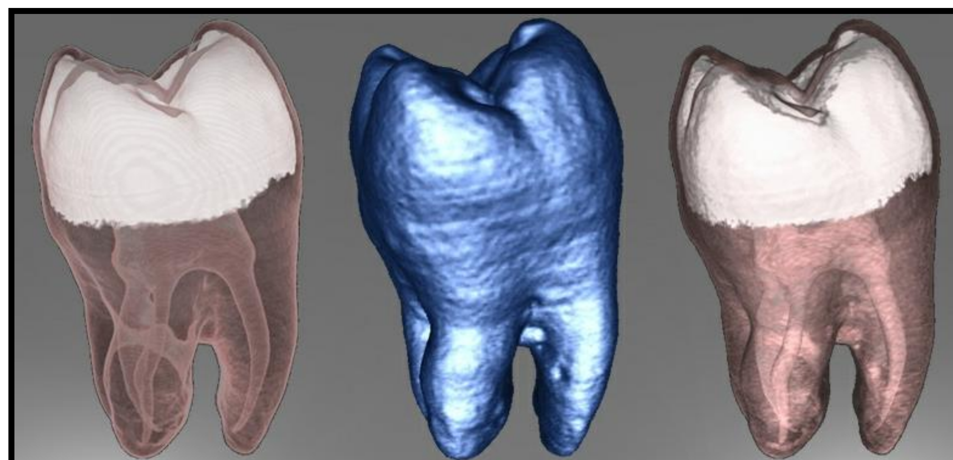
- High-quality volume visualization on consumer PC





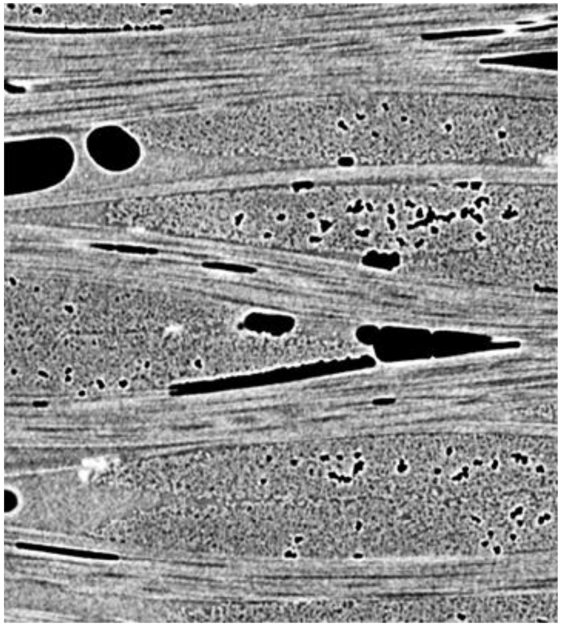




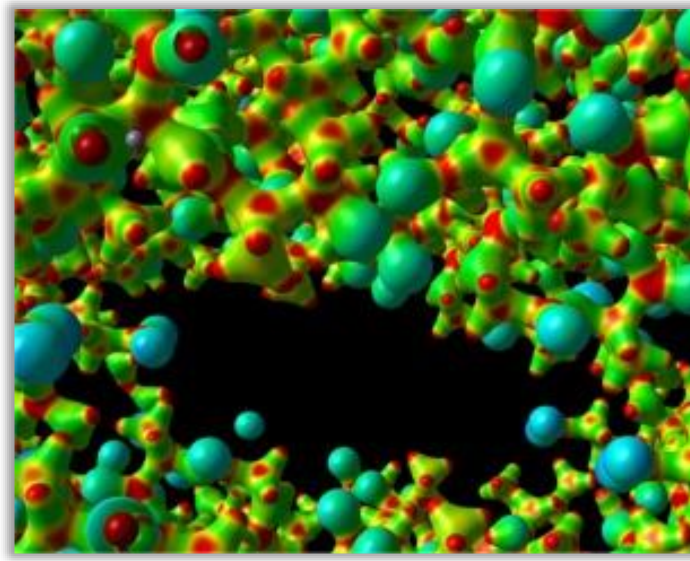


Data sources – material sciences

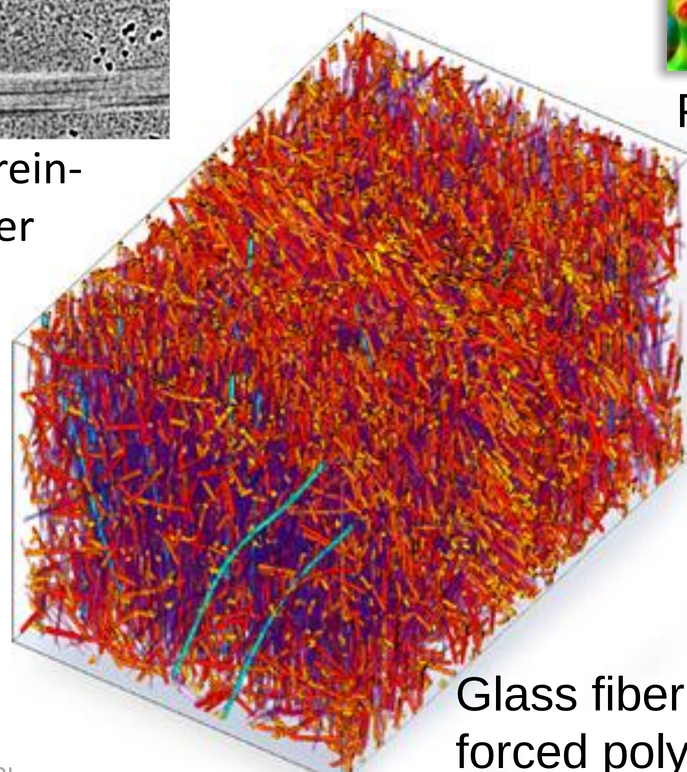
SIEMENS
Ingenuity for life



Carbon fiber rein-
forced polymer



Potential field in crystal structure



Glass fiber rein-
forced polymer

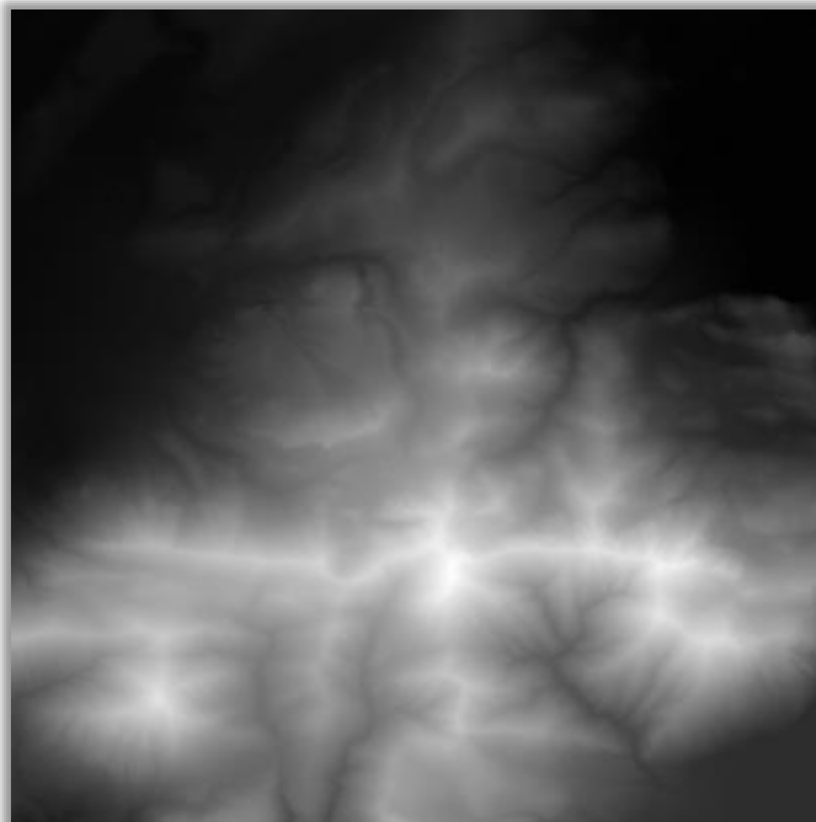
Porosity evaluation

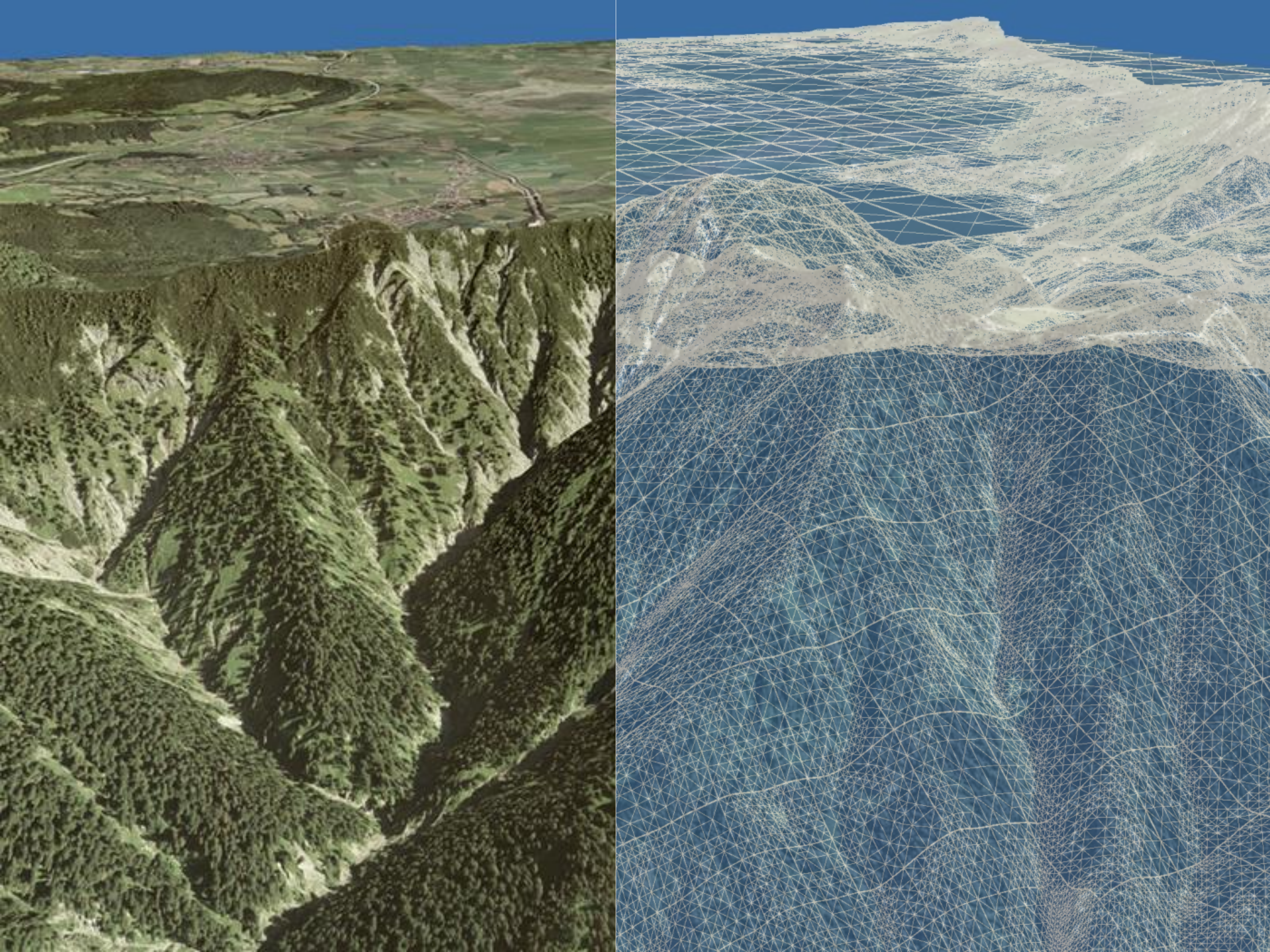
Data sources – remote sensing

SIEMENS
Ingenuity for life

Orthophoto

Digital Elevation Map





Data sources

SIEMENS
Ingenuity for life

- Computer simulations
  - Sciences
    - Molecular dynamics
    - Quantum chemistry
    - Mathematics

    TB

    - Computational physics
    - Meteorology and climate research
    - Computational fluid dynamics (CFD)

    PB

  - Engineering
    - Architectural walk-throughs

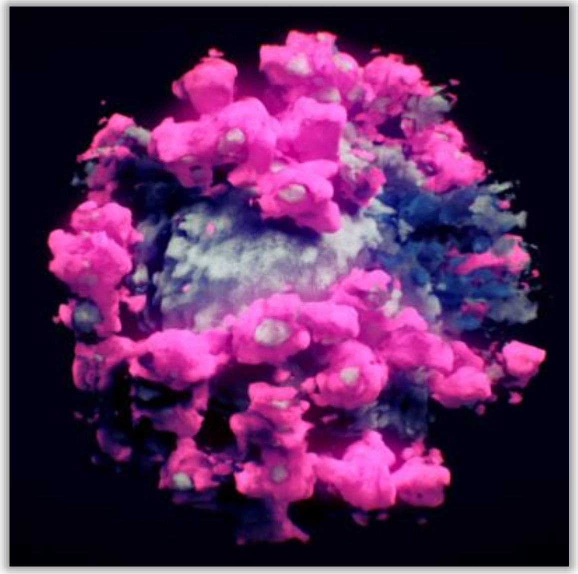
    MB

    - Structural mechanics
    - Car body design

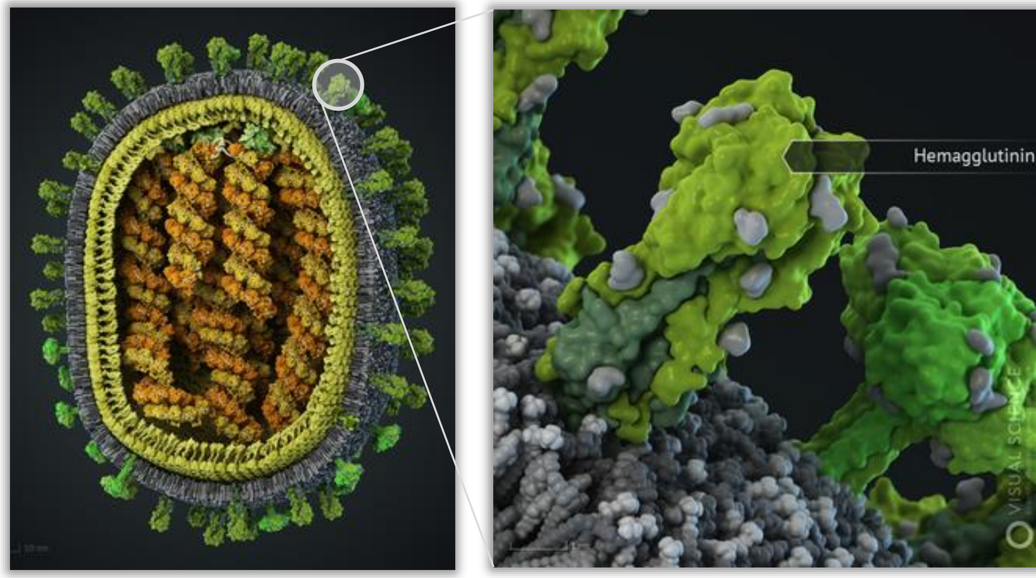
    GB

Data sources – bio sciences

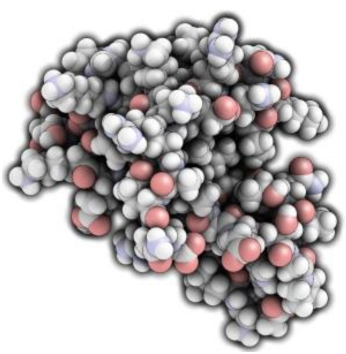
SIEMENS
Ingenuity for life



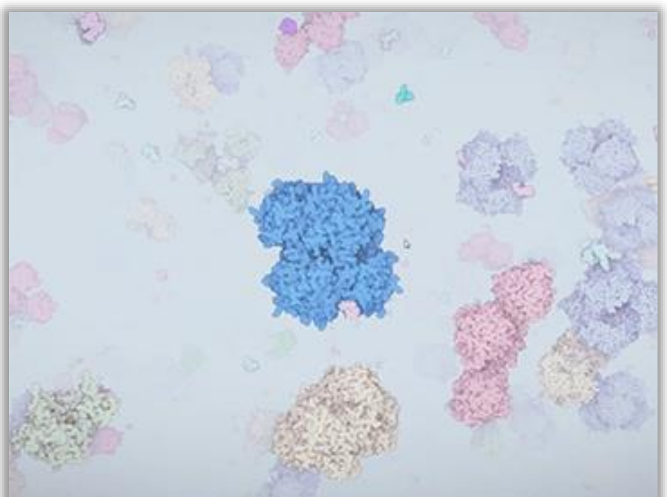
SARS-CoV-2 Cryo-ET (link)



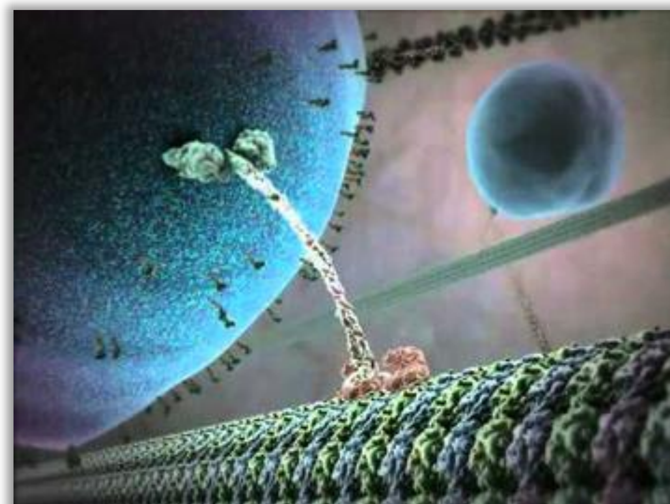
Influenza A/H1N1



Molecular visualization



Molecular reactions (link)

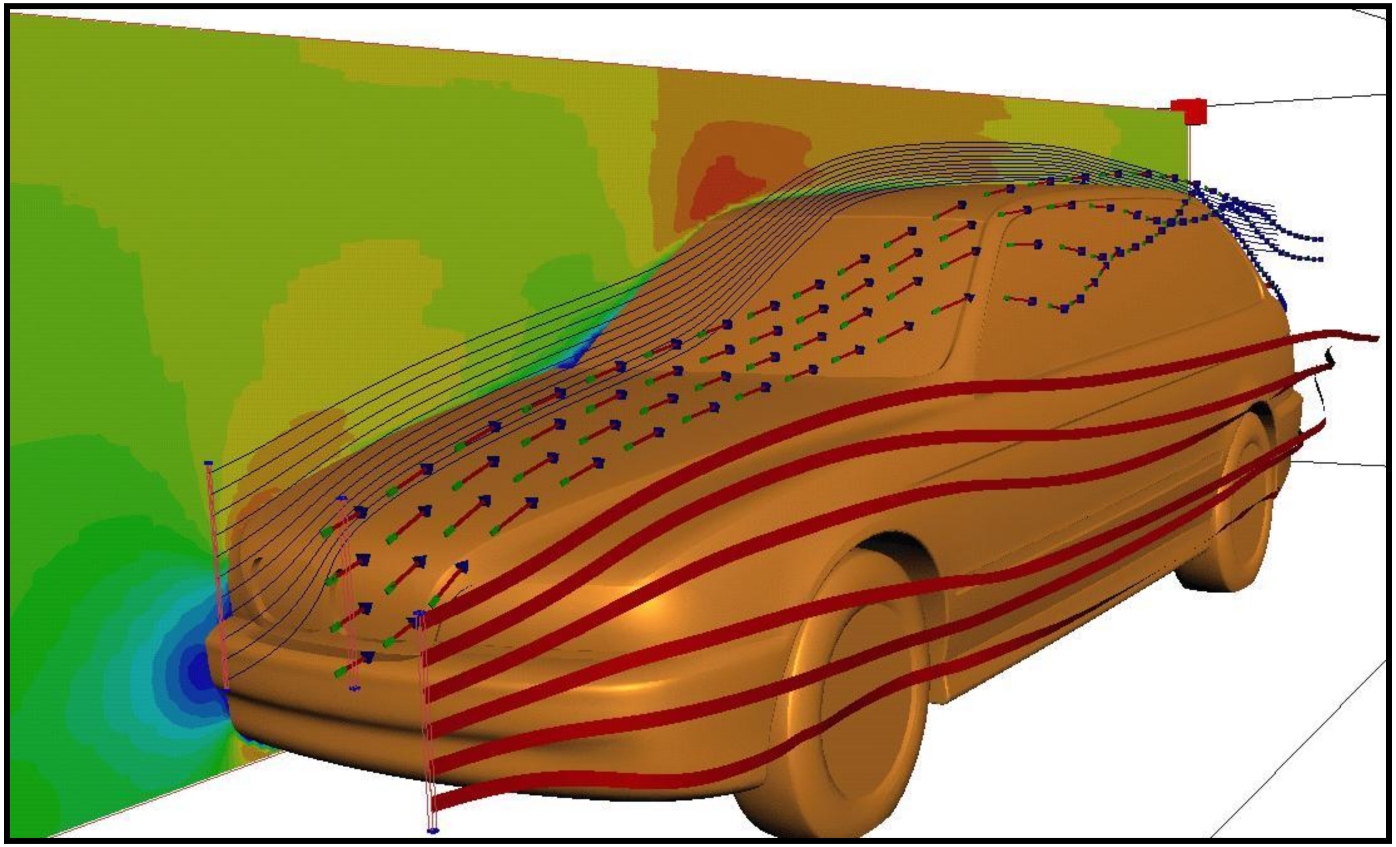


Inner life of a cell (link)

Data sources - CFD

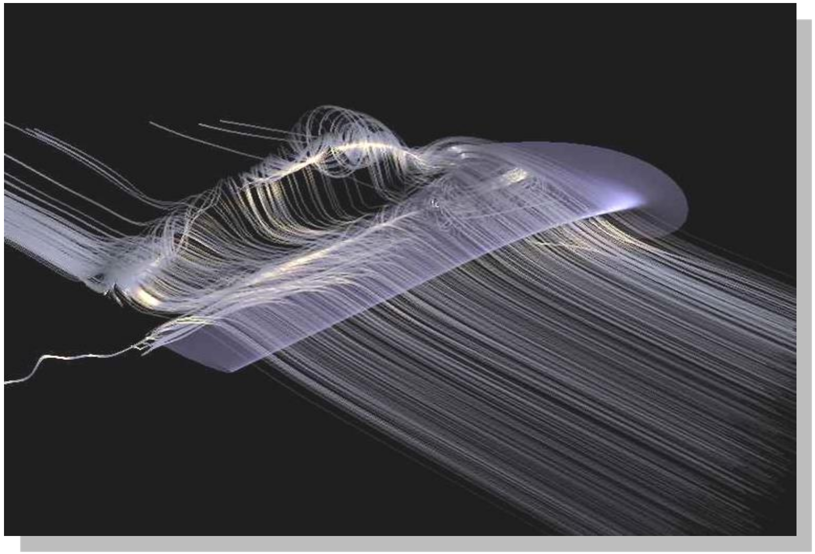
SIEMENS
Ingenuity for life

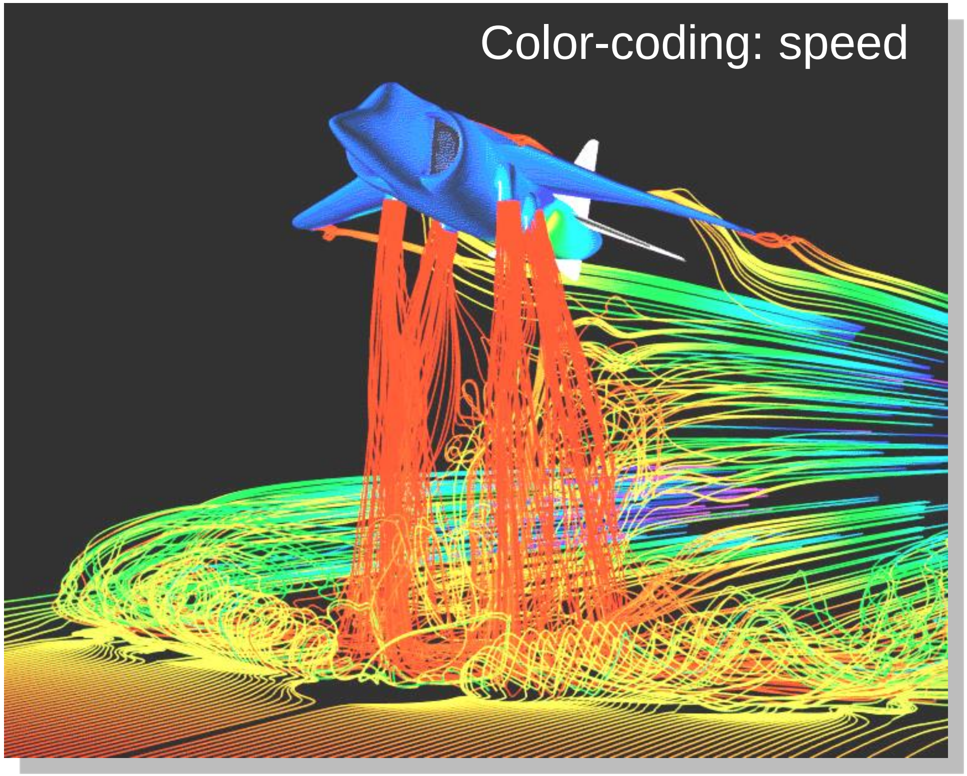
- Visualization of flow data

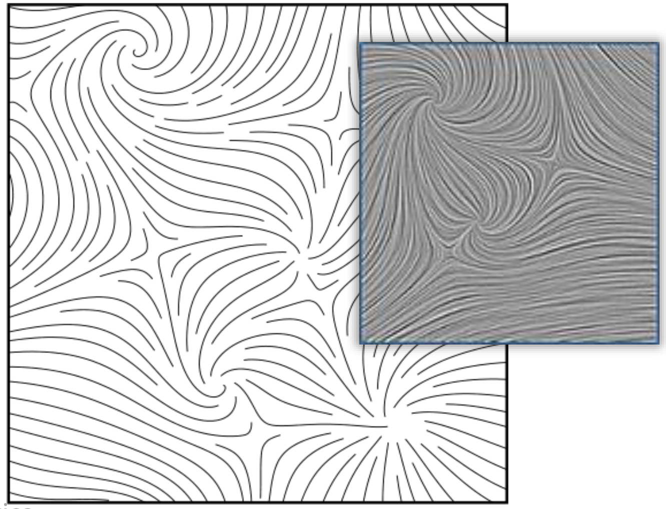


Data sources - CFD

- Visualization of flow data

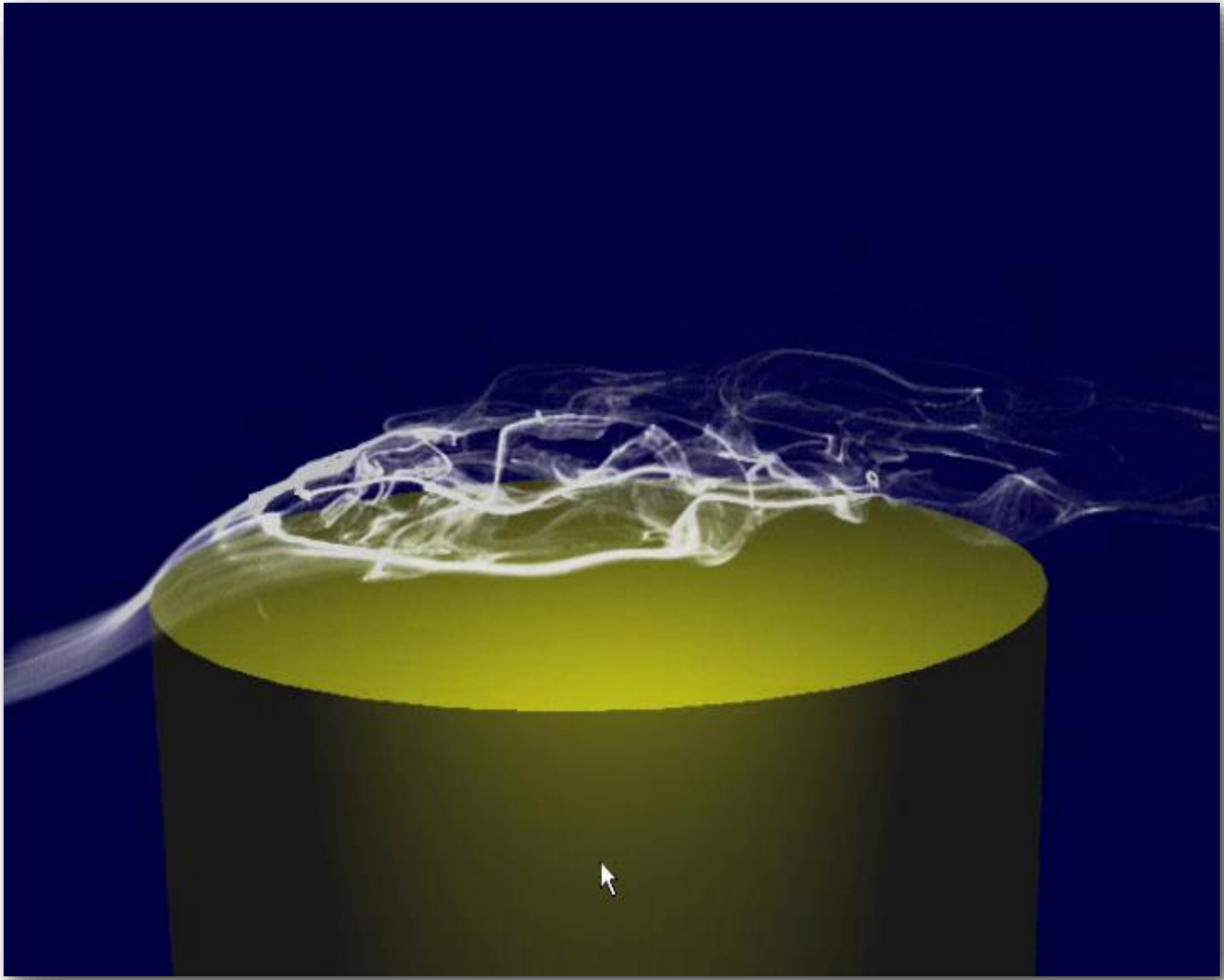






Data sources - CFD

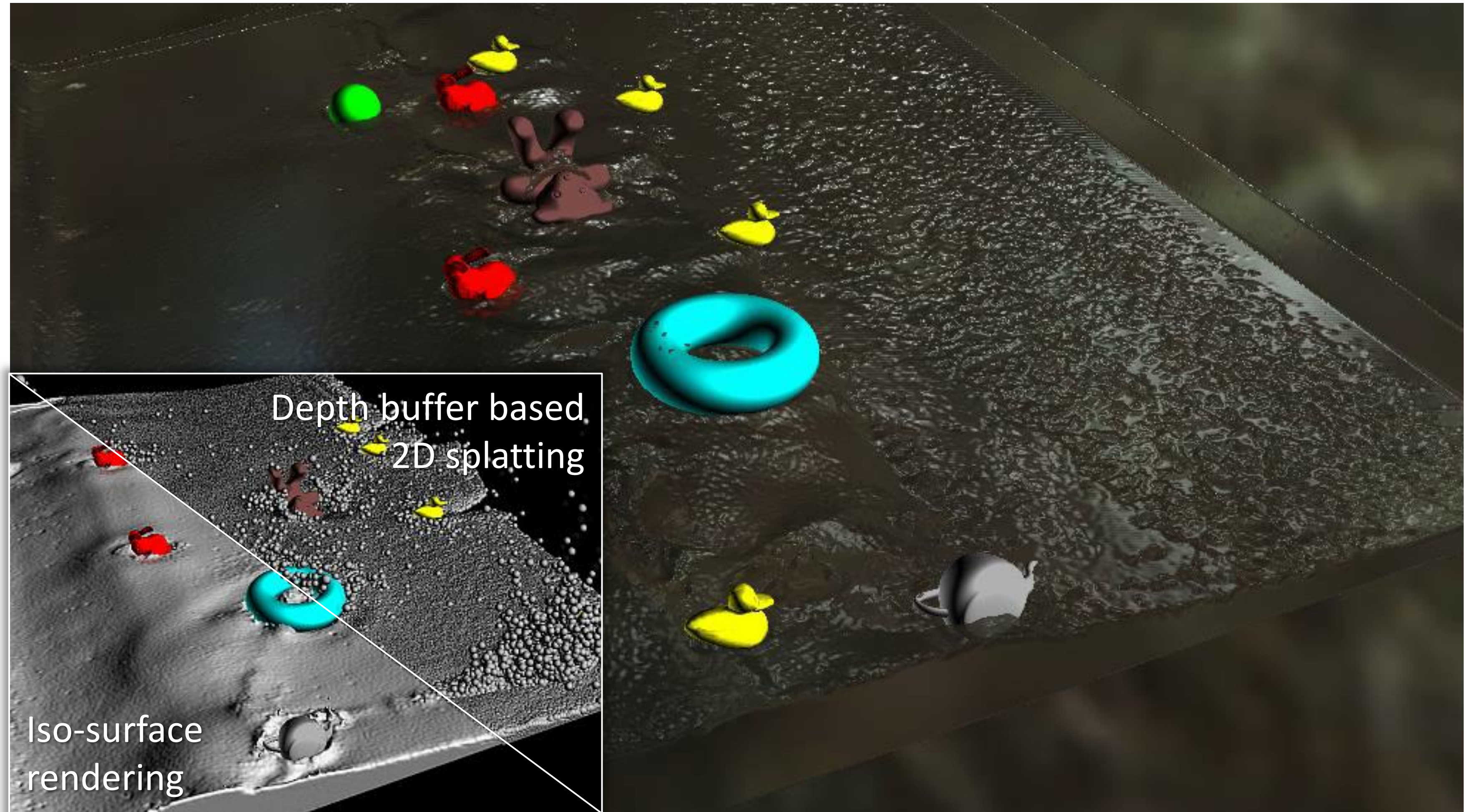
- The virtual windtunnel



Data sources - CFD

SIEMENS
Ingenuity for life

- Visualization of particle simulation data



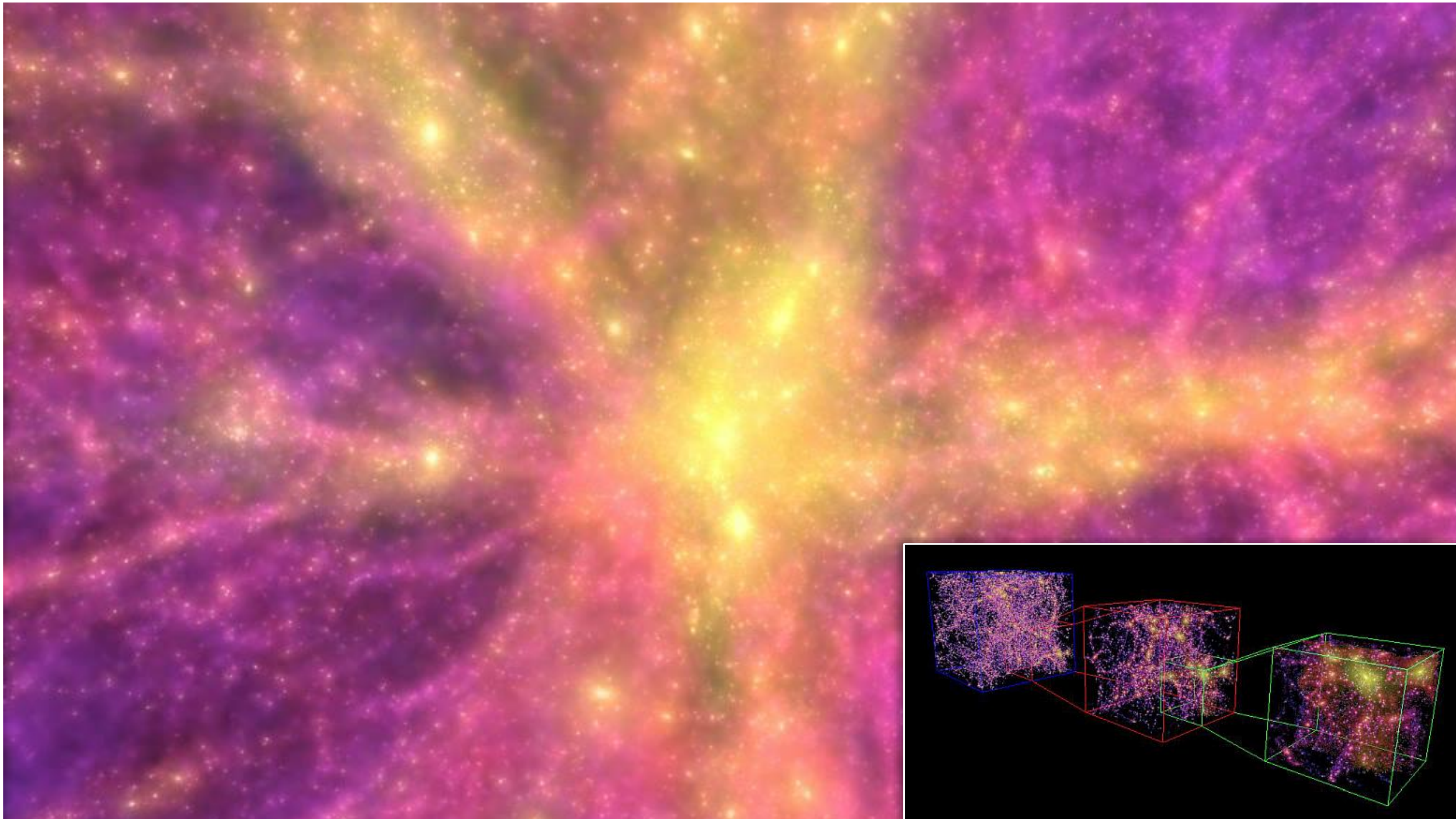
Large Wave with Moving Obstacles, 2.5 Million Particles

[Fraedrich et al. 10]

Data sources – gas dynamics

SIEMENS
Ingenuity for life

- Visualization of particle simulation data



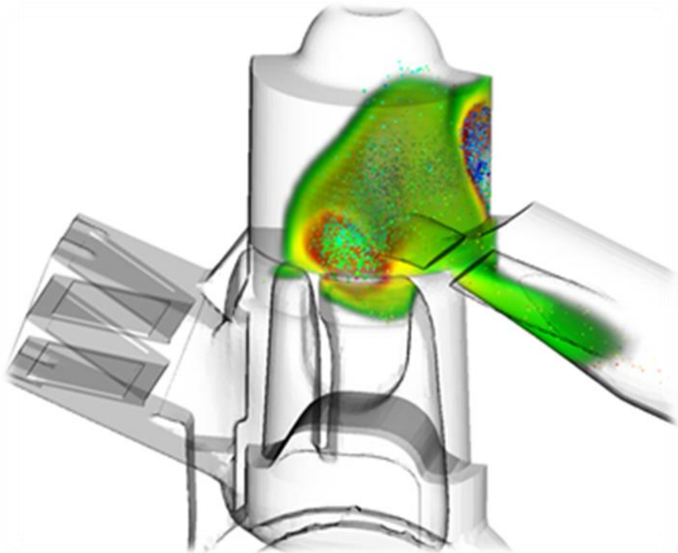
Millennium Run Dataset, 42 Million visible Particles

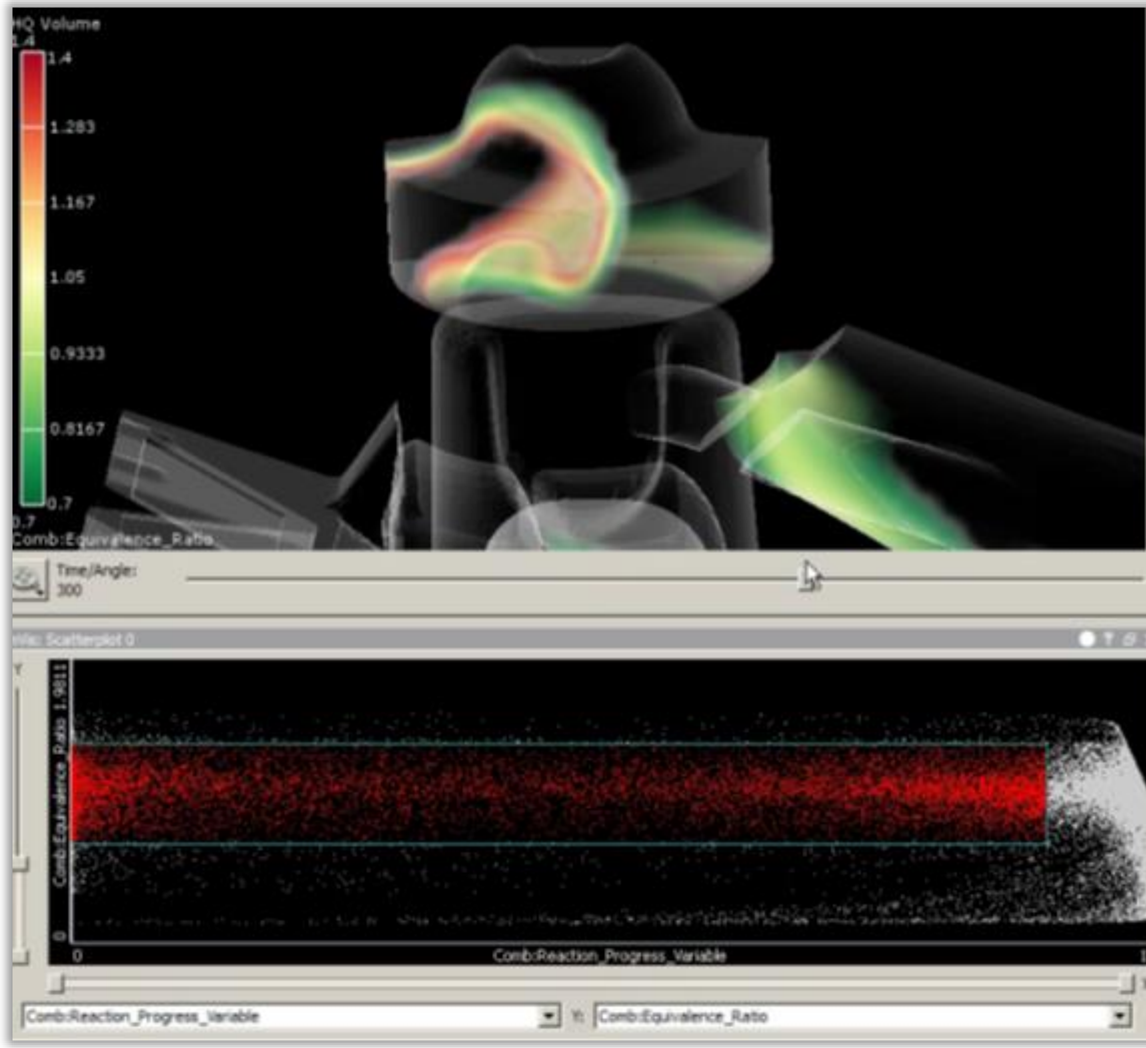
[Fraedrich et al. 10]

Data sources - CFD

SIEMENS
Ingenuity for life

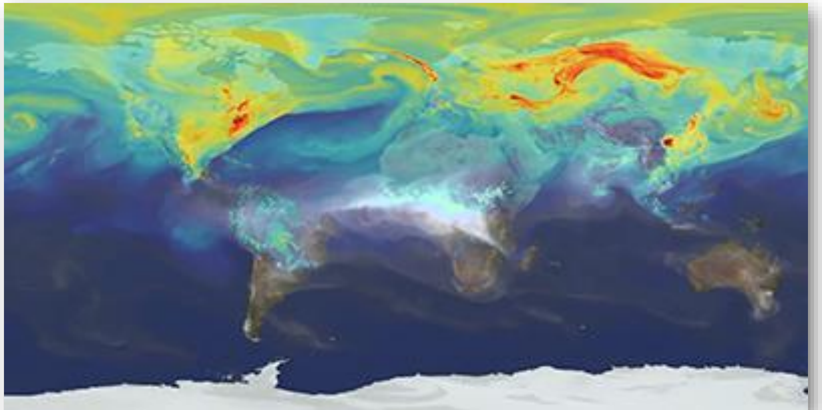
- Combustion process in a two-stroke engine



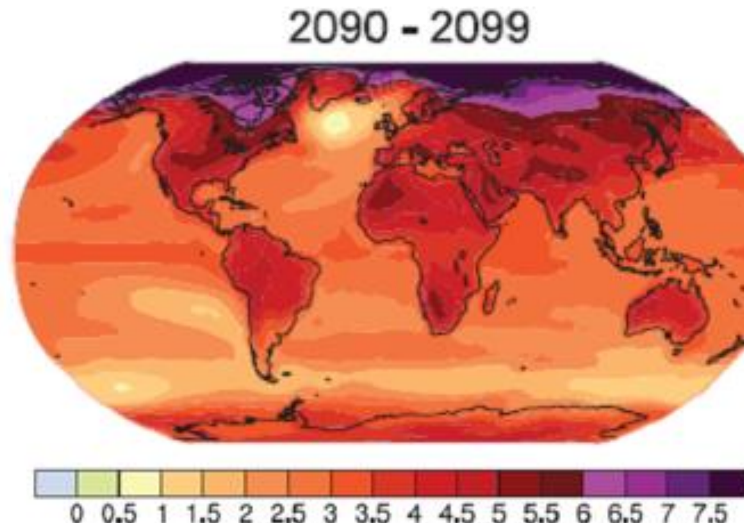


Data sources – geosciences

SIEMENS
Ingenuity for life

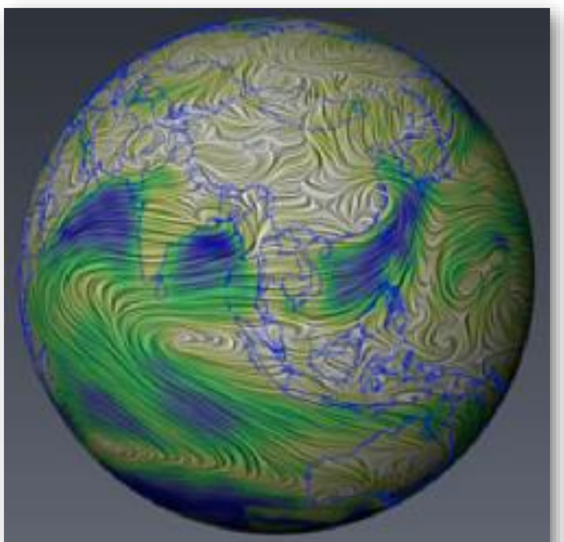


CO2 in atmospheric flow ([link]())

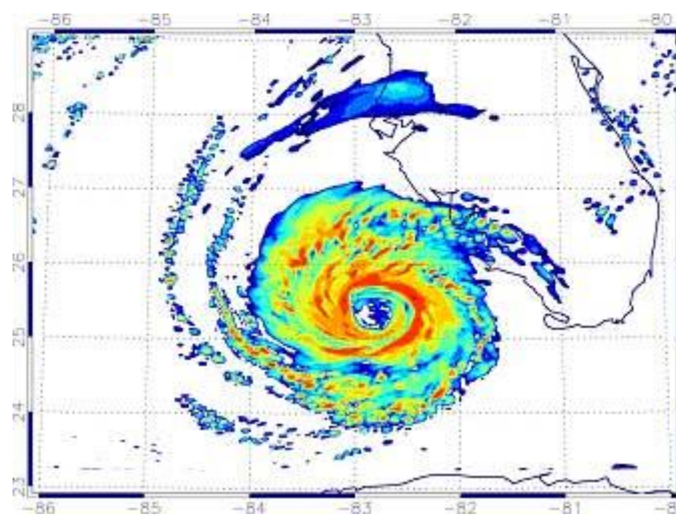


Average temp.in ten years (°C)

[from IPCC AR #4, 2007]



Wind flow paths over Earth's surface



Hurricane visualization

Data sources

SIEMENS
Ingenuity for life

- Theoretical world
- Computer simulations
  - Commercial
    - Business graphics *TB*
    - Economic models
    - Financial modeling *GB*
- Artificial world
  - Drawings
  - Painting
  - Publishing *MB*
  - TV (teasers, commercials) *GB*
  - Movies (animations, special effects) *TB*

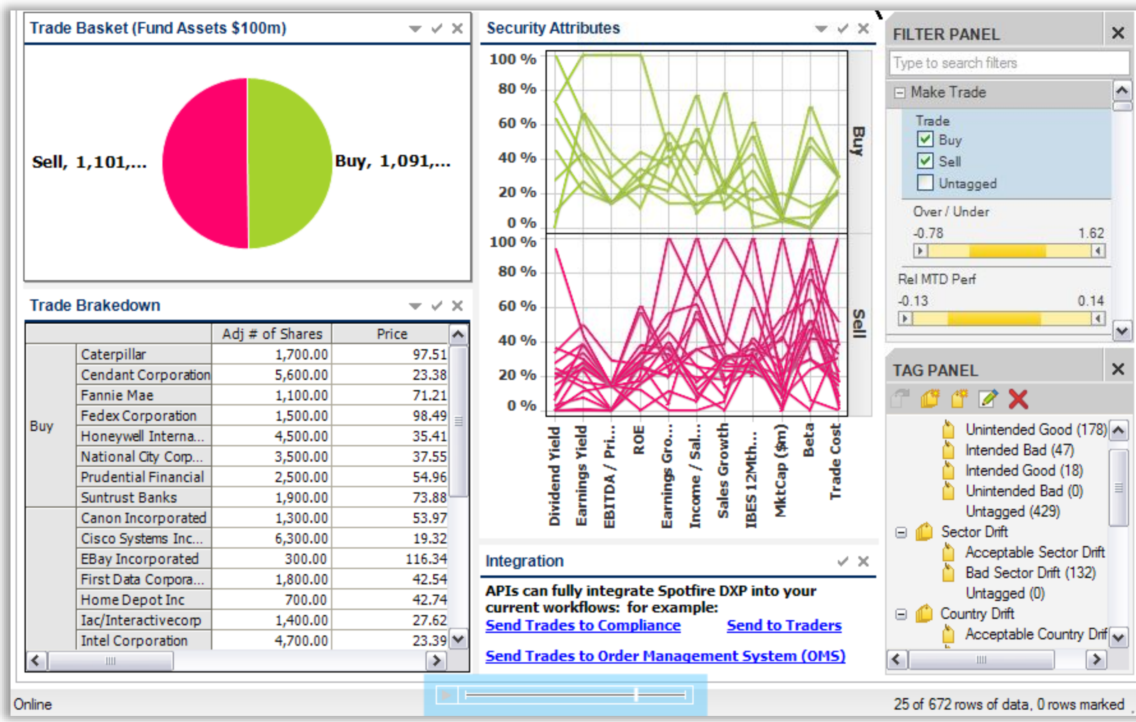
Data sources

SIEMENS
Ingenuity for life

- Information systems
  - Stock market
    (billions of transactions per day)
  - Social Networks
  - World Wide Web !!!

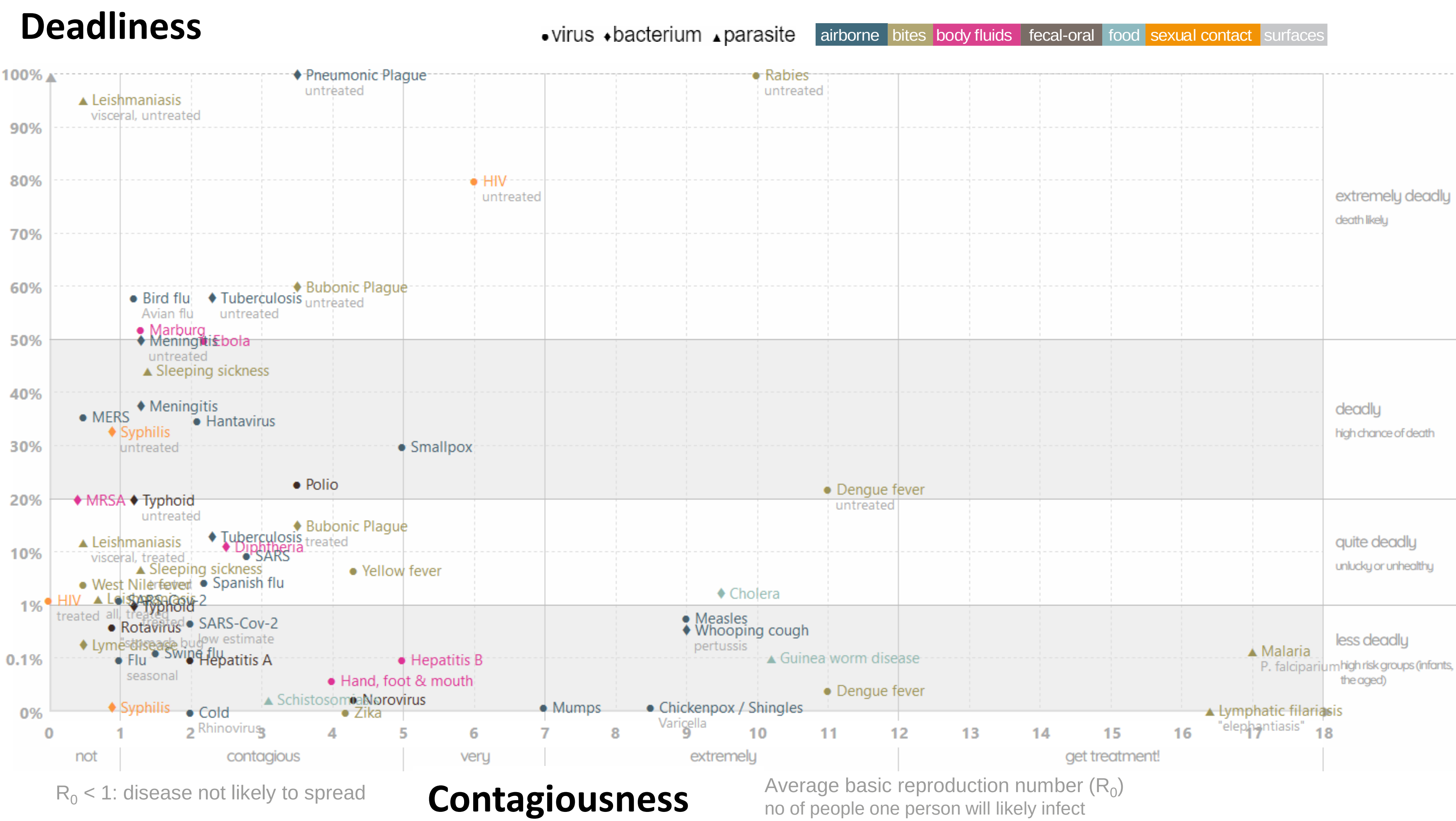
PB

- Examples
  - Spotfire
  - Tableau
  - Google Analytics



Infectious Diseases in Context

David McCandless et al. (2014)

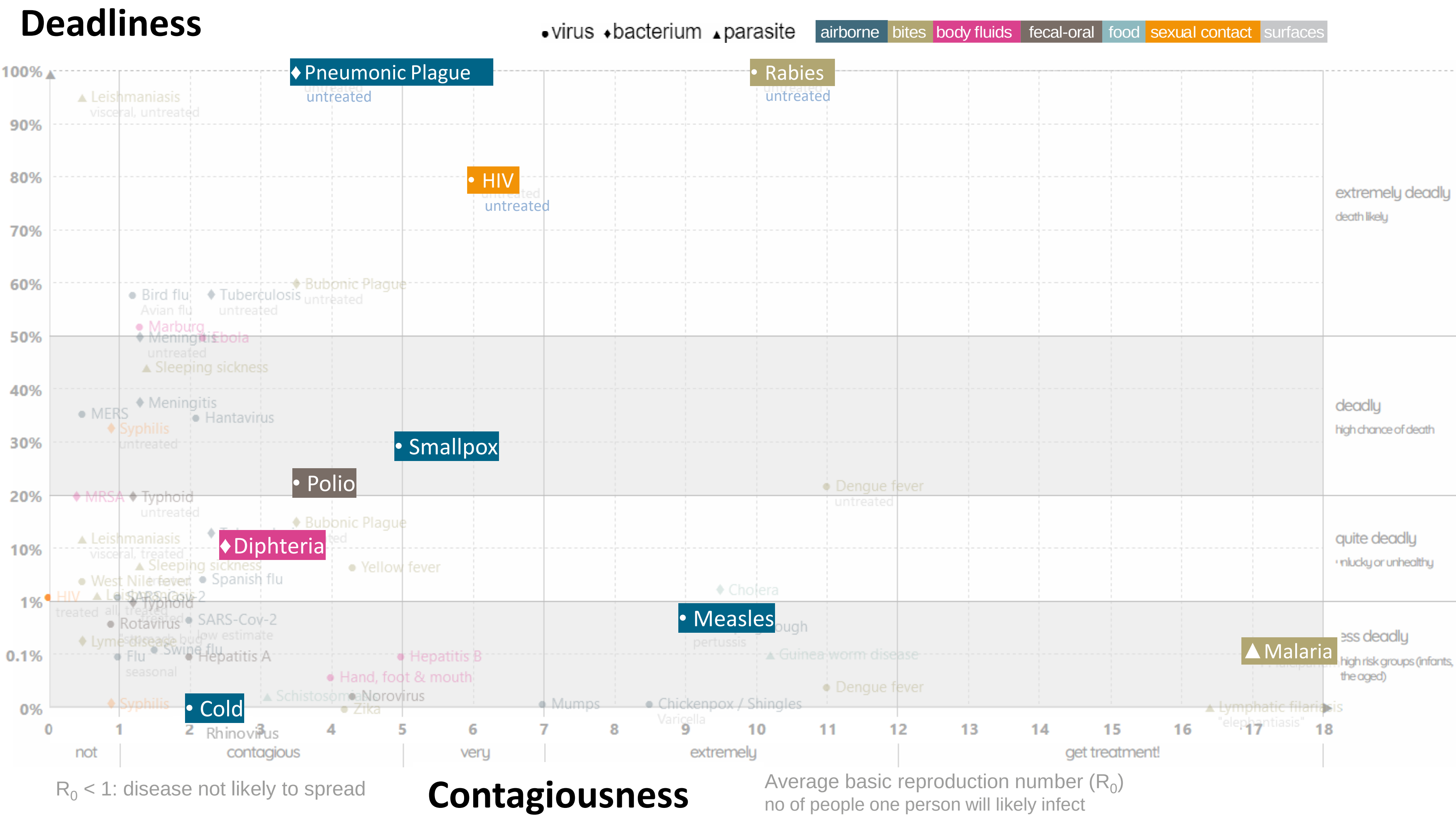


Sources: Centers for Disease Control, World Health Org, CIDRAP, studies

www.informationisbeautiful.net/visualizations/the-microbescope-infectious-diseases-in-context/

Infectious Diseases in Context

David McCandless et al. (2014)

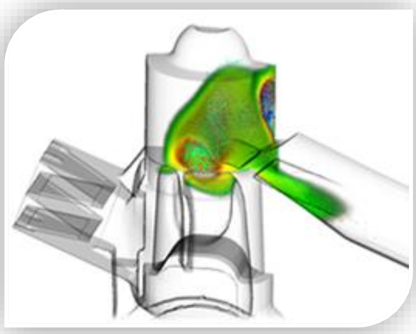


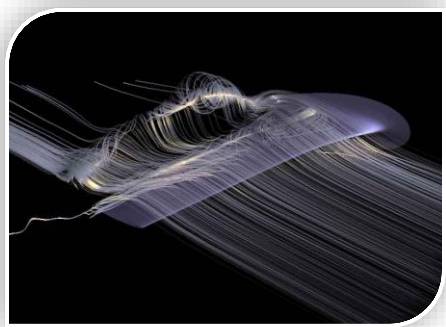
Sources: Centers for Disease Control, World Health Org, CIDRAP, studies

www.informationisbeautiful.net/visualizations/the-microbescope-infectious-diseases-in-context/

Visualization basics

Visualization pipeline & scenarios

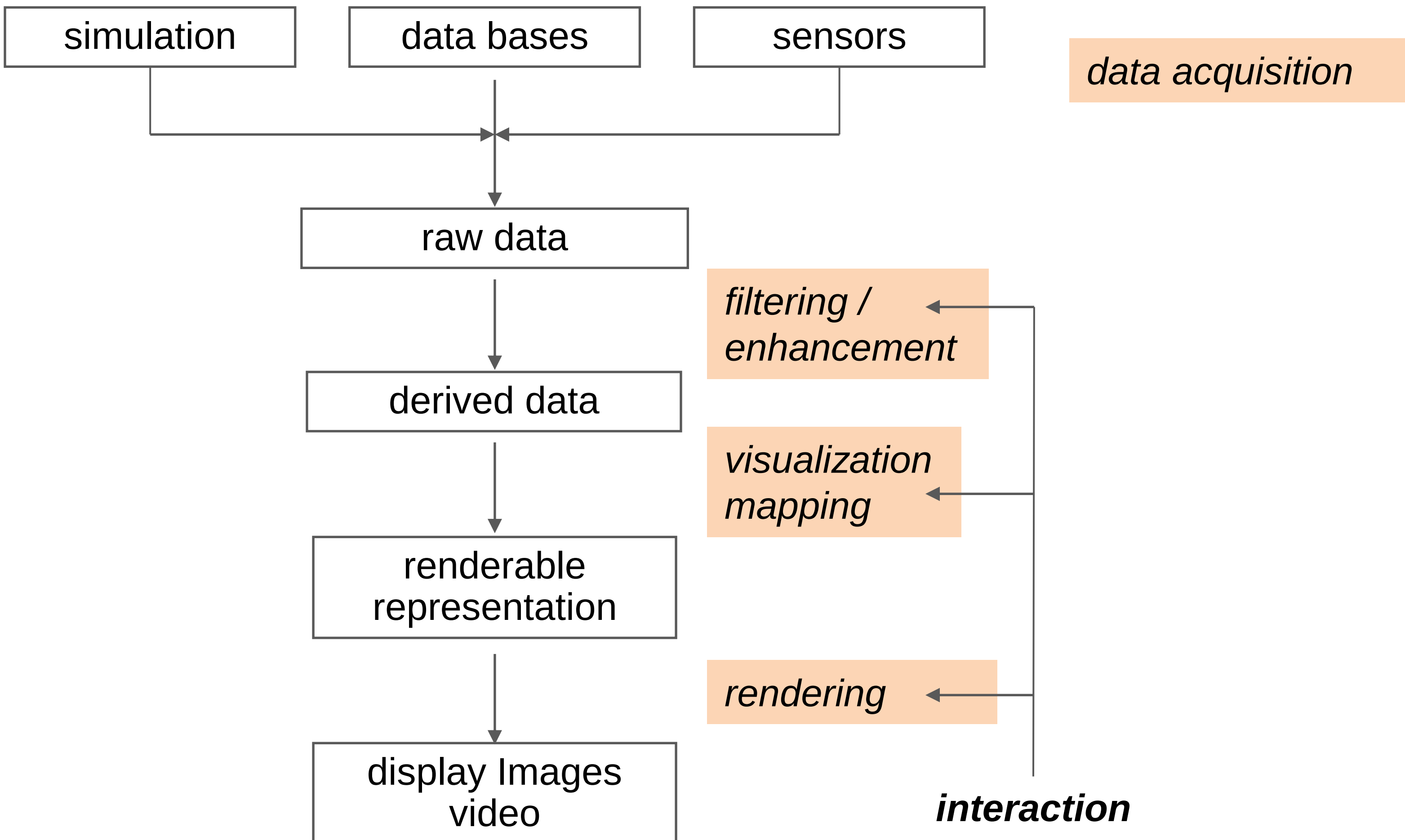




Visualization pipeline

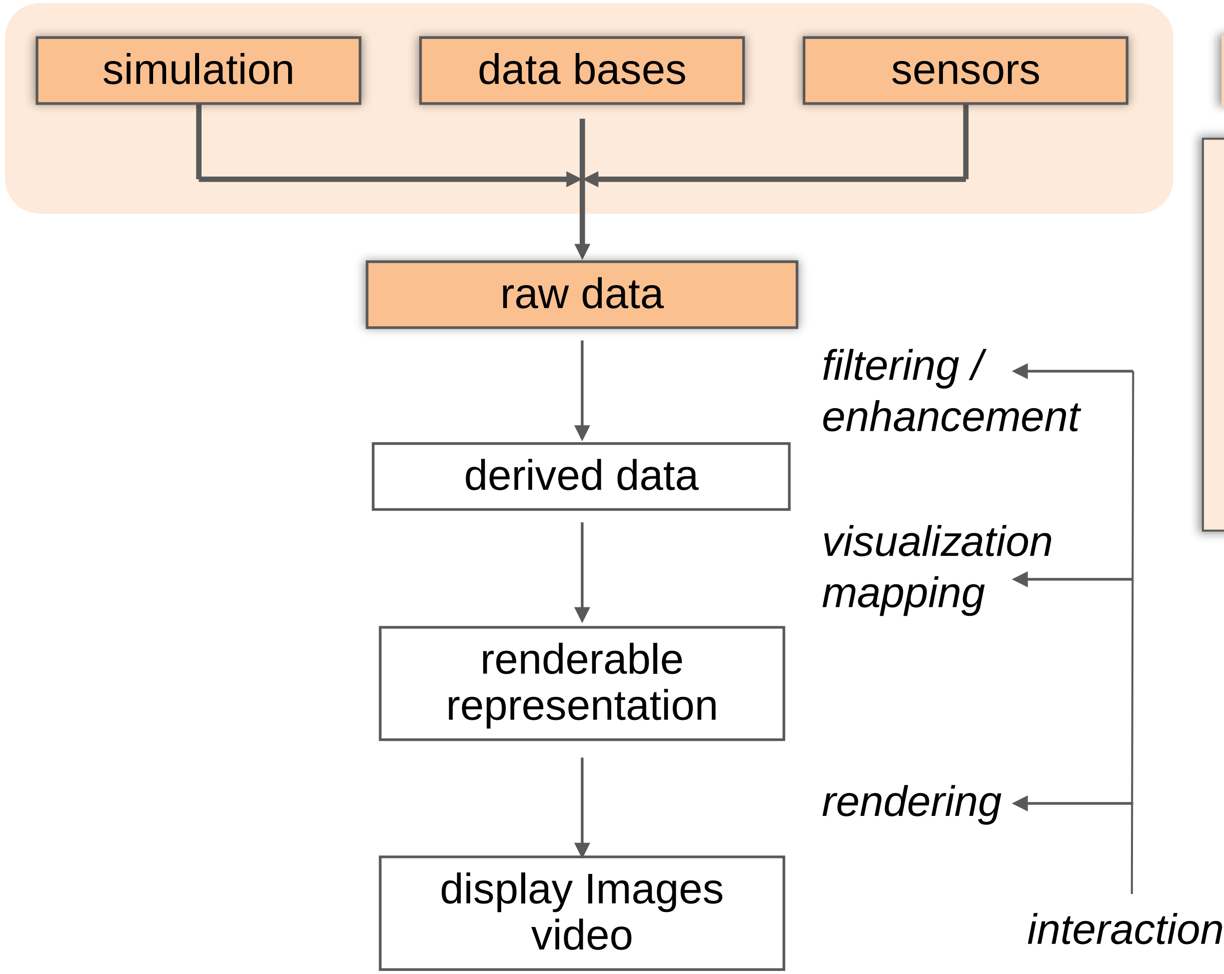
SIEMENS
Ingenuity for life

- A general model of visualization process with 4 stages
- User can interact with the process at each stage



Visualization pipeline

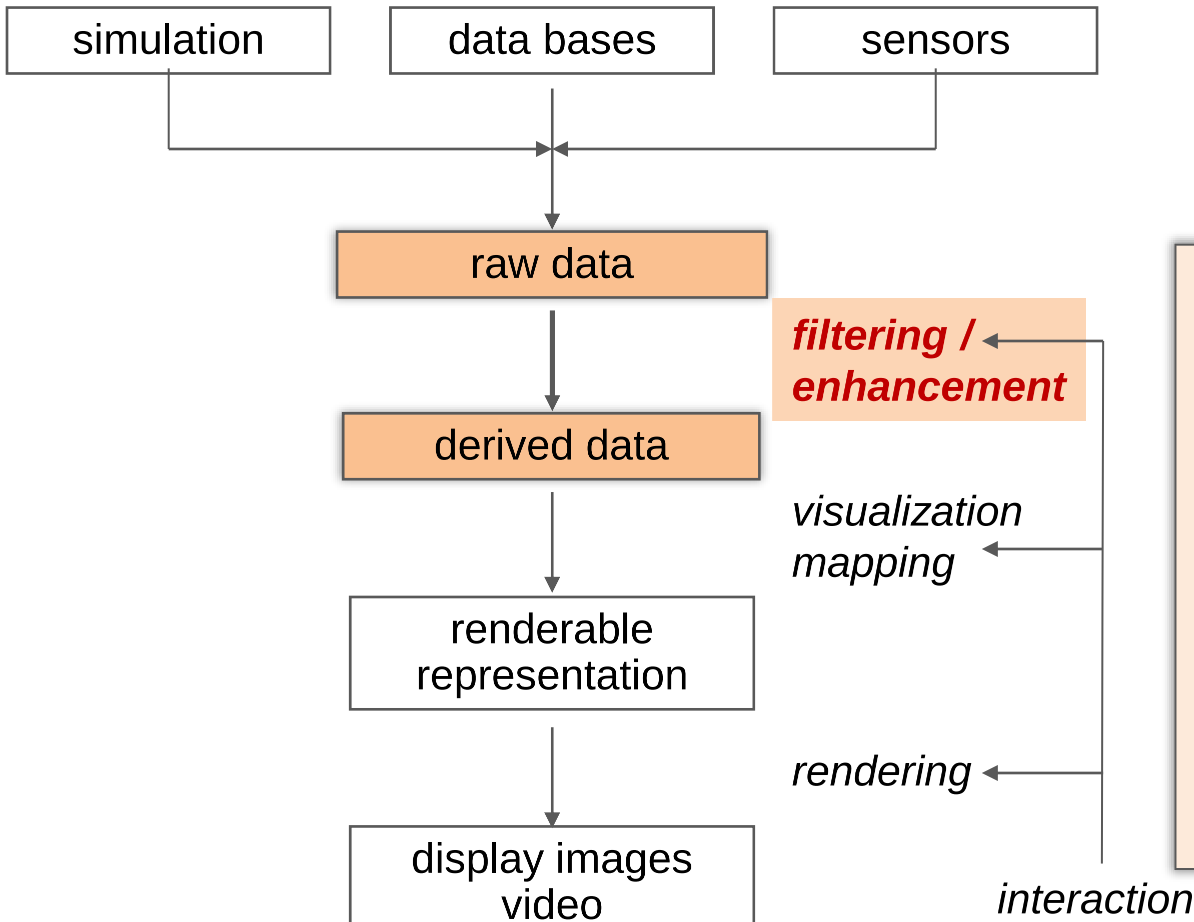
SIEMENS
Ingenuity for life



data acquisition

Examples
- global climate simulation
- projected densities from CT
- images of particles in fluid
- concentration of contrast agent in blood flow
- weather radar data
- sonar data, …

Visualization pipeline

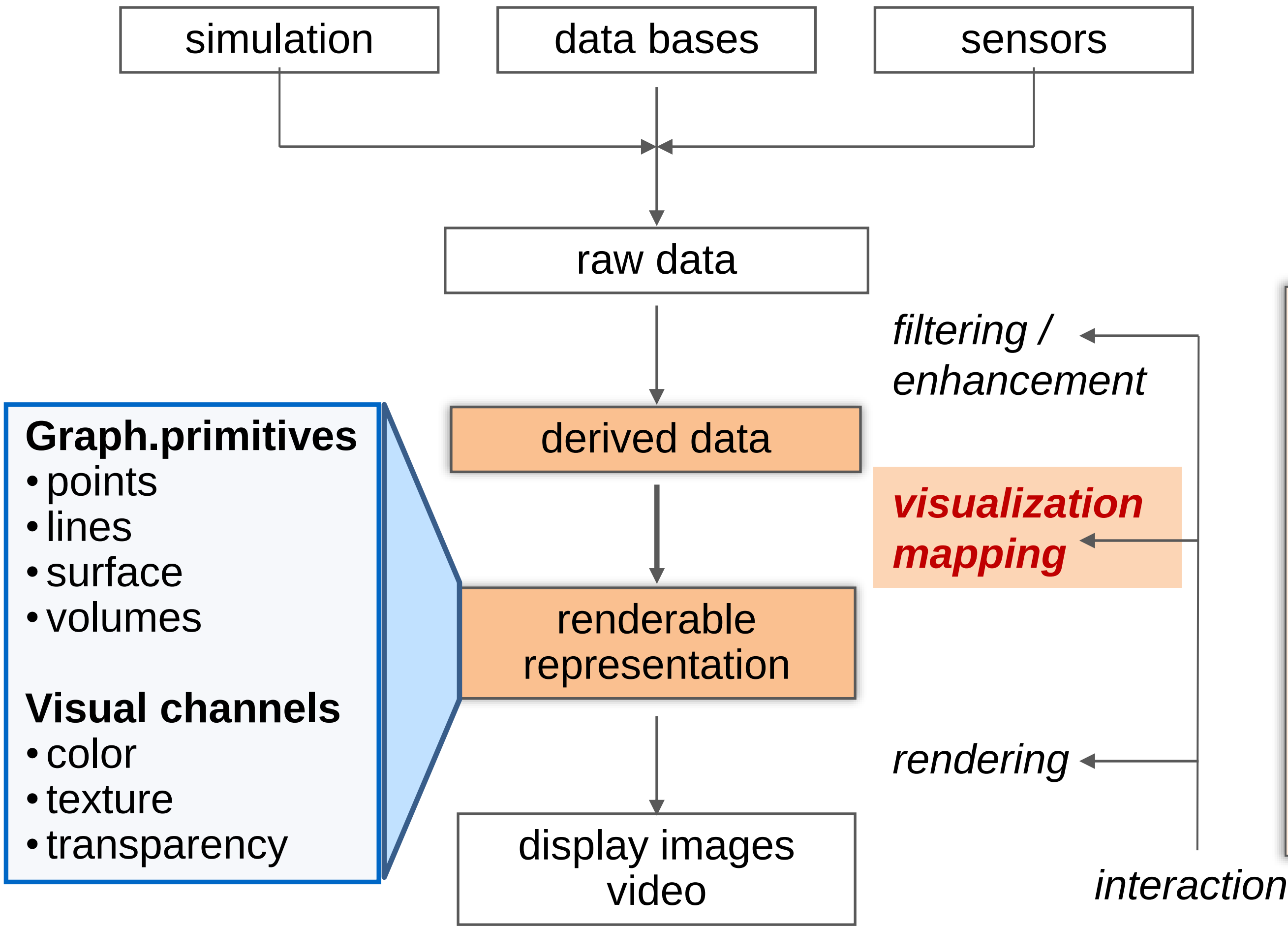


Process to obtain useful data (e.g., 3D volume)

- data format conversion
- co-registration of data sets
- resampling to grid
- interpolation/approximation of missing values
- data reduction (remove nonrelevant data)
- clipping/cleaning/denoising
- characteristic properties (gradients, curvature, extrema, segmentation)

Visualization pipeline

SIEMENS
Ingenuity for life



Derived data
→ graphical primitives

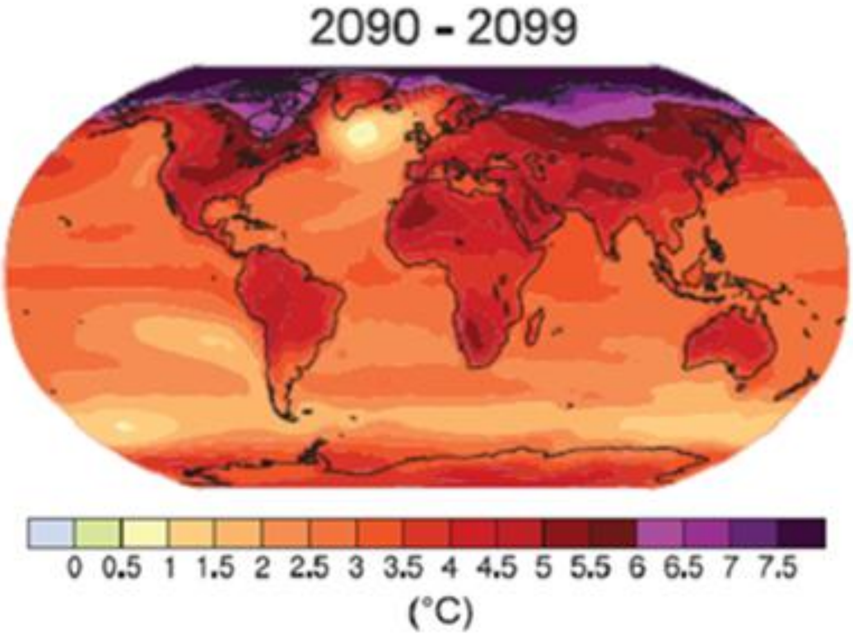
- scalar field → isosurface
- 2D field → height field
- vector field → vectors
- tensor field → glyphs
- 3D field → volume (C,α)

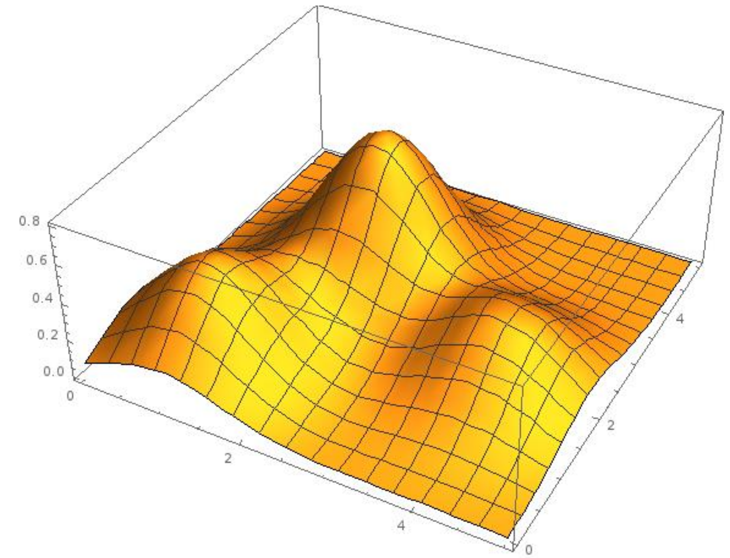
Decide

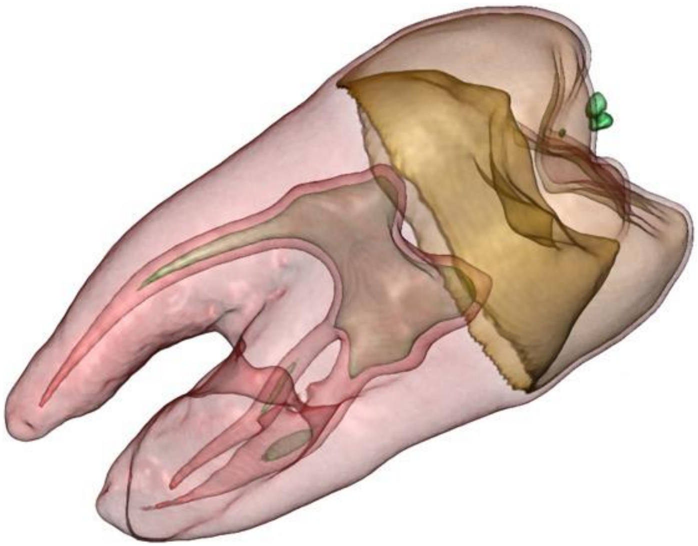
- Which parts of the data are shown?
- How to represent them?

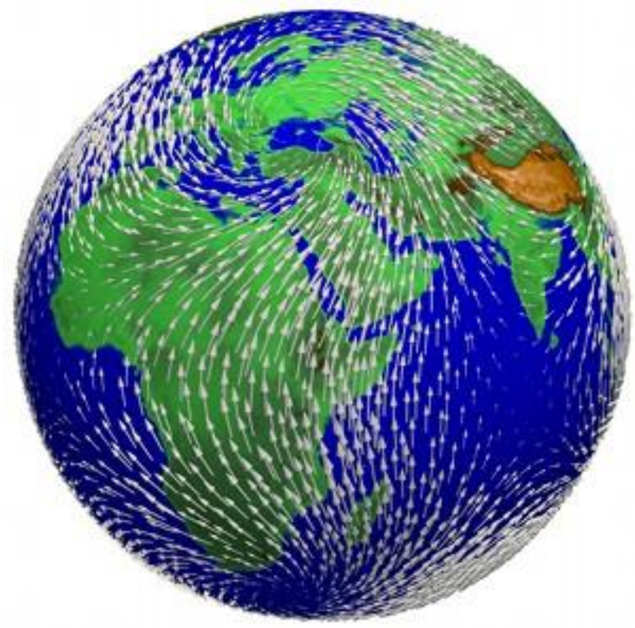
Visualization Mappings – Examples

SIEMENS
Ingenuity for life



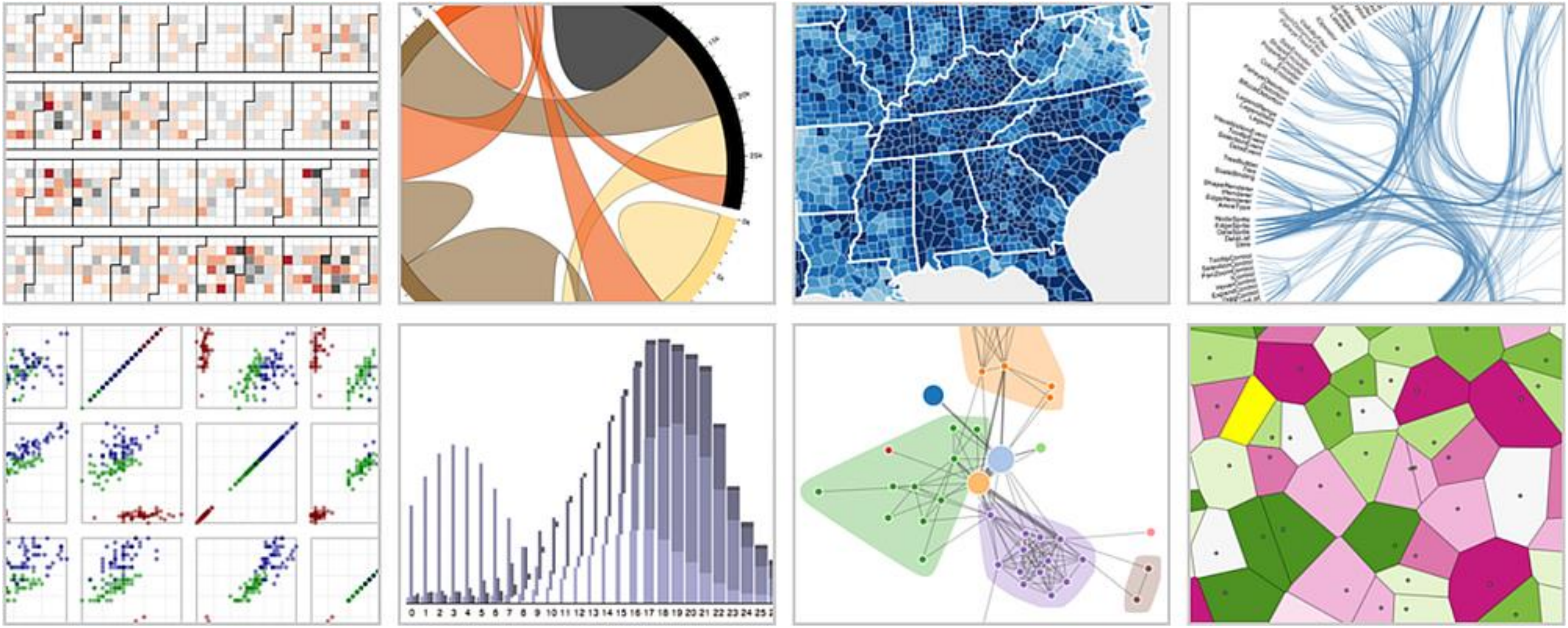






Visualization Mappings – Examples

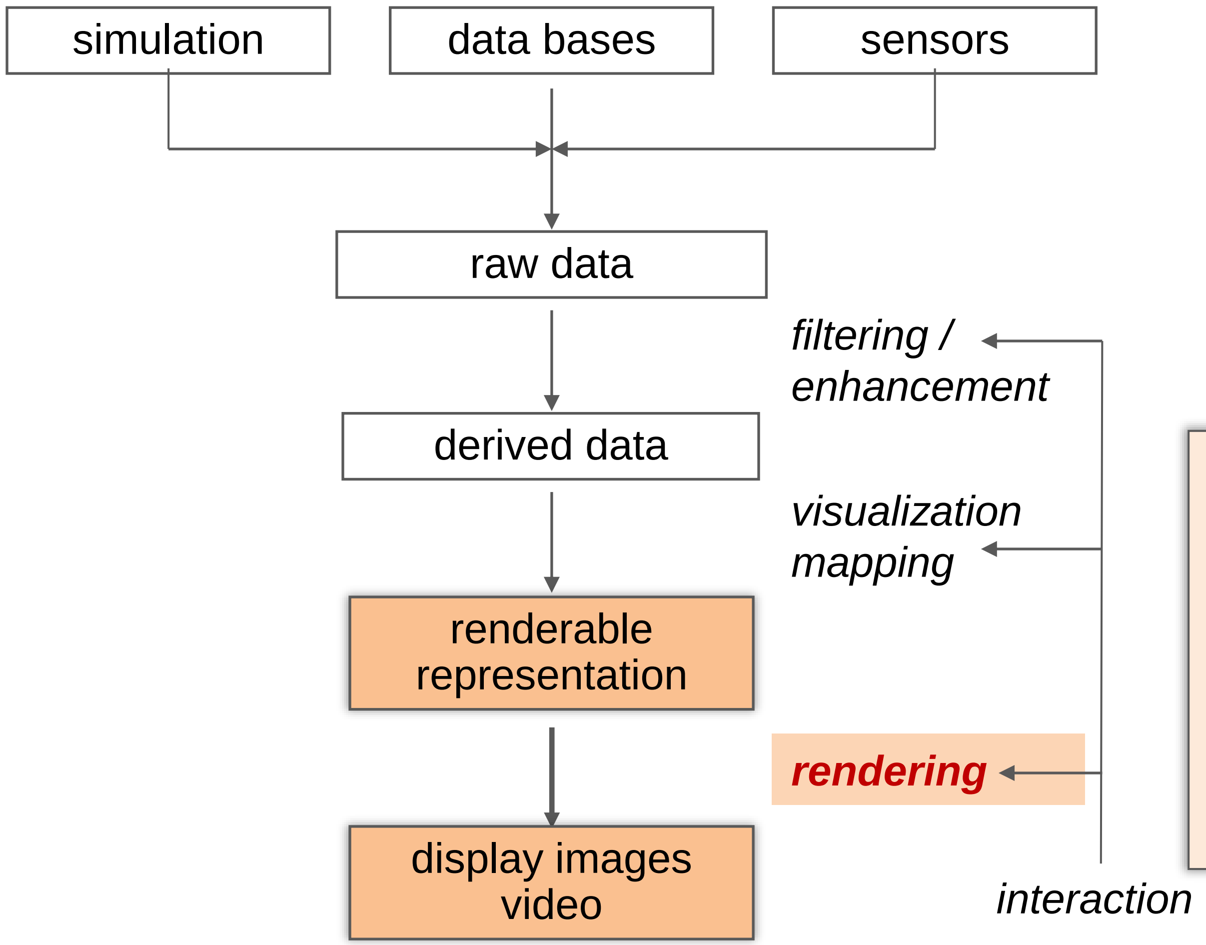
SIEMENS
Ingenuity for life



Interactive visualizations built with D3. From left to right: calendar view, chord diagram, choro-pleth map, hierarchical edge bundling, scatterplot matrix, grouped & stacked bars, force-directed graph clusters, Voronoi tessellation

https://d3js.org/

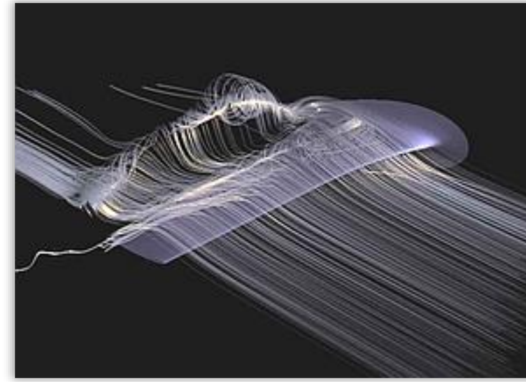
Visualization pipeline

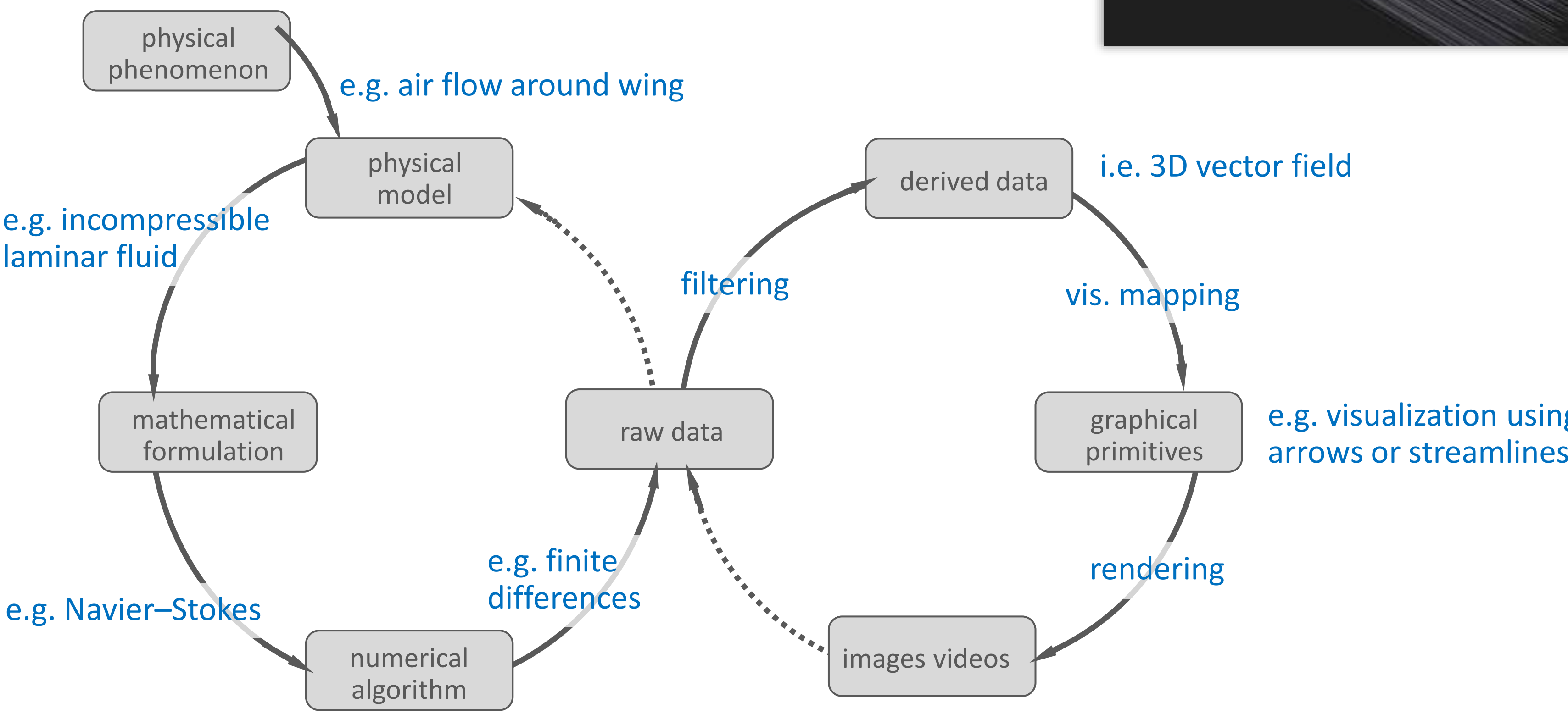


Generate 2D image/video
- viewpoint specification
- visibility calculation
- *lighting/shading:* evaluate illumination model
- *composition:* determine contribution to final image
- compute values at sampling points

Visualization pipeline

- Example: simulation of the flow around an airplane wing



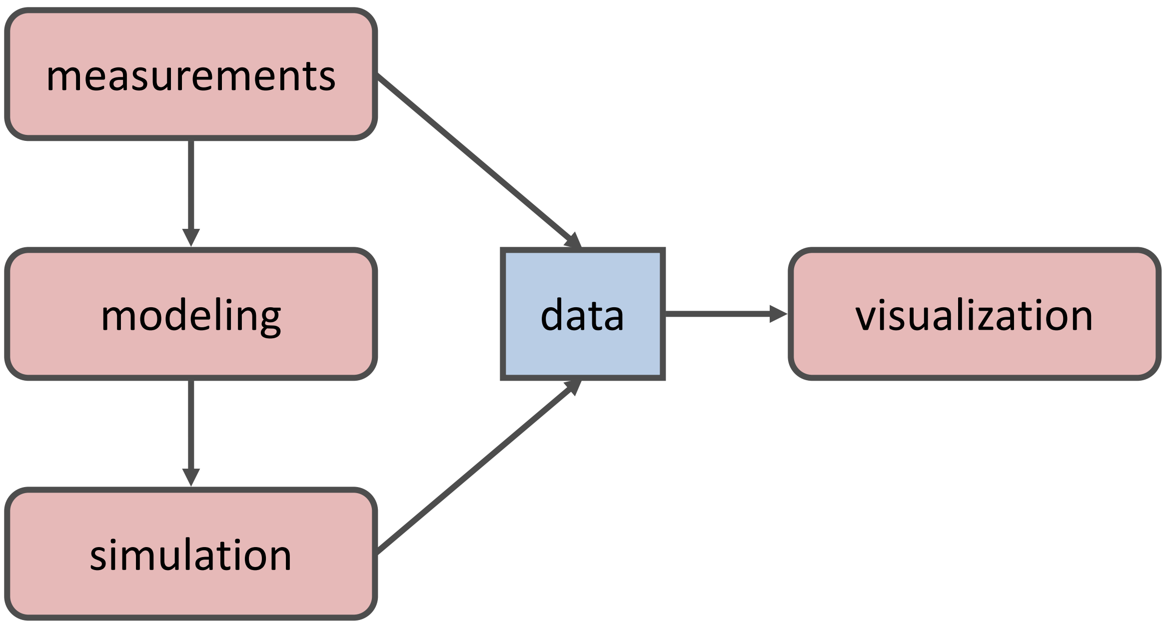


Data, Visualization, Interaction

- Data generation/acquisition
  - Measuring, simulation, modeling
  - Can take very long (measuring, simulation)
  - Can be very expensive (simulation, modeling)
- Visualization (rest of vis. pipeline)
  - Data filtering/enhancement, vis.mapping, rendering
  - Depends on computer/implementation: fast or slow
- Interaction (user input)
  - How can users change parameters, viewpoint, etc.

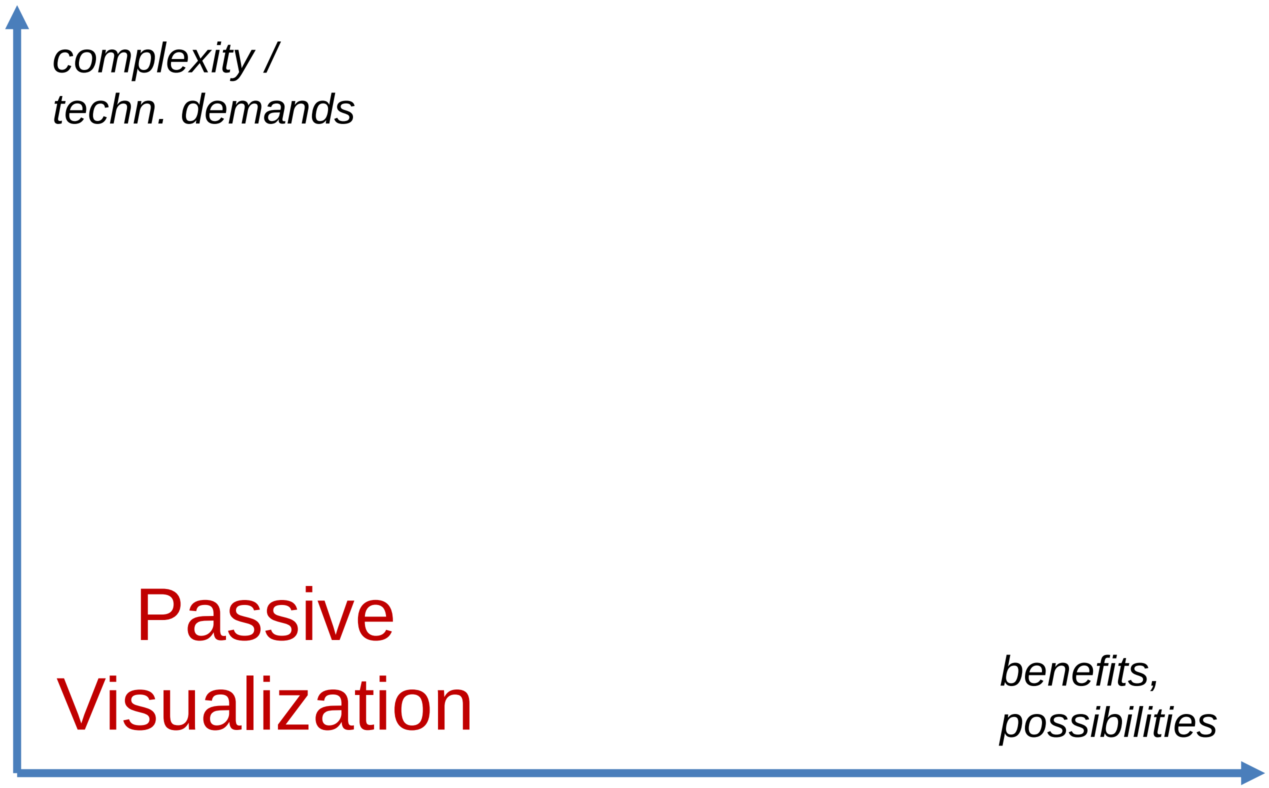
Visualization Scenarios

- How closely is visualization connected to data generation?
- Scenario: Passive visualization



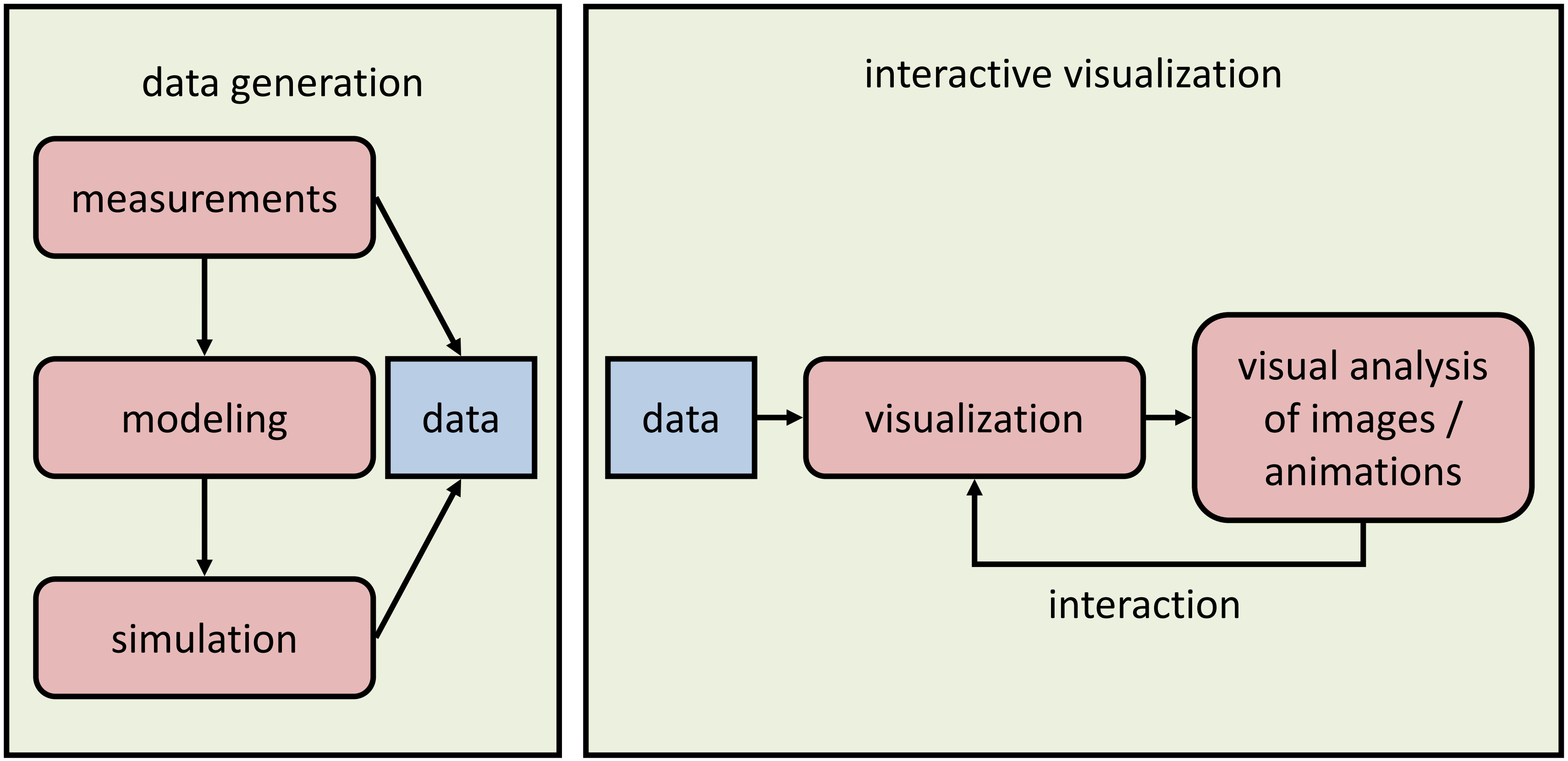
Visualization Scenarios

- How closely is visualization connected to data generation?



Visualization Scenarios

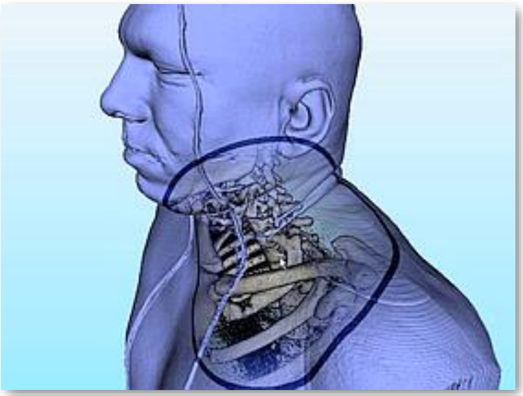
- Scenario: Interactive visualization

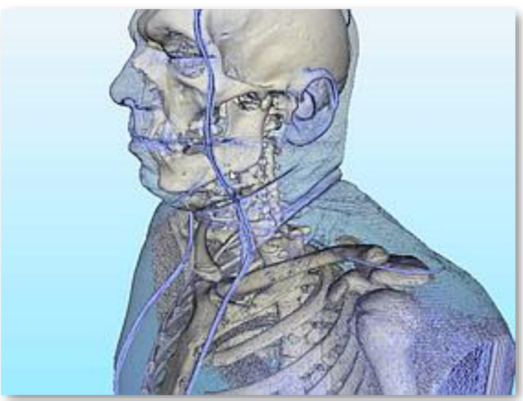


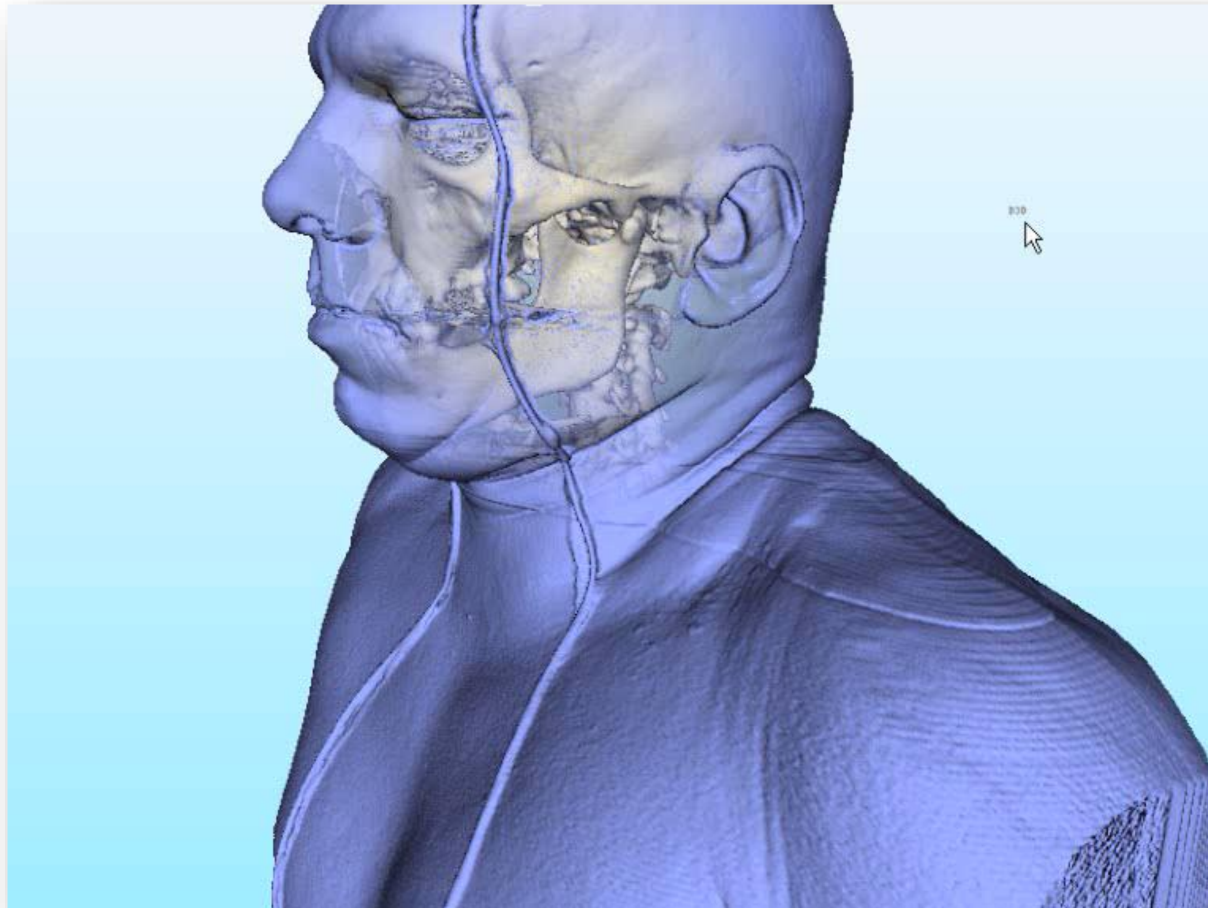
Example: Interactive Visualization

SIEMENS
Ingenuity for life

- Exploration of volumetric dataset



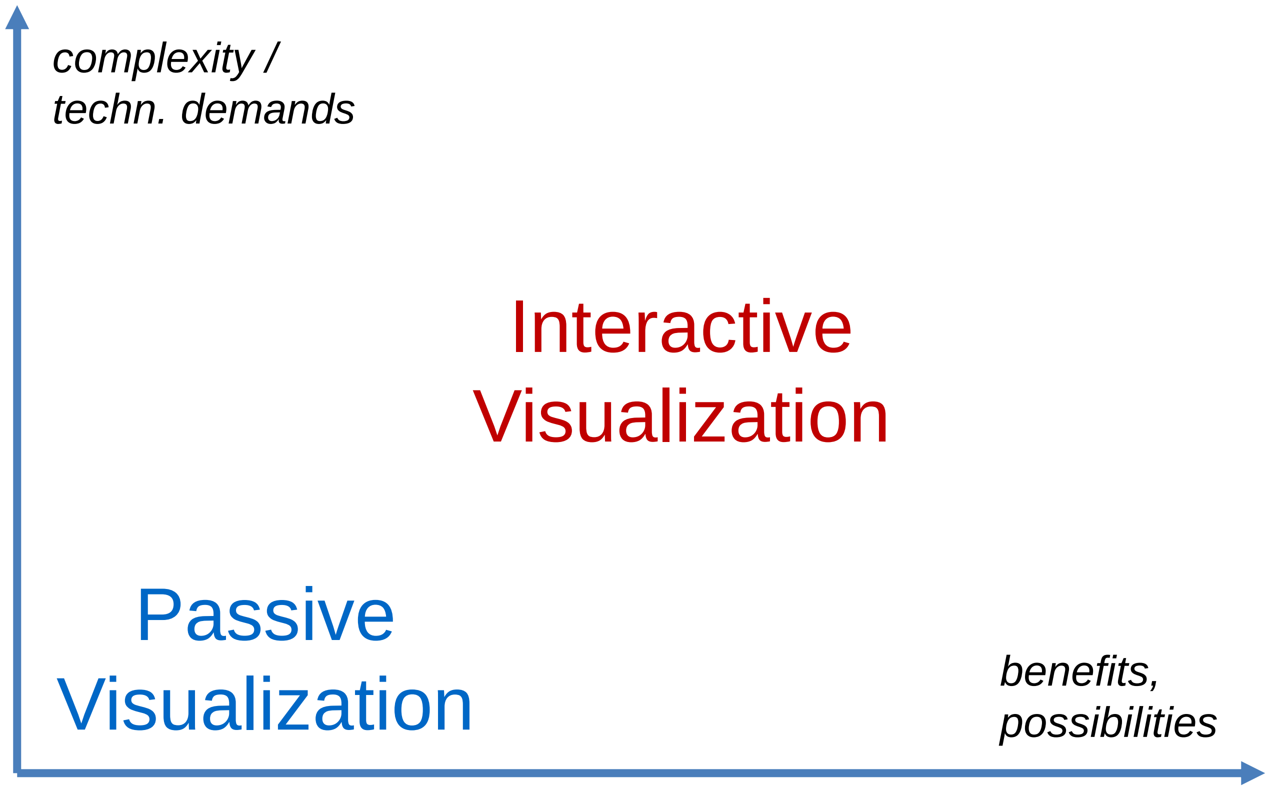




Visualization Scenarios

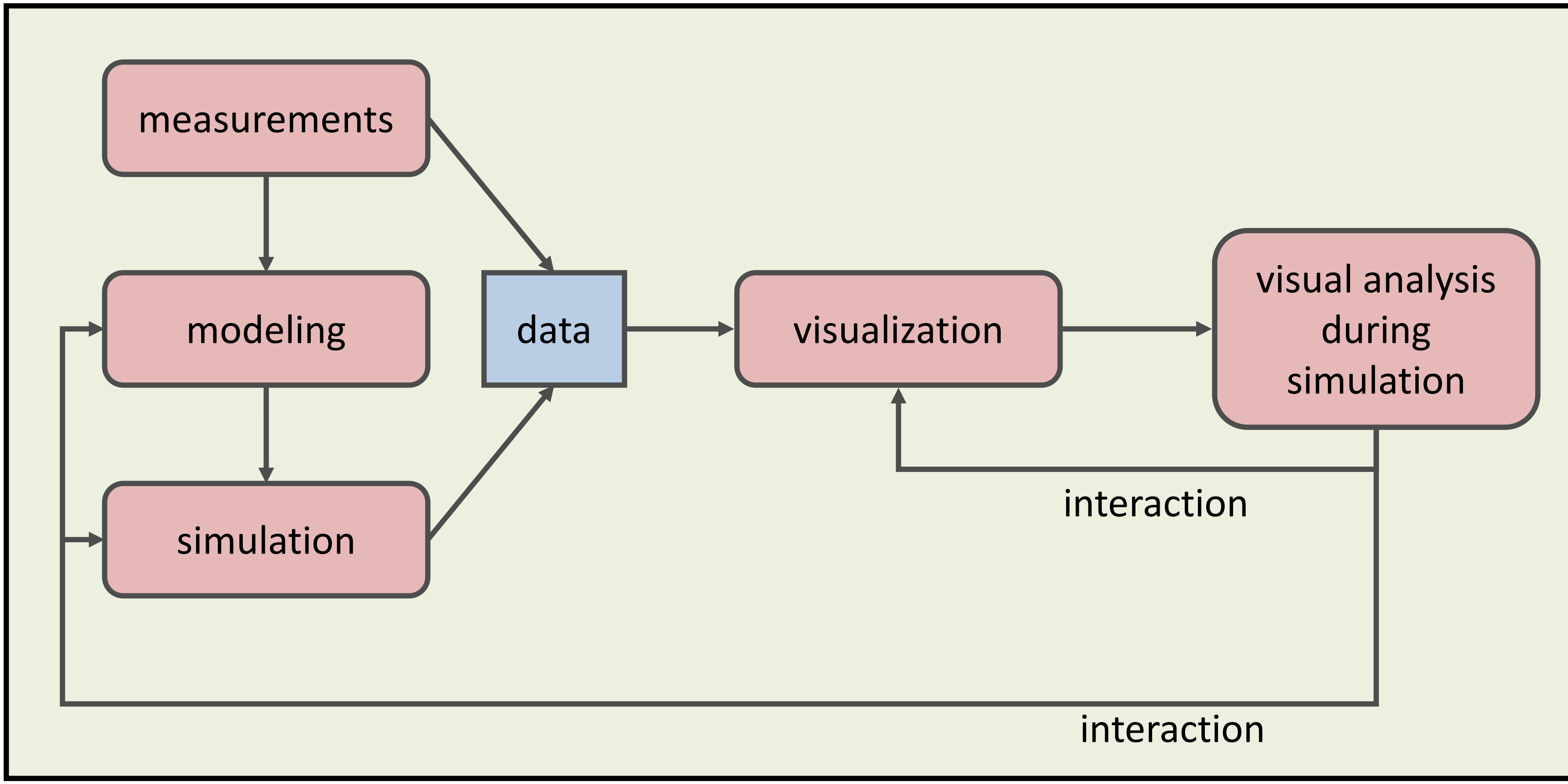
SIEMENS
Ingenuity for life

- How closely is visualization connected to data generation?



Visualization Scenarios

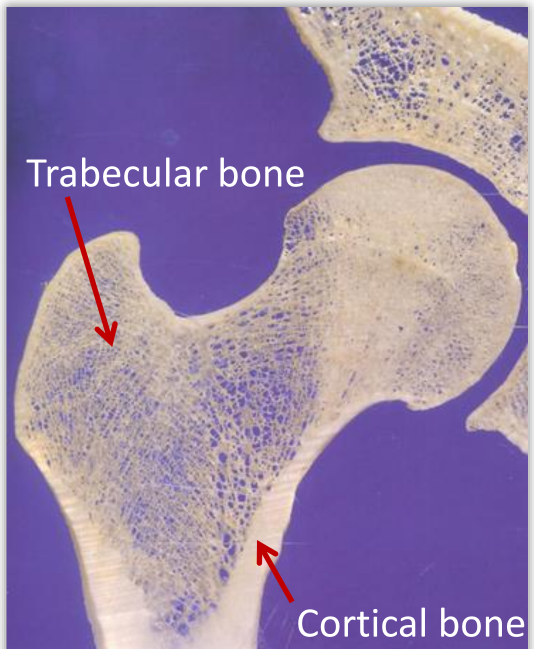
- Scenario: Computational / visual steering



Example: Visual Steering

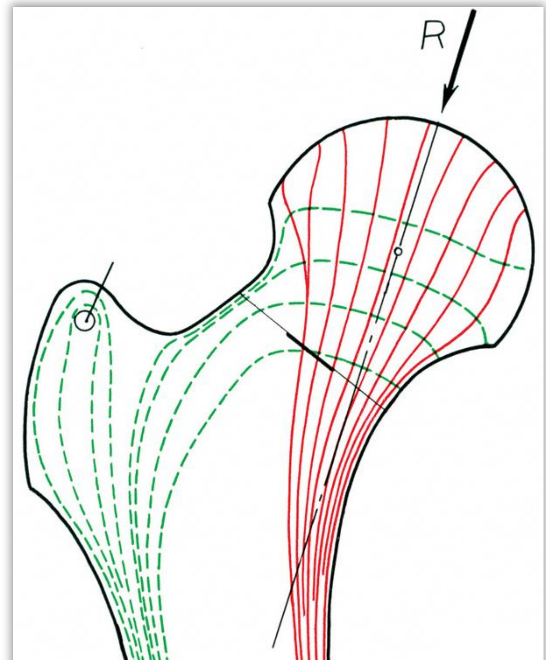
SIEMENS
Ingenuity for life

- Two types of bone tissue
  - Trabeculae are aligned along the principal stress directions (Wolff's law)



Cross section

X-ray

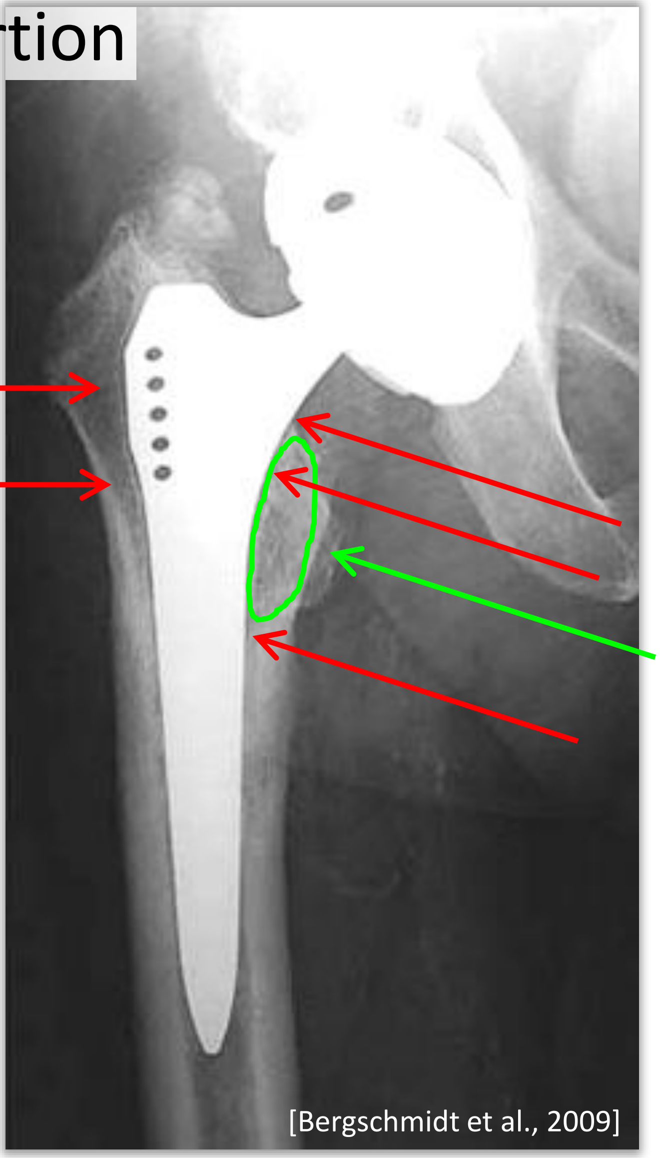


Principal stress directions [Pauwels, 1973]

Example: Visual Steering

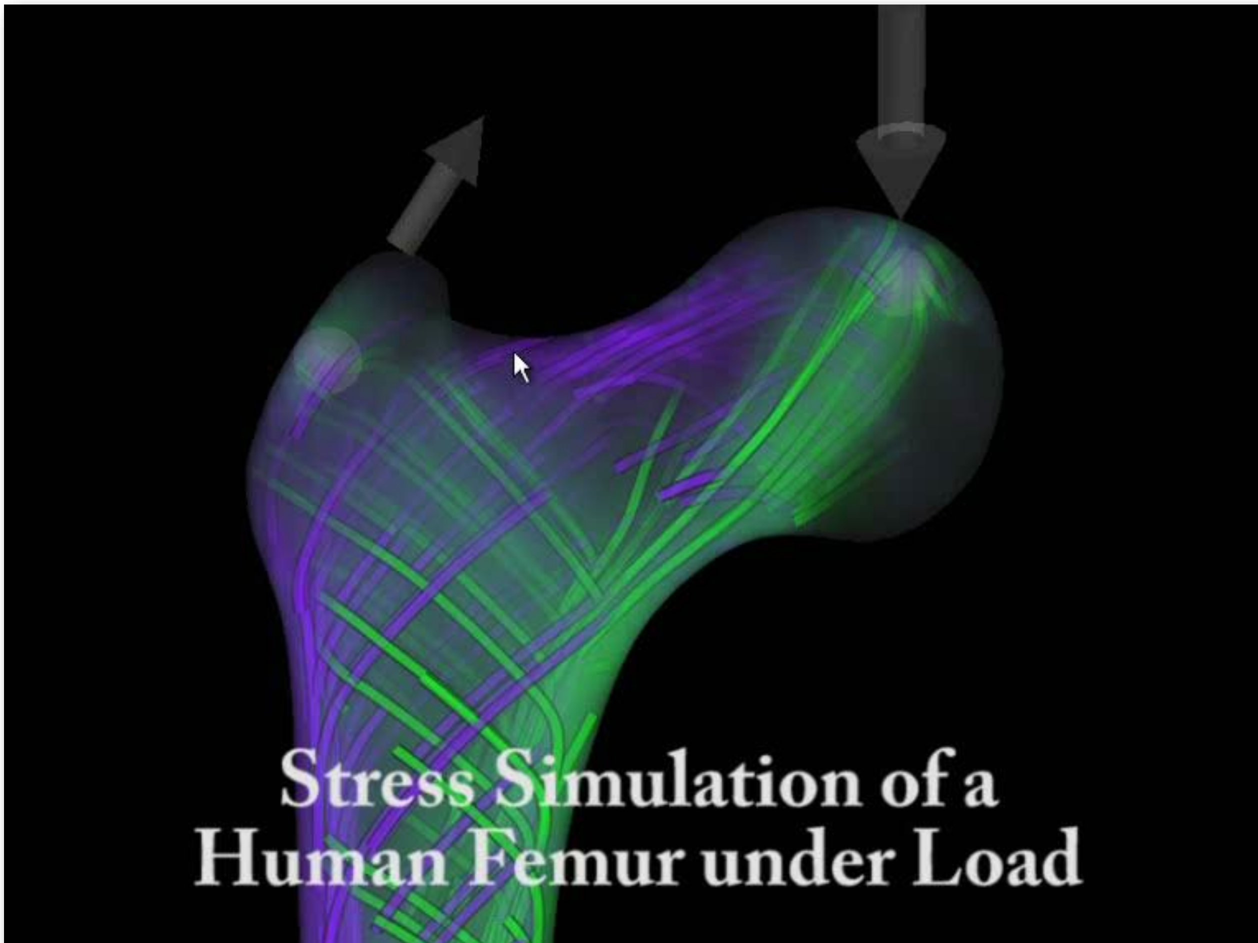
SIEMENS
Ingenuity for life

- Stress Shielding due to implant insertion
  - Stresses are bypassed by the implant
  - Adaptive bone remodeling
  - Bone degeneration, fracture, loosening of the implant
  - An optimal implant should replicate the preoperative stress state



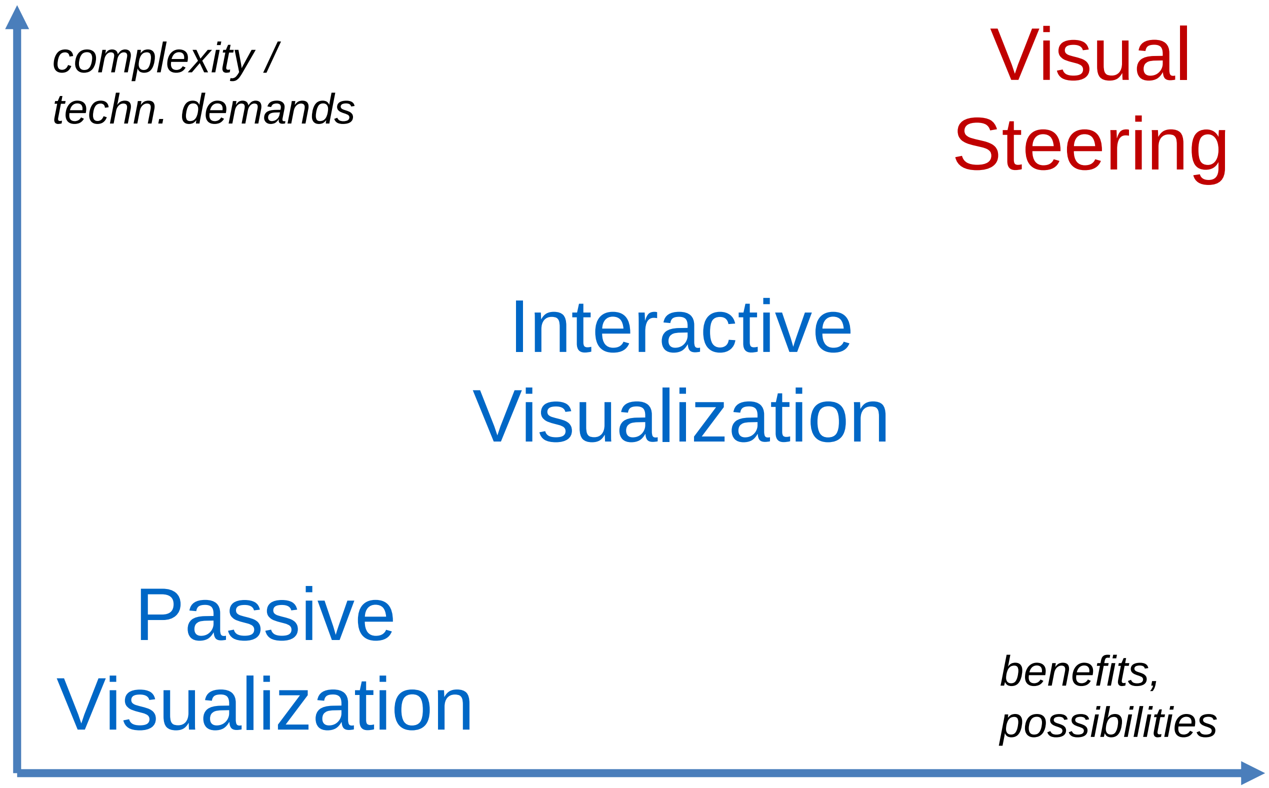
Example: Visual Steering

- Preoperative surgery planning



Visualization Scenarios

- How closely is visualization connected to data generation?



Acknowledgements

- M. Eduard Gröller
- Helwig Hauser
- Torsten Möller
- Tamara Munzner
- many more

Contact information

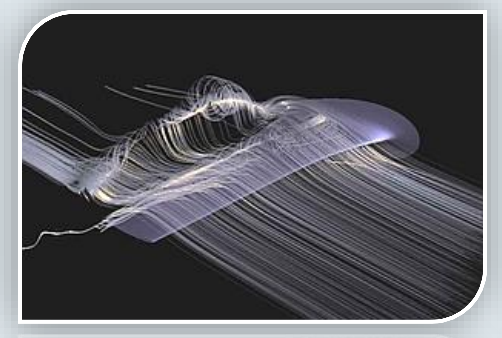
SIEMENS
Ingenuity for life

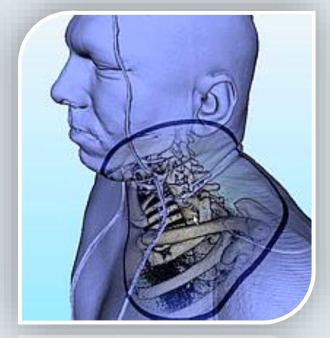
Dr. Johannes Kehrer

Siemens AG
Technology
T DAI HCA-DE
Otto-Hahn-Ring 6
81739 München, Deutschland

E-mail:
kehrer.johannes@siemens.com

Internet
siemens.com/innovation





Visual Data Analytics
Data Representation

Dr. Johannes Kehrer – Siemens Technology, Munich

Disclaimer

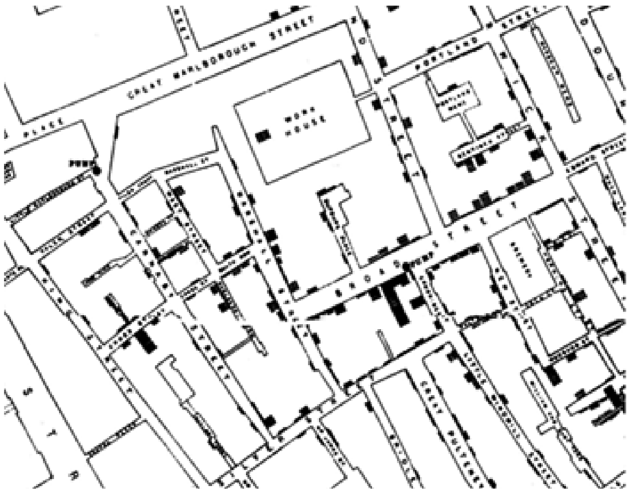
These lecture slides and materials are for personal use by students and are intended for educational purposes only. Distribution or reproduction of the material is prohibited.

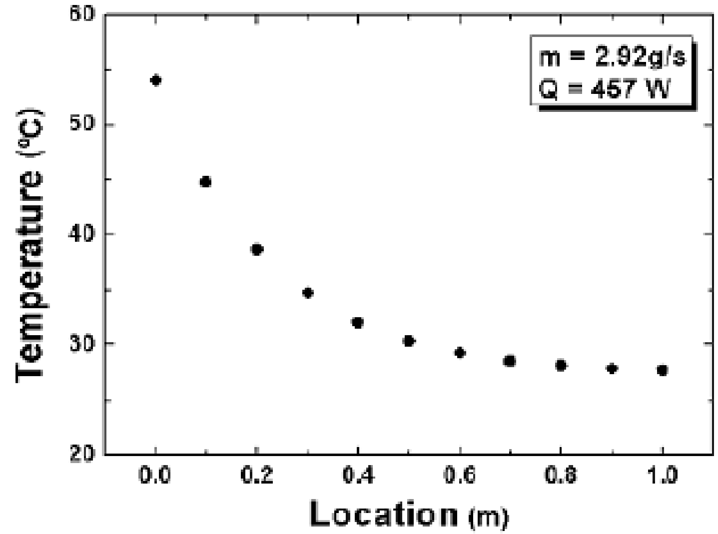
The views and opinions expressed by any individual during the course of these lecturers are their own and do not necessarily represent the views, opinions, or positions of Siemens or the Technical University of Munich.

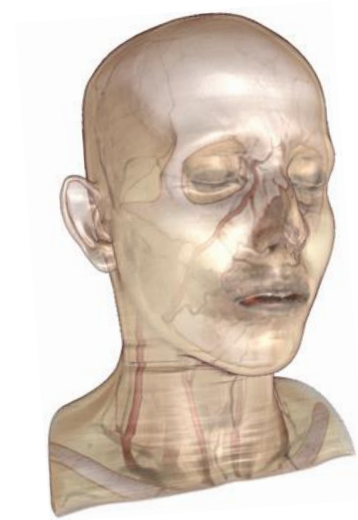
Although reasonable efforts have been made to ensure the information in these lecture slides is current and accurate, we do not assume any responsibility for the content of the posted material.

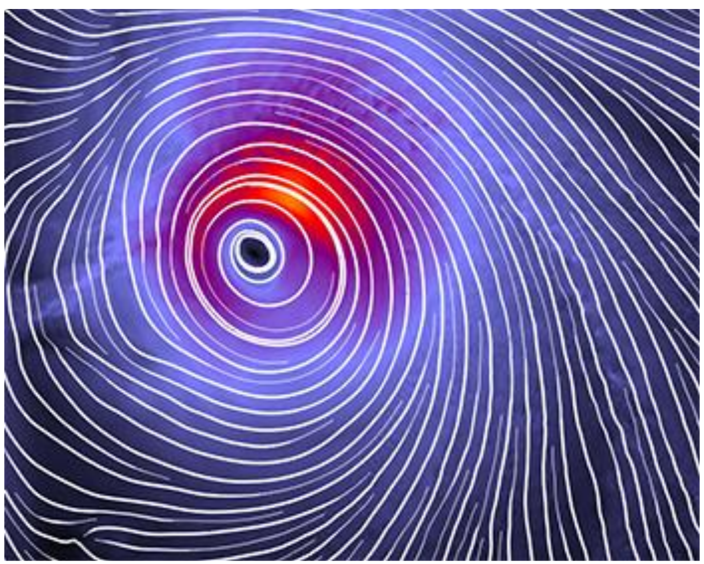
These slides also contain links to websites of third parties. As the content of these websites is not under our control, we cannot assume any liability for such external content.

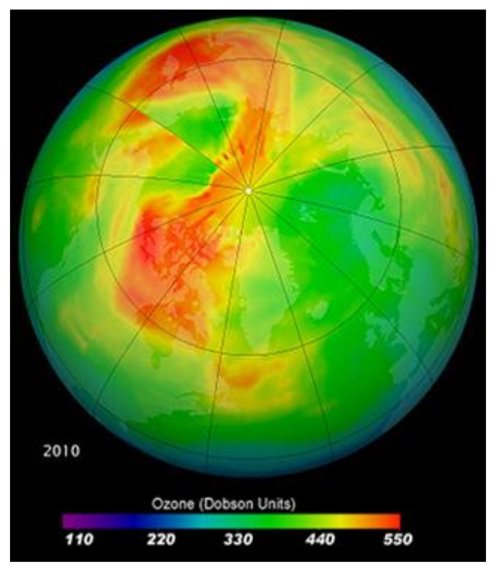
Visualization Examples

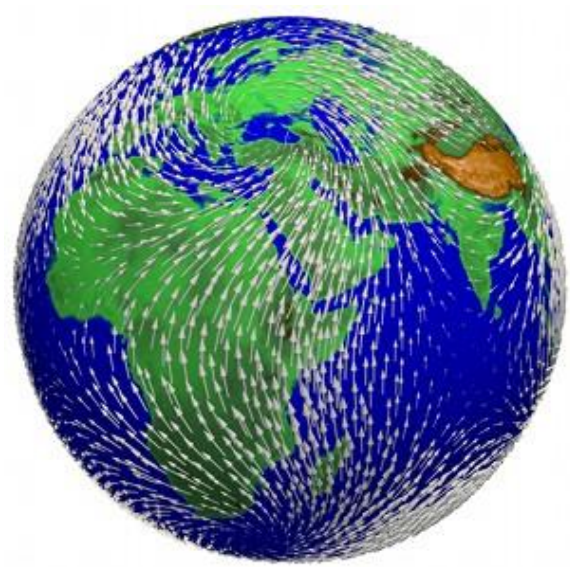












Data representation

- Classification of visualization techniques according to
  - Dimension of the domain of the problem (independent variables)
  - Type and dimension of the data to be visualized (dependent variables)

Data representation

SIEMENS
Ingenuity for life

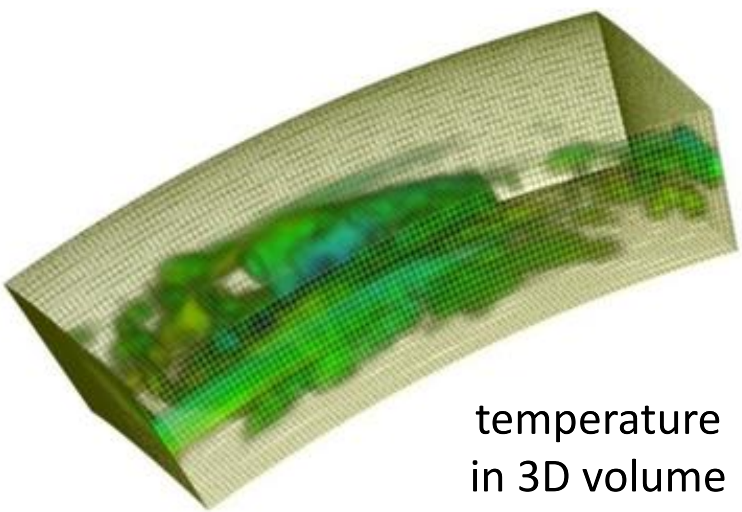
R^n $\times$ R^m

domain **x**

data values $f_i(\mathbf{x})$

independent variables
(e.g., 2D/3D space, time)

dependent variables
(e.g., temperature, density values, velocity vectors)

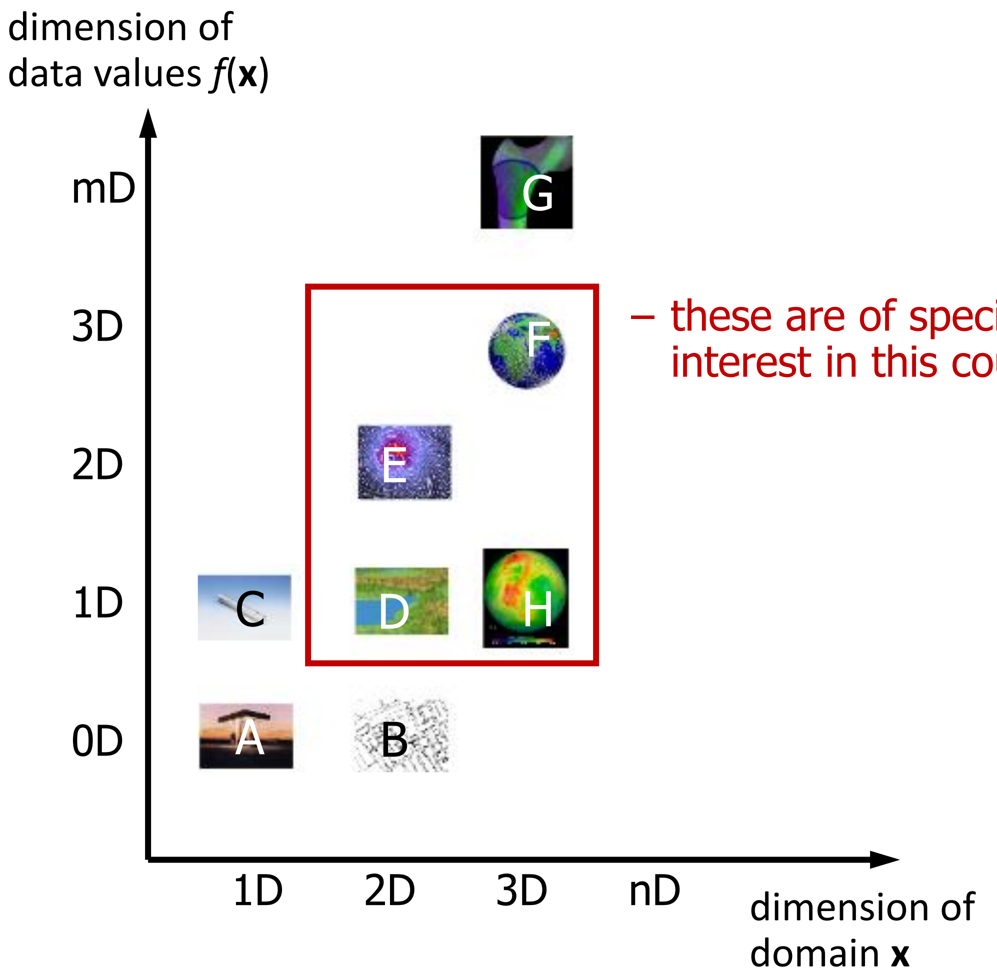


temperature
in 3D volume

scientific data $\subseteq R^{n+m}$

Data representation

SIEMENS
Ingenuity for life



Examples:

A: gas stations along a road
B: map of cholera in London
C: temperature along a rod
D: height field of a continent
E: 2D air flow
F: 3D air flow in the atmosphere
G: stress tensor in a joint
H: ozone concentration in the atmosphere

Data representation

SIEMENS
Ingenuity for life

- Multidimensional data

Car name	MPG	Cylinders	Horse-power	Weight	Accel-eration	Model year	Origin
chevrolet woody	24,5	4	98	2164	22,1	1976	US
vw rabbit	29,0	4	90	1937	14,2	1976	Europe
honda civic	33,0	4	91	1795	17,4	1976	Japan
dodge aspen se	20,0	6	225	3651	17,7	1976	US
ford granada ghia	18,0	6	250	3574	21	1976	US
pontiac ventura sj	18,5	6	250	3645	16,2	1976	US
amc pacer d/l	17,5	6	258	3193	17,8	1976	US
volkswagen rabbit	29,5	4	97	1825	12,2	1976	Europe
datsun b-210	32,0	4	85	1990	17	1976	Japan
toyota corolla	28,0	4	97	2155	16,4	1976	Japan
ford pinto	26,5	4	140	2565	13,6	1976	US
volvo 245	20,0	4	130	3150	15,7	1976	Europe
plymouth volare premier v8	13,0	8	318	3940	13,2	1976	US
peugeot 504	19,0	4	120	3270	21,9	1976	Europe
toyota mark ii	19,0	6	156	2930	15,5	1976	Japan
mercedes-benz 280s	16,5	6	168	3820	16,7	1976	Europe
cadillac seville	16,5	8	350	4380	12,1	1976	US

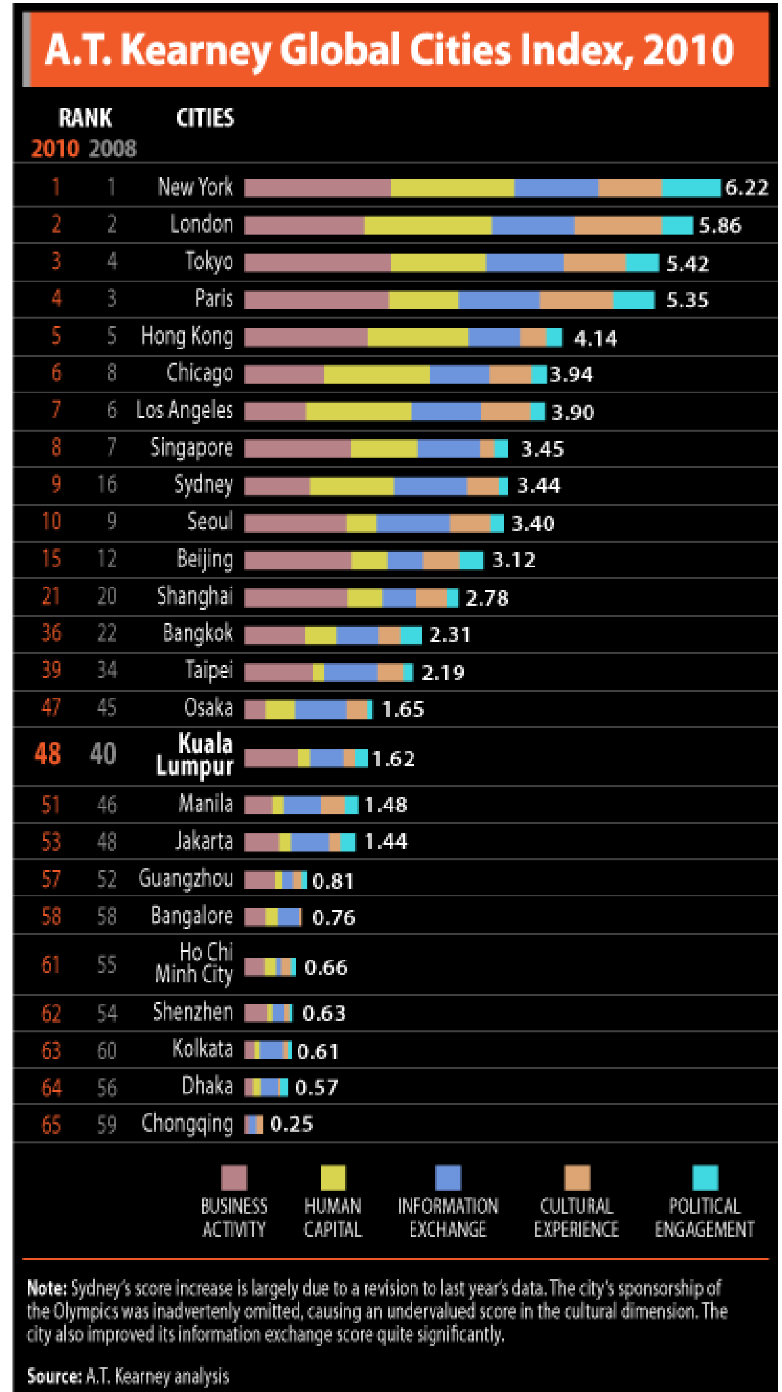
Attribute / Dimension

Record/Item

Data representation

- Multidimensional data

1	Vienna	108.6
2	**Zurich**	**108.0**
3	**Geneva**	**107.9**
4	Vancouver	107.4
4	Auckland	107.4
6	Dusseldorf	107.2
7	Frankfurt	107.0
7	Munich	107.0
9	**Bern**	**106.5**
10	Sydney	106.3
11	Copenhagen	106.2
13	Amsterdam	105.7
15	Brussels	105.4
17	Berlin	105.0
19	Luxembourg	104.6
26	Dublin	103.6
28	Singapore	103.5
34	Paris	102.9
39	London	101.6
40	Tokyo	101.4
49	New York City	100.0



Data representation

- Characteristics of data values
  - Attribute types (quantitative vs. qualitative)
  - Domain (continuous vs. discrete)
  - Value range (includes precision of values)
  - Data type (categorical, scalar, vector, tensor data)
  - Dimension (number of components)
  - Error and uncertainty
  - (physical) interpretation

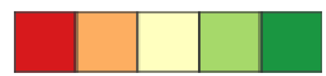
Attribute types

SIEMENS
Ingenuity for life

- Quantitative (numerical, measurable)
  - Objective data produced through a systematic process, not subject to interpretation (e.g., length, mass, temperature)
  - Metric scale – allows measure of distance
  - Continuous (real) or discrete (distinct & separate values)

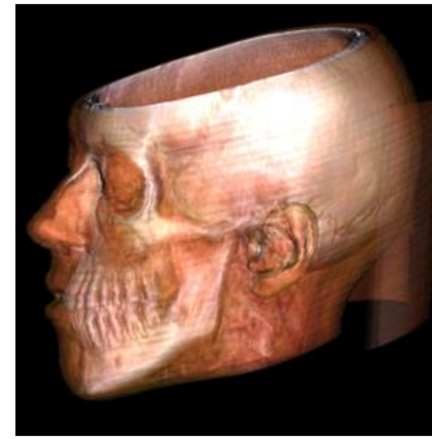
- Qualitative (categorical data, not measurable)
  - No metric scale; cannot be measured
  - Requires a subjective decision in order to be categorized
  - Discrete

Attribute types

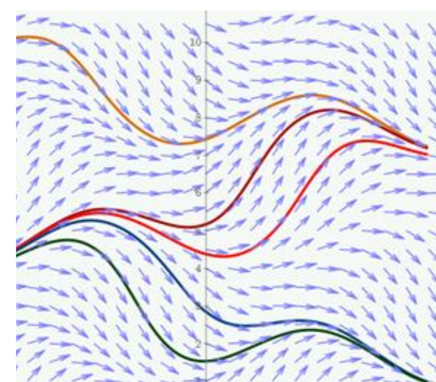
- Qualitative (categorical data)
  - Nominal  
    - No natural ordering or indication of values, only equivalence and membership ($=, \neq$)
    - Eye color (blue, green, brown)
  - Ordinal  
    - Logical order relation ($<, >$), but no relative size or degree of difference
    - Judgment of size (small, medium, large, etc.)
    - Attitudes (strongly disagree, disagree, neutral, agree, strongly agree)
    - Living quality (very high, high, medium, low, very low)

Attribute types

- Categorical data
  - Values from a fixed number of categories
- Scalar data
  - Given by a function $f(x_1,\dots,x_n)\colon \mathbb{R}^n \rightarrow \mathbb{R}$ with n independent variables x_i



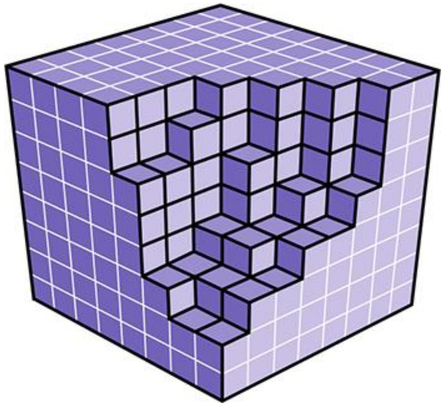
- Vector data
  - Represent direction and magnitude and given by an n-tupel $(f_1, \dots, f_n)$ with $f_k = f_k(x_1, \dots, x_n)$
  - 2D vector field where every sample represents a 2D vector (u, v) with $u = f(x, y)$ and $v = g(x, y)$



- Tensor data
  - A multi-dimensional matrix

Data representation

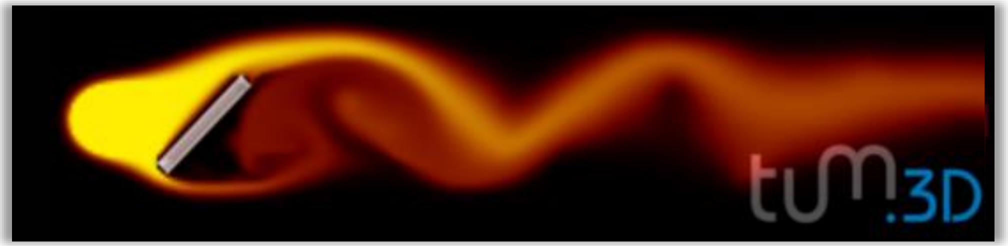
- In many cases, the visualization data represent a continuous real object
  - e.g., oscillating membrane, velocity field around a body, human organ or tissue, etc.
  - This object lives in an *n*-dimensional space - the domain
- Usually, the data is only given at a finite set of locations in space and/or time
  - Think of measurement devices & numerical simulations (note similarity to pixel images)



- We call this a discrete representation of a continuous object

Scientific Visualization

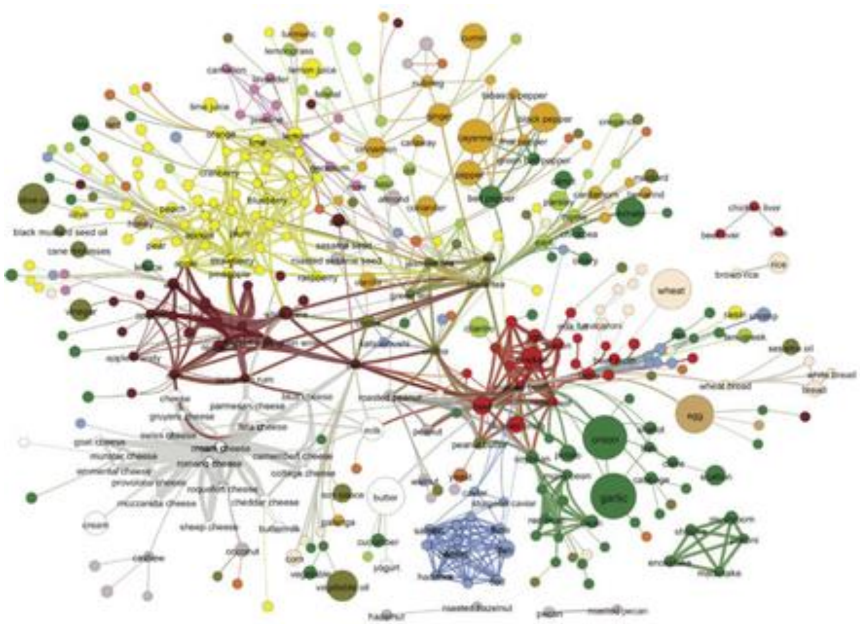
- Deals with the reconstruction of a continuous real object from a given discrete representation
- Data that has some physical or geometric correspondence



Information Visualization / Visual Analytics

SIEMENS
Ingenuity for life

- Deal with data that is discrete and more abstract
  - Does not have a physical or geometric correspondence
  - Symbolic, tabular, networked, graphs, textual information



Flavor compounds shared by culinary ingredients

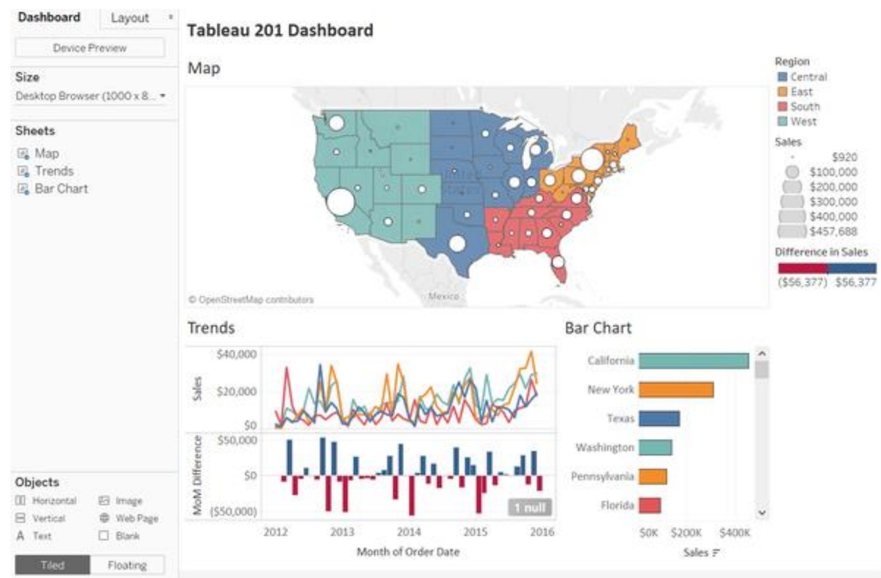


Tableau Software

Visualization – Major Areas

- Volume Visualization
- Flow Visualization

Scientific Visualization

Inherent spatial reference

3D

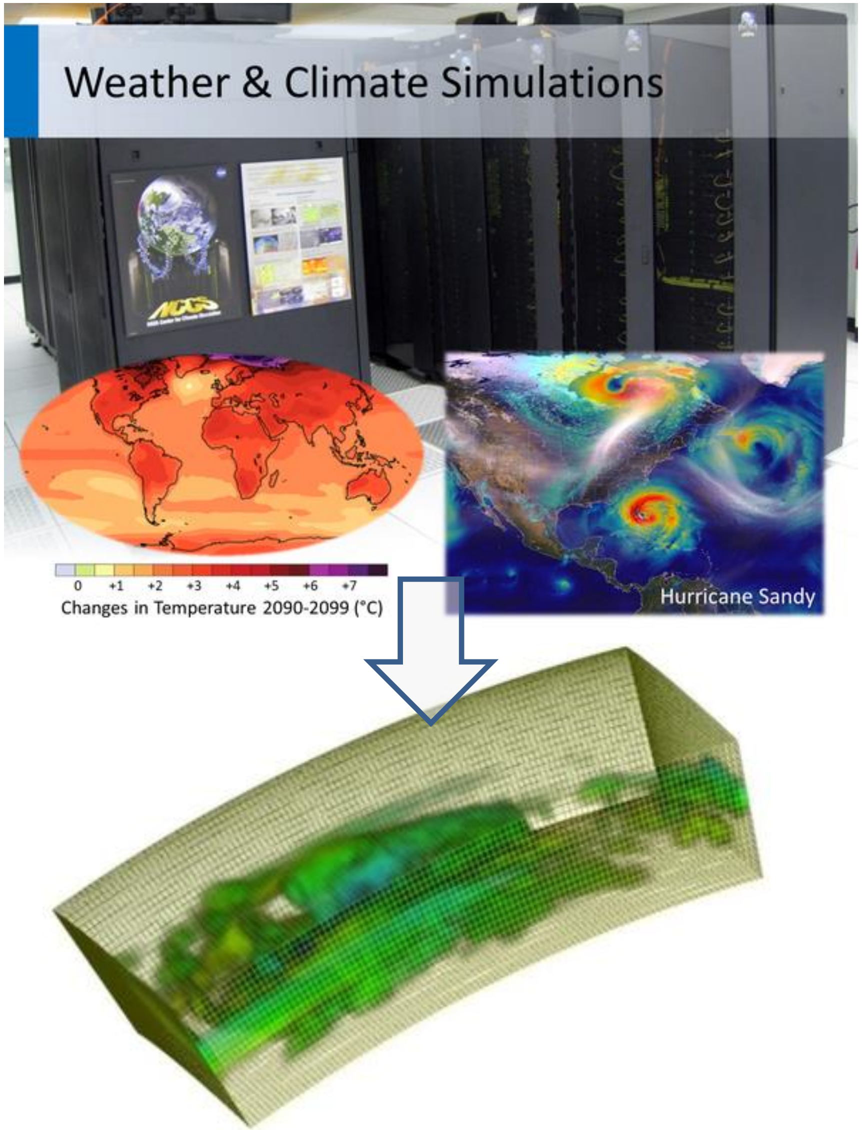
- Information Visualization
- Visual Analytics

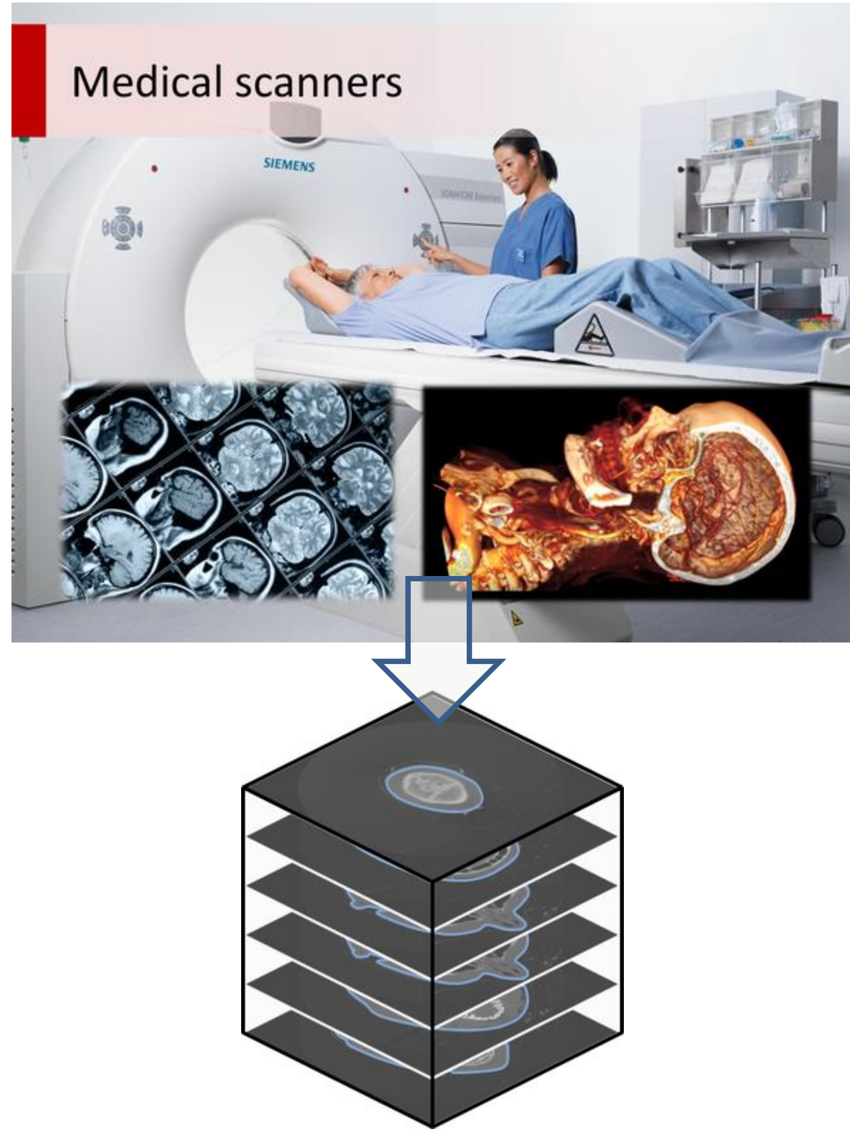
nD

Usually no spatial reference

Data representation

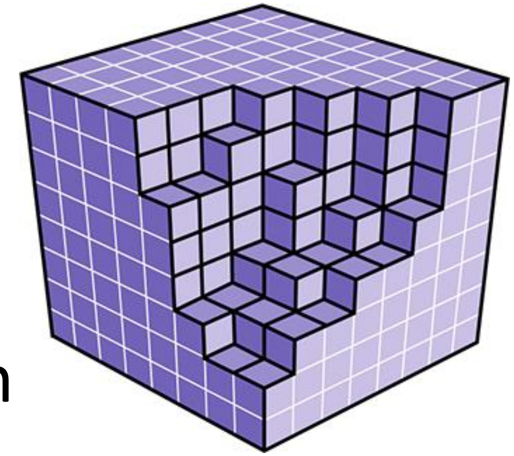
SIEMENS
Ingenuity for life





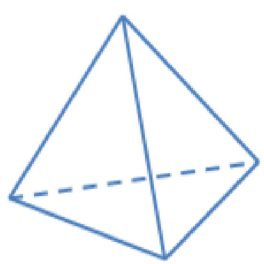
Data representation

- Discrete representations
  - Data samples (values) typically given on meshes/grids consisting of cells
  - Compact/efficient data representation

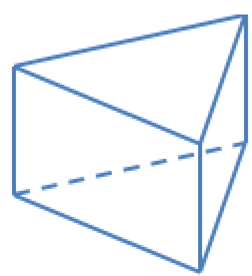


Primitives in different dimensions

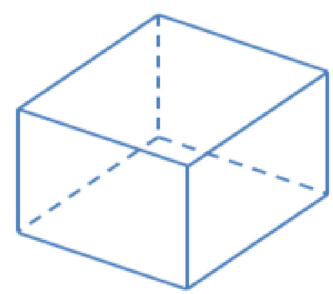
dimension	cell	mesh
0D	points	
1D	lines (edges)	polyline
2D	triangles, quadrilaterals (rectangles)	2D mesh
3D	tetrahedra, prisms, hexahedra	3D mesh or grid



Tetrahedron



Prism

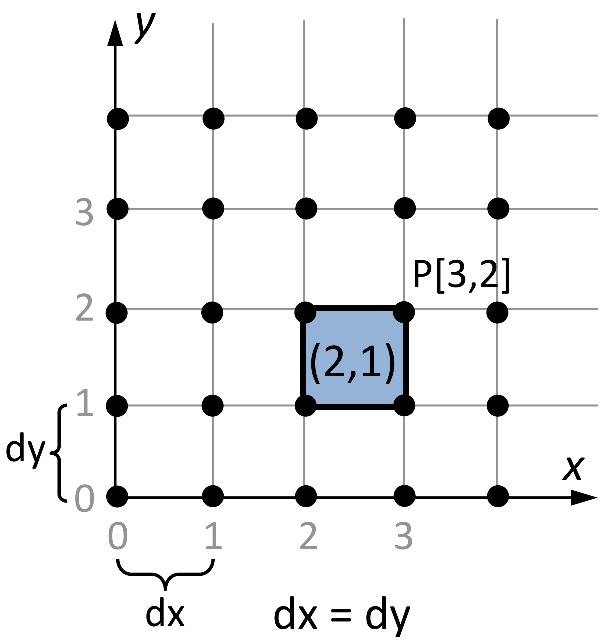


Hexahedron

Data representation

SIEMENS
Ingenuity for life

- Grids: Cartesian or equidistant grid
  - Samples at equidistant intervals along Cartesian coordinate axes
  - Neighboring samples are connected via edges
  - Cells formed by 4 (2D) or 8 (3D) samples
  - Cells and samples (grid vertices) are numbered sequentially with respect to increasing coordinates



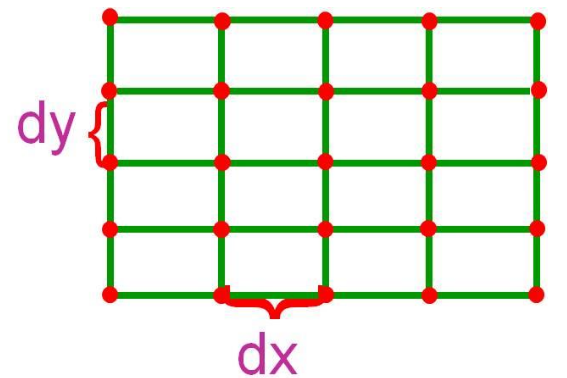
Data representation

- Some properties of Cartesian grids
  - Assuming N_x and N_y vertices along *x*- and *y*-axis
  - Number of vertices = $N_x \cdot N_y$
  - Number of cells = $(N_x - 1) \cdot (N_y - 1)$
  - Vertex positions are given implicitly from indices [*i*,*j*]:
    - P[*i*,*j*].x = origin + $i \cdot$ dx
    - P[*i*,*j*].y = origin + $j \cdot$ dy
  - It is a structured grid
    - Neighboring information (topology) is given implicitly
    - Neighbors obtained by incrementing/decrementing indices

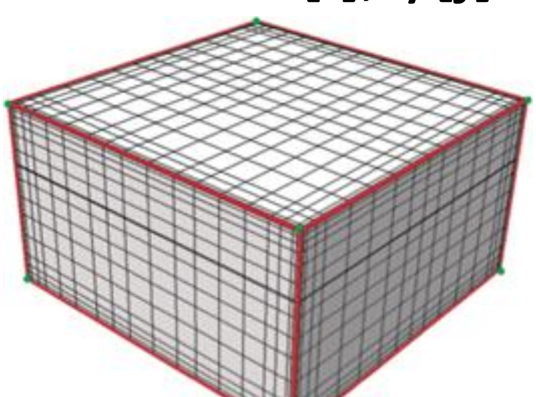
Data representation

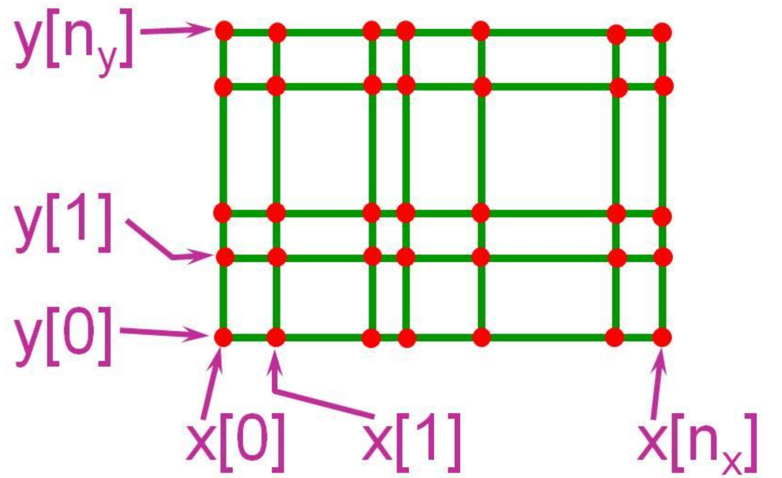
SIEMENS
Ingenuity for life

- Uniform or Regular Grid
  - Orthogonal, equidistant grid
  - dx ≠ dy



- Rectilinear Grid
  - Varying sample-distances x[i], y[j]



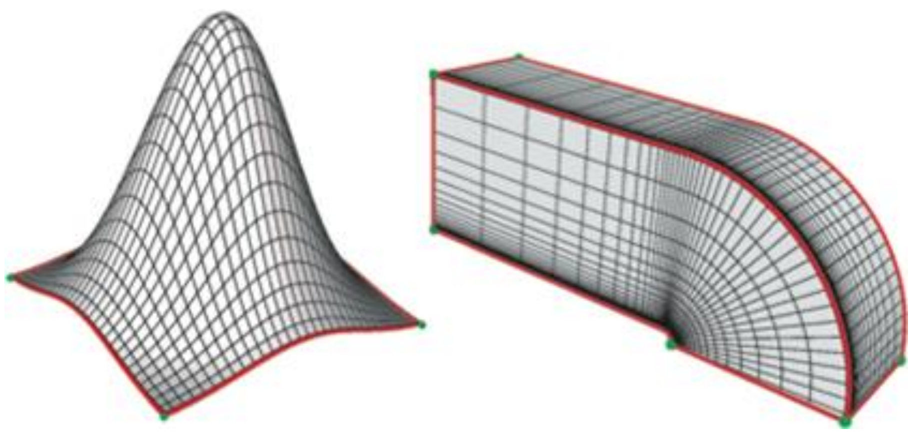


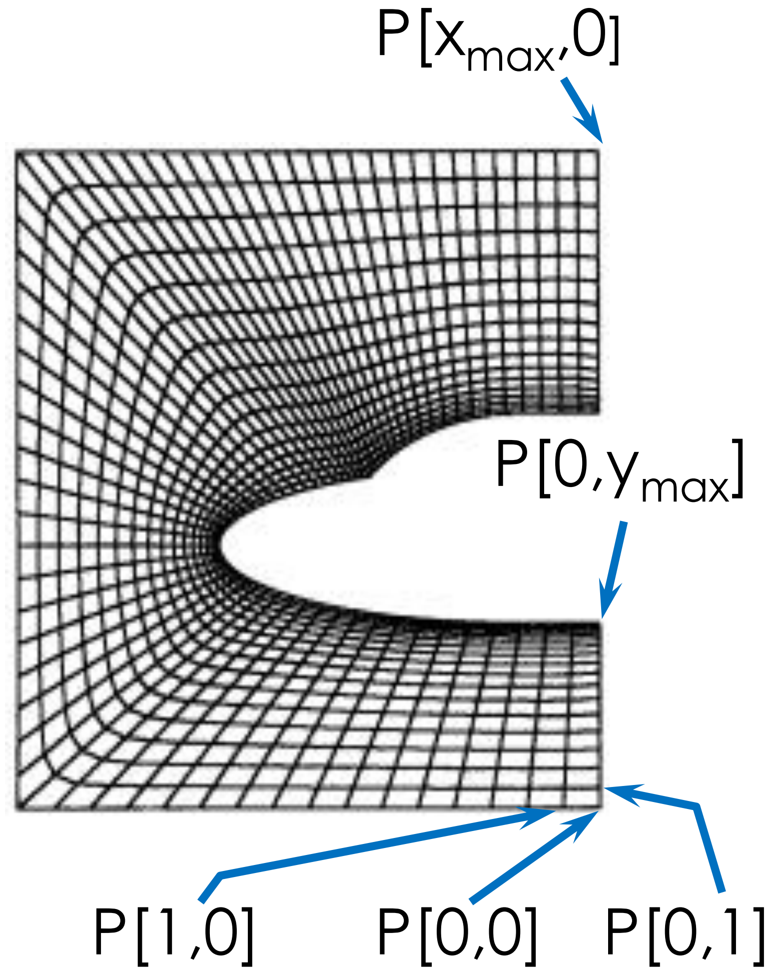
Gröller & Hauser / Telea

Data representation

SIEMENS
Ingenuity for life

- Curvilinear Grid
  - Non-orthogonal grid
  - Grid-points specified explicitly (P[i,j])
  - Implicit neighborhood relationship



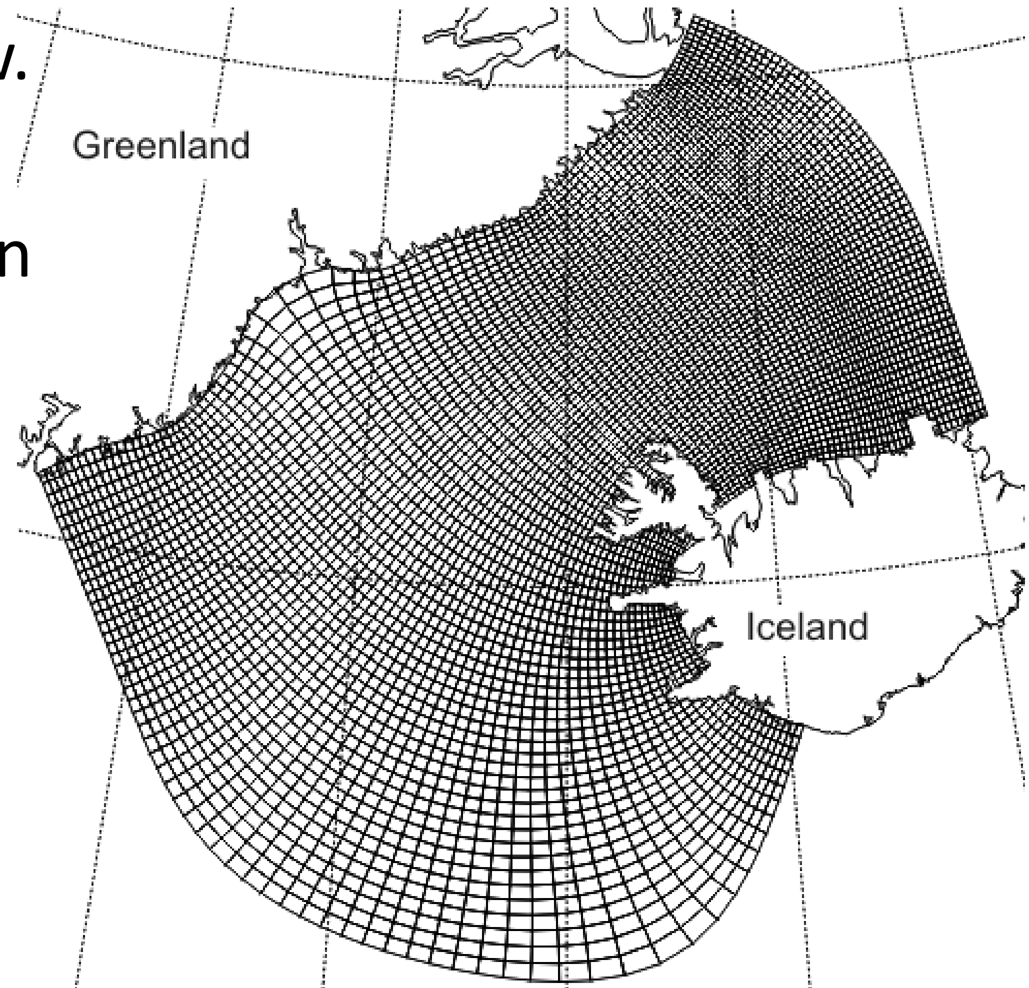


Gröller & Hauser / Telea

Data representation

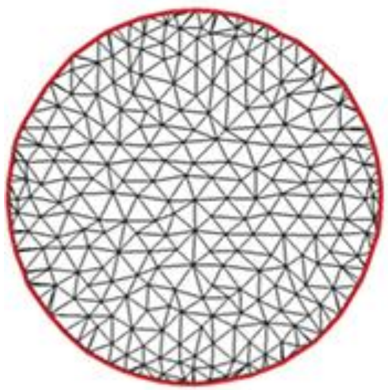
SIEMENS
Ingenuity for life

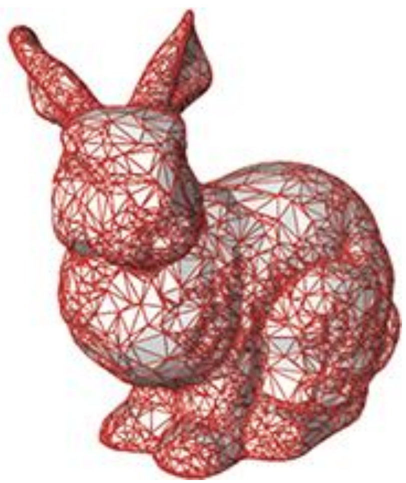
- Curvilinear Grid
  - Example: Region betw. Greenland & Iceland
  - Grid spacings between 5 and 40 km

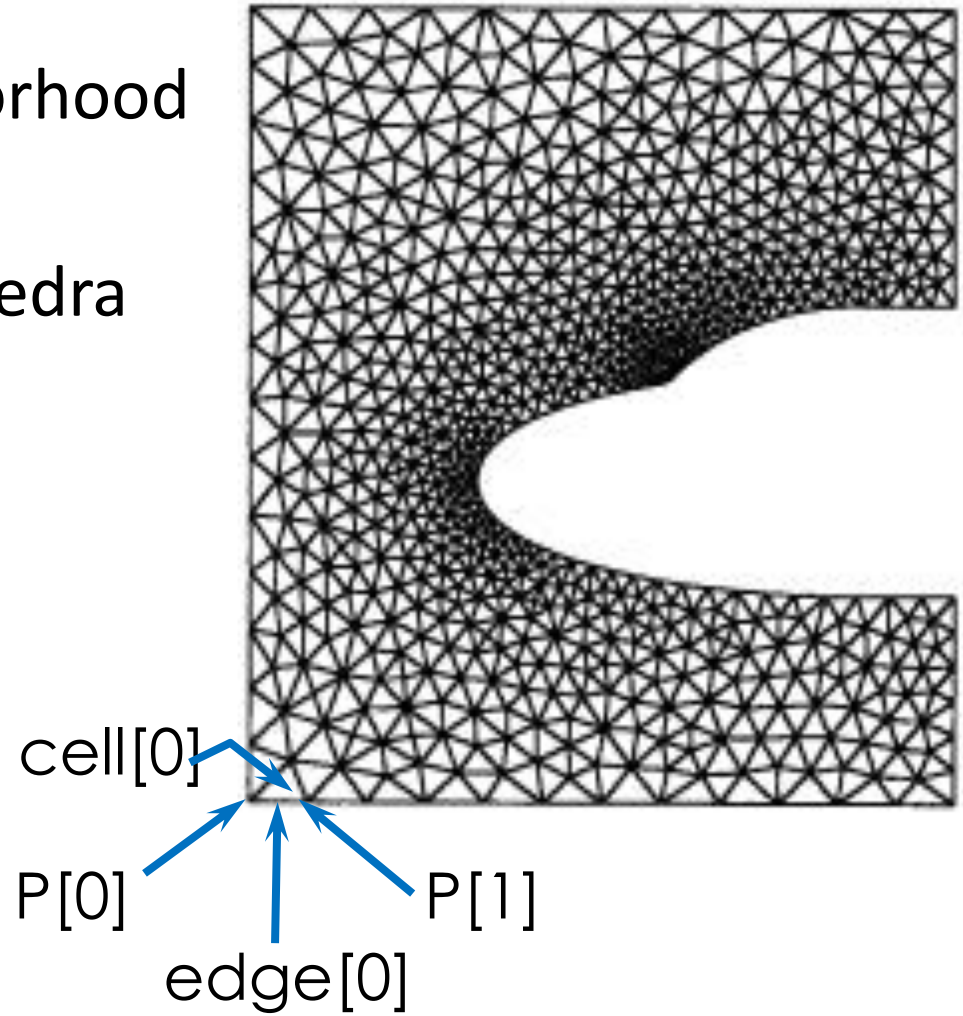


Data representation

- Unstructured grid
  - Grid points and neighborhood specified explicitly
  - Cells: tetrahedra, hexahedra



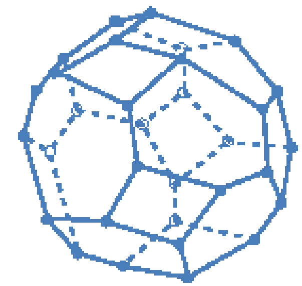




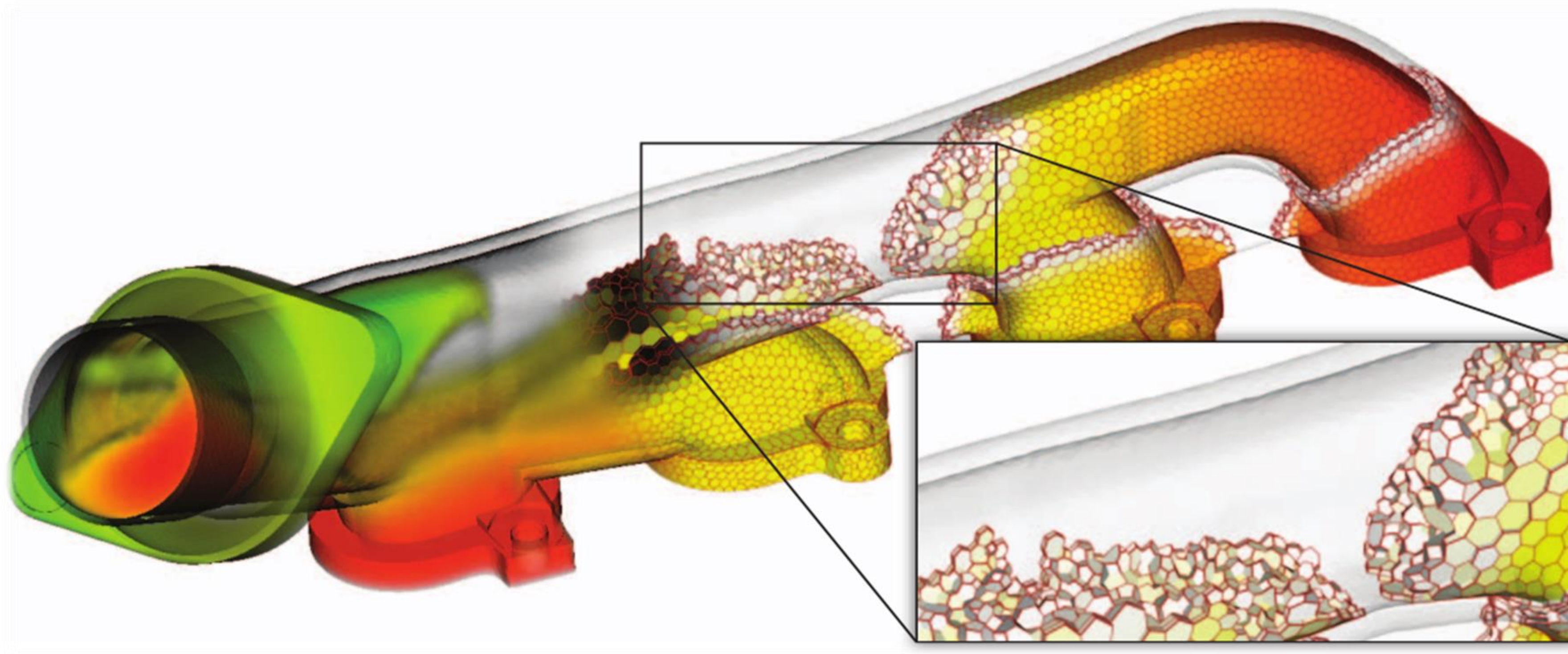
Gröller & Hauser / Telea

Data representation

SIEMENS
Ingenuity for life



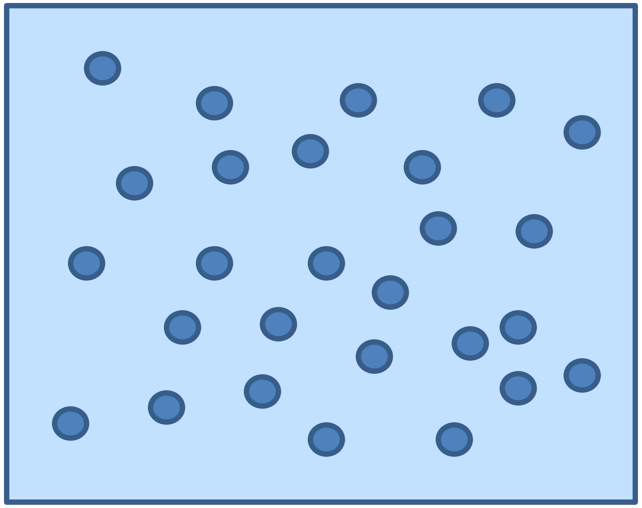
- Unstructured grid
  - Example: Exhaust Manifold with general (non-convex) polyhedra



Muigg et al. 2011

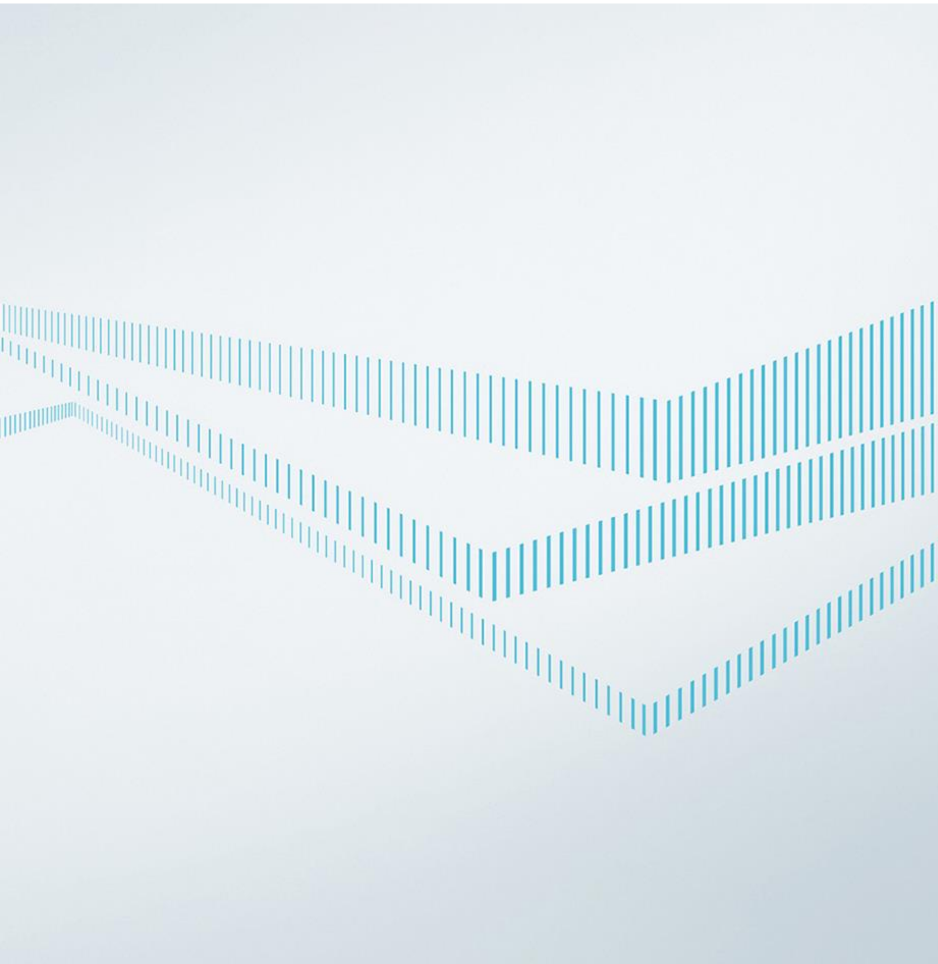
Data representation

- Scattered Data
  - Grid-free data
  - Data points given without neighborhood-relationship
  - Influence on neighborhood defined by spatial proximity
  - Scattered data interpolation



Contact information

SIEMENS
Ingenuity for life

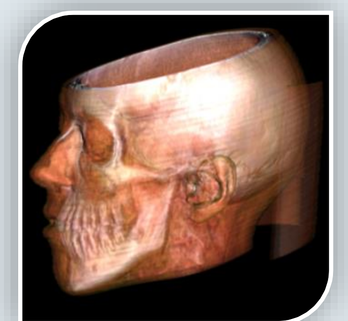


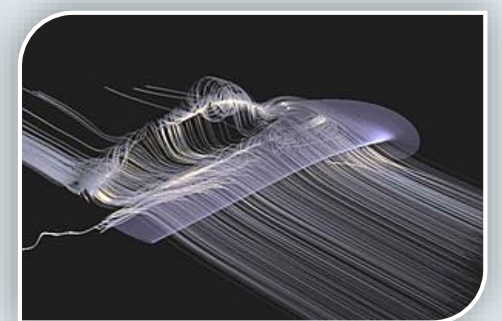
Dr. Johannes Kehrer

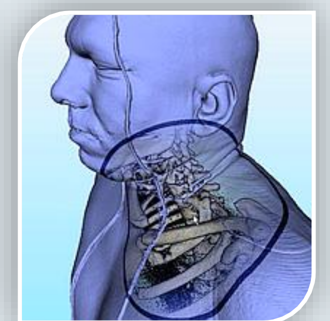
Siemens AG
Technology
T DAI HCA-DE
Otto-Hahn-Ring 6
81739 München, Deutschland

E-mail:
kehrer.johannes@siemens.com

Internet
siemens.com/innovation







Visual Data Analytics

Data Reconstruction and Interpolation

Dr. Johannes Kehrer

Disclaimer

SIEMENS
Ingenuity for life

These lecture slides and materials are for personal use by students and are intended for educational purposes only. Distribution or reproduction of the material is prohibited.

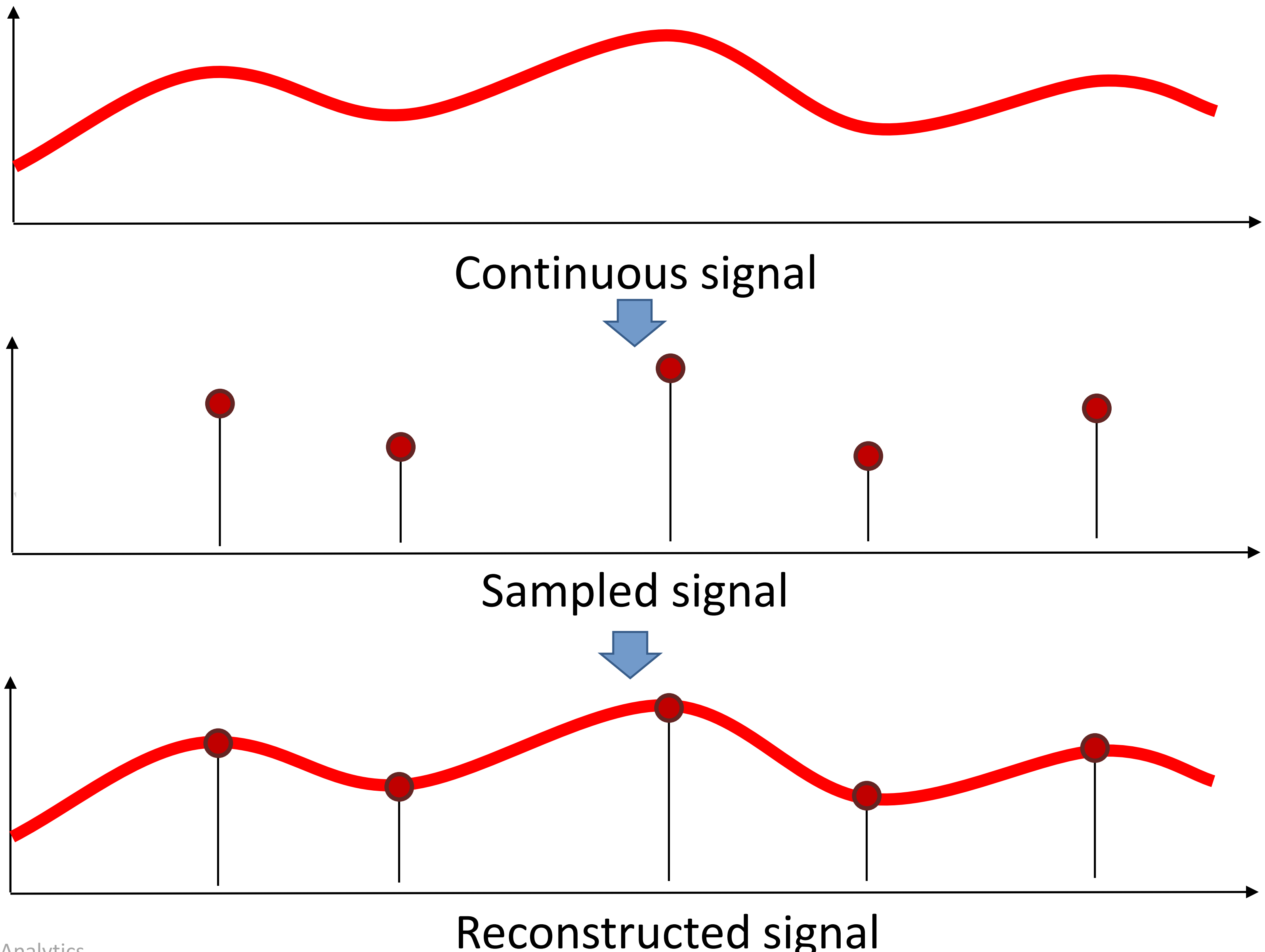
The views and opinions expressed by any individual during the course of these lecturers are their own and do not necessarily represent the views, opinions, or positions of Siemens or the Technical University of Munich.

Although reasonable efforts have been made to ensure the information in these lecture slides is current and accurate, we do not assume any responsibility for the content of the posted material.

These slides also contain links to websites of third parties. As the content of these websites is not under our control, we cannot assume any liability for such external content.

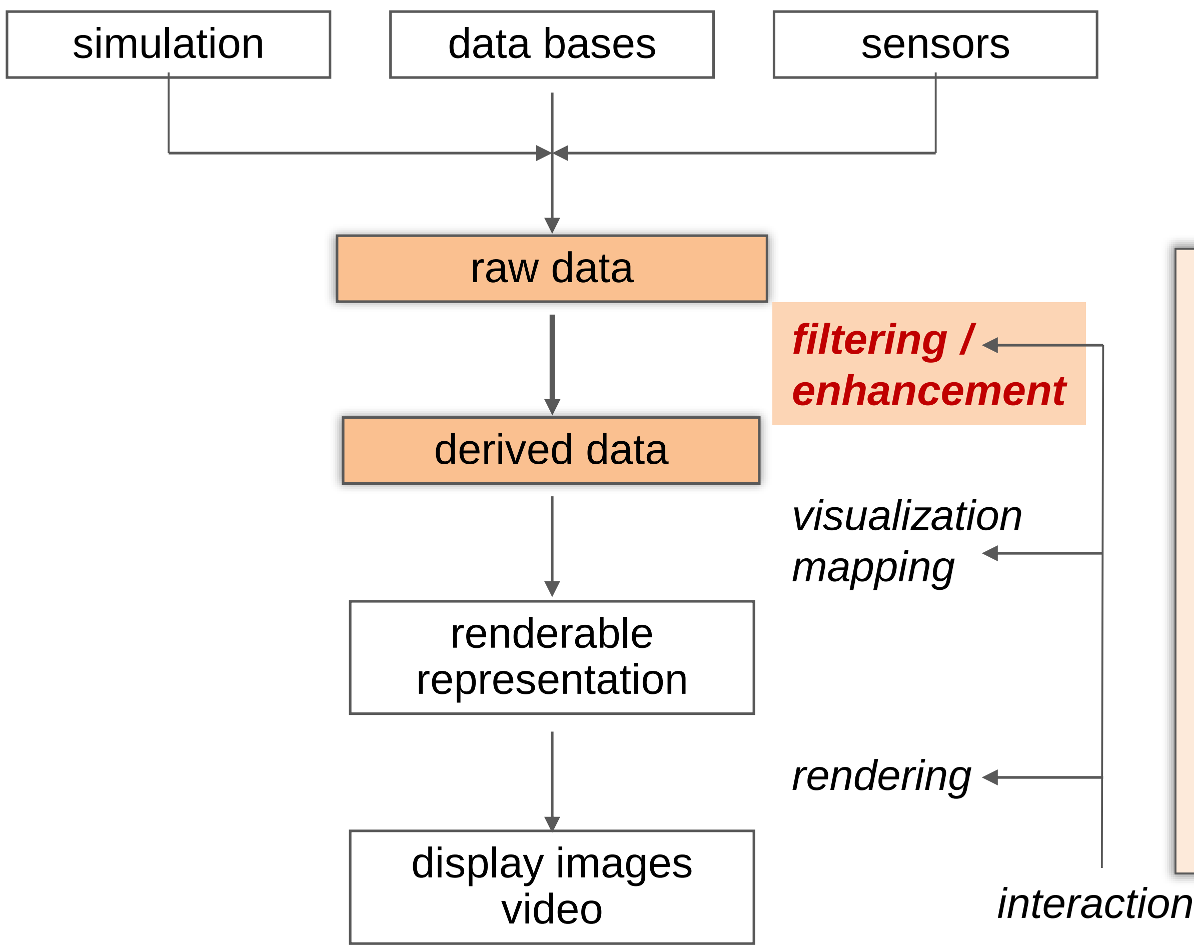
Sampling & Reconstruction

SIEMENS
Ingenuity for life



Data reconstruction

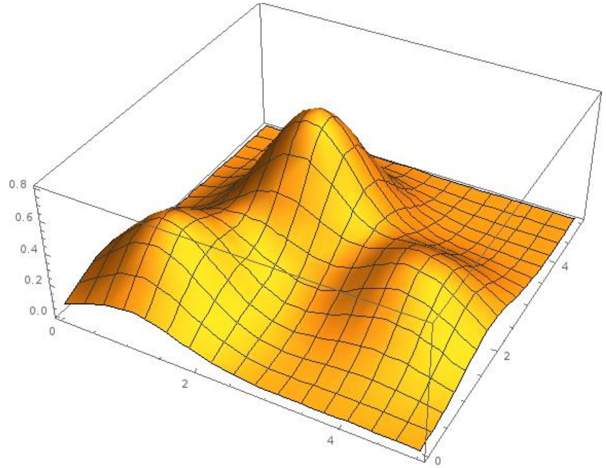
- From discrete samples to a continuous representation

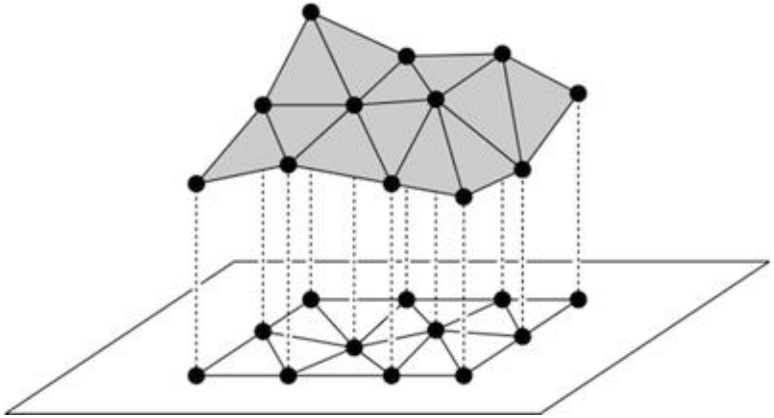


Process to obtain useful data (e.g., 3D volume)
- data format conversion
- co-registration of data sets
- resampling to grid
- interpolation/approximation of missing values
- data reduction (remove nonrelevant data)
- clipping/cleaning/denoising
- characteristic properties (gradients, curvature, extrema, segmentation)

Overview

- Scattered data interpolation
  - Continuous interpolation functions
  - Piecewise interpolation via triangulation



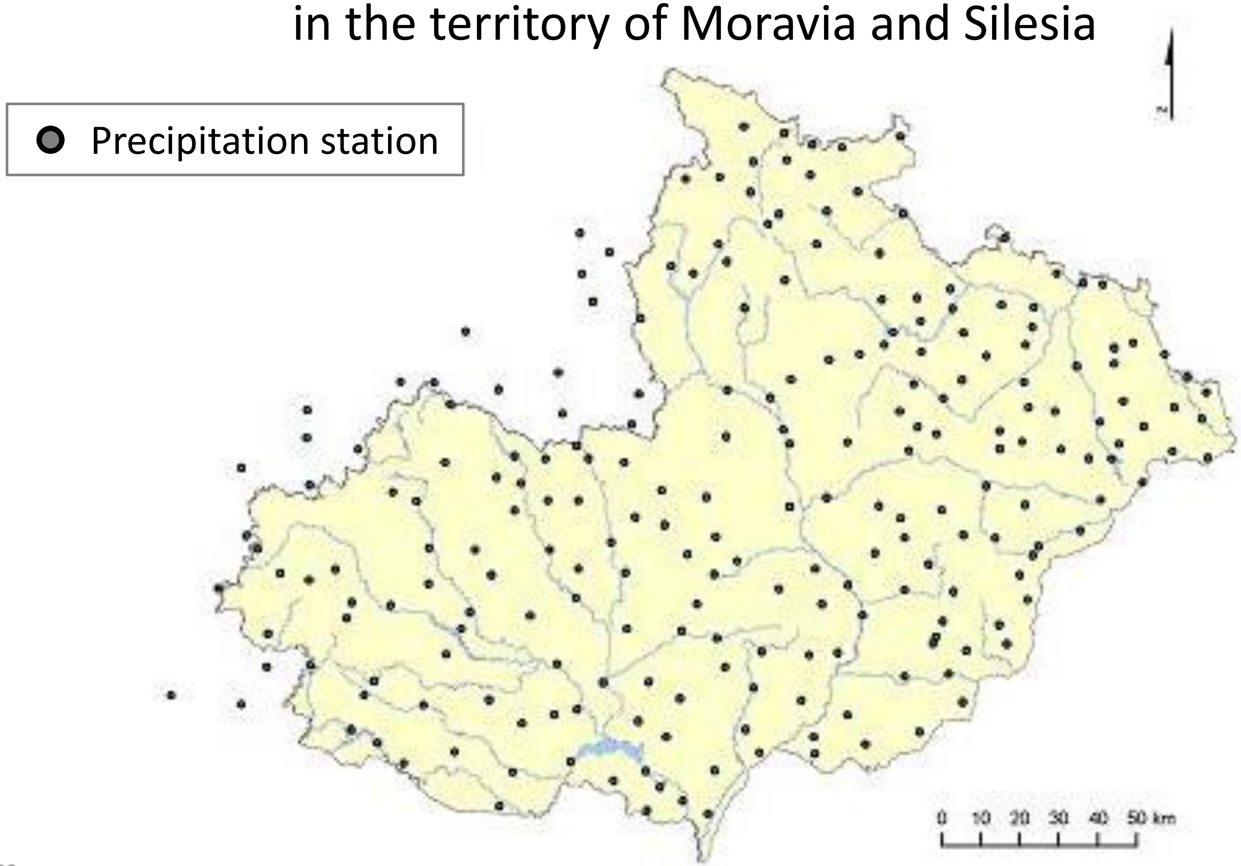


- Interpolation on grids (next lecture)

Data reconstruction

- Initial data often given at a discrete set of scattered points (samples) in the domain

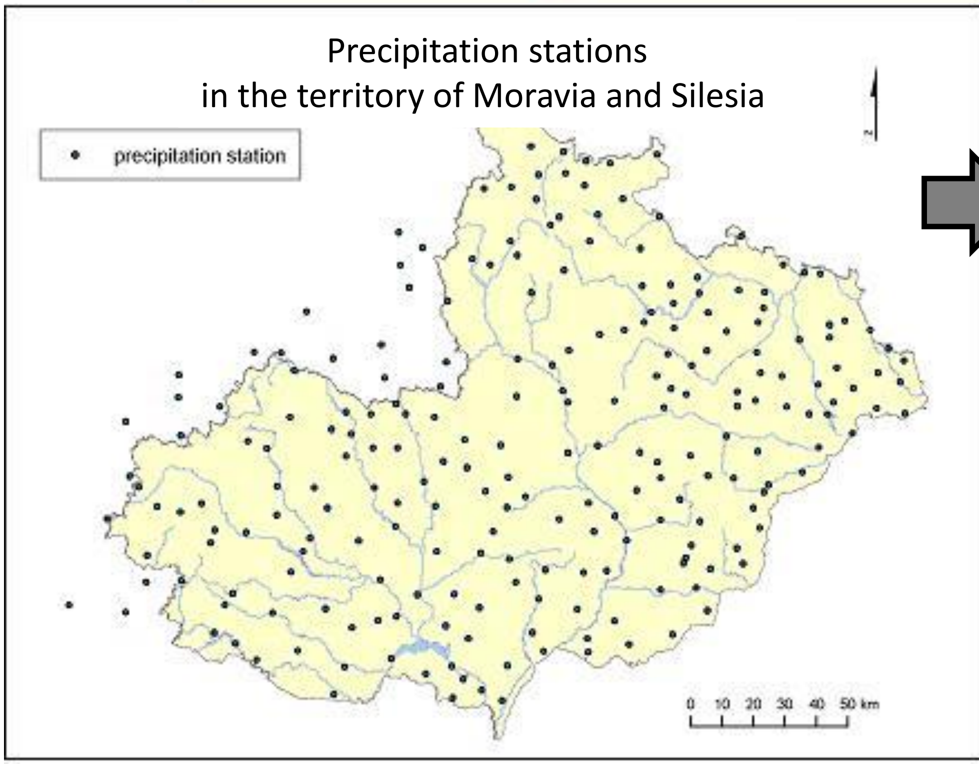
Precipitation stations
in the territory of Moravia and Silesia

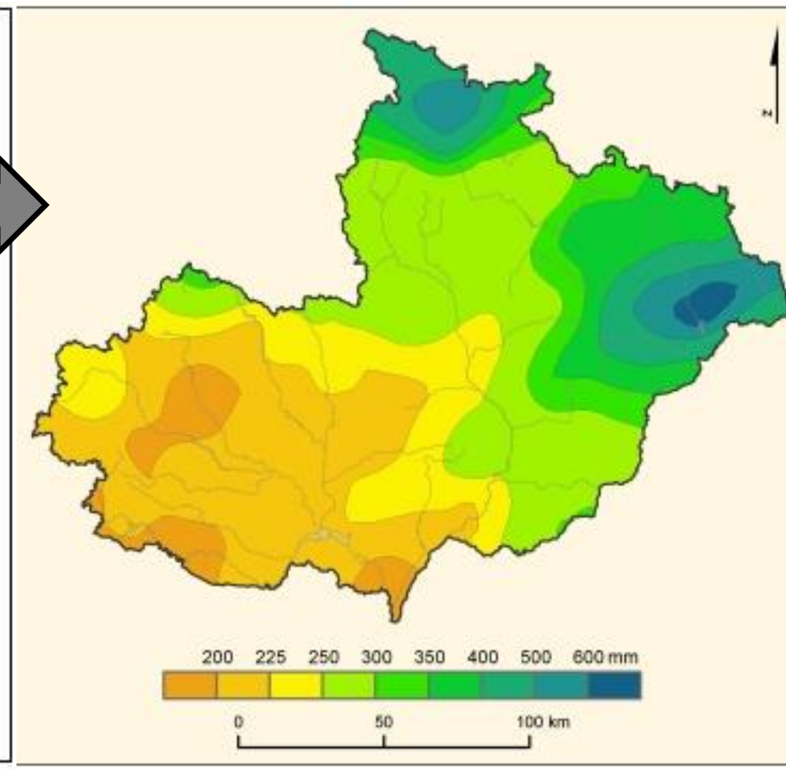


Data reconstruction

SIEMENS
Ingenuity for life

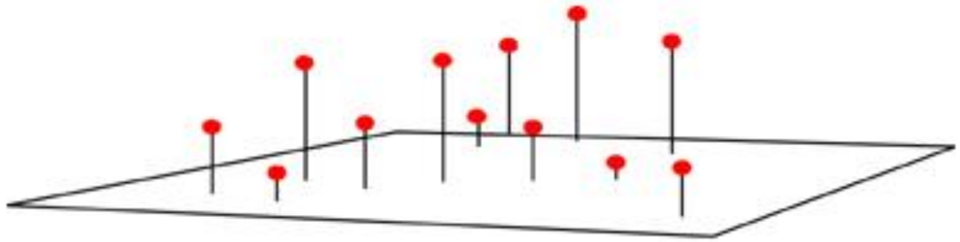
- We want to derive a continuous representation from data given at scattered points
  - Better communication of spatial data distribution

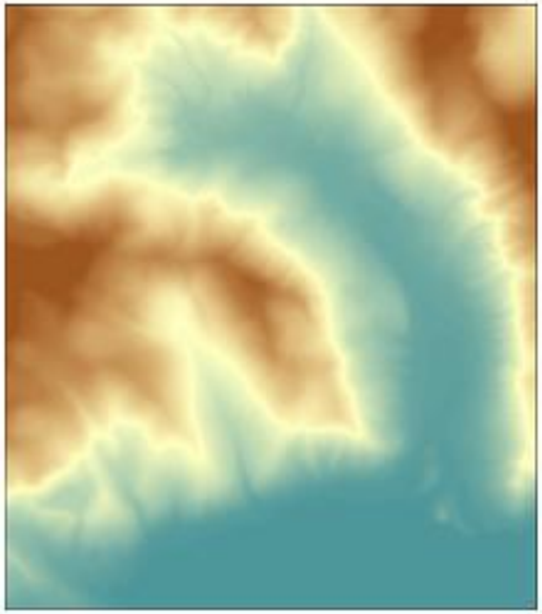




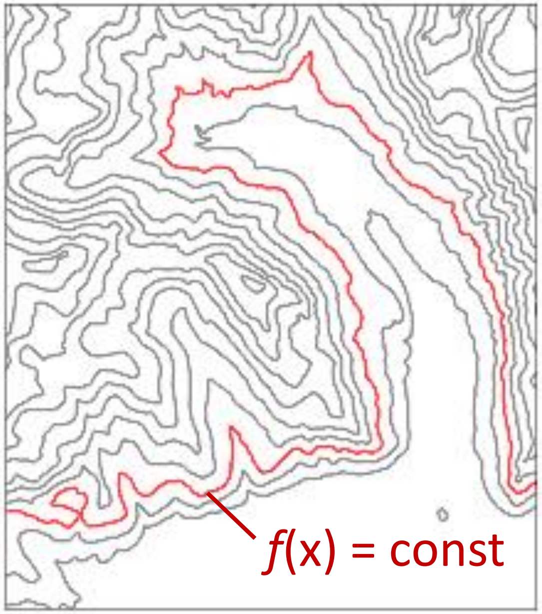
Data reconstruction

- Some analysis techniques require a continuous representation





Data distribution



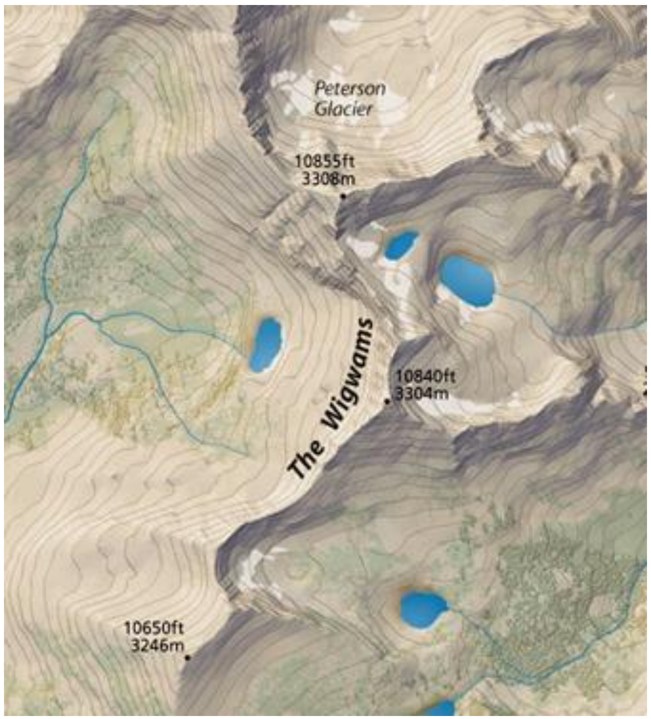
Isocontours – curves on which all points have a certain value

Data reconstruction

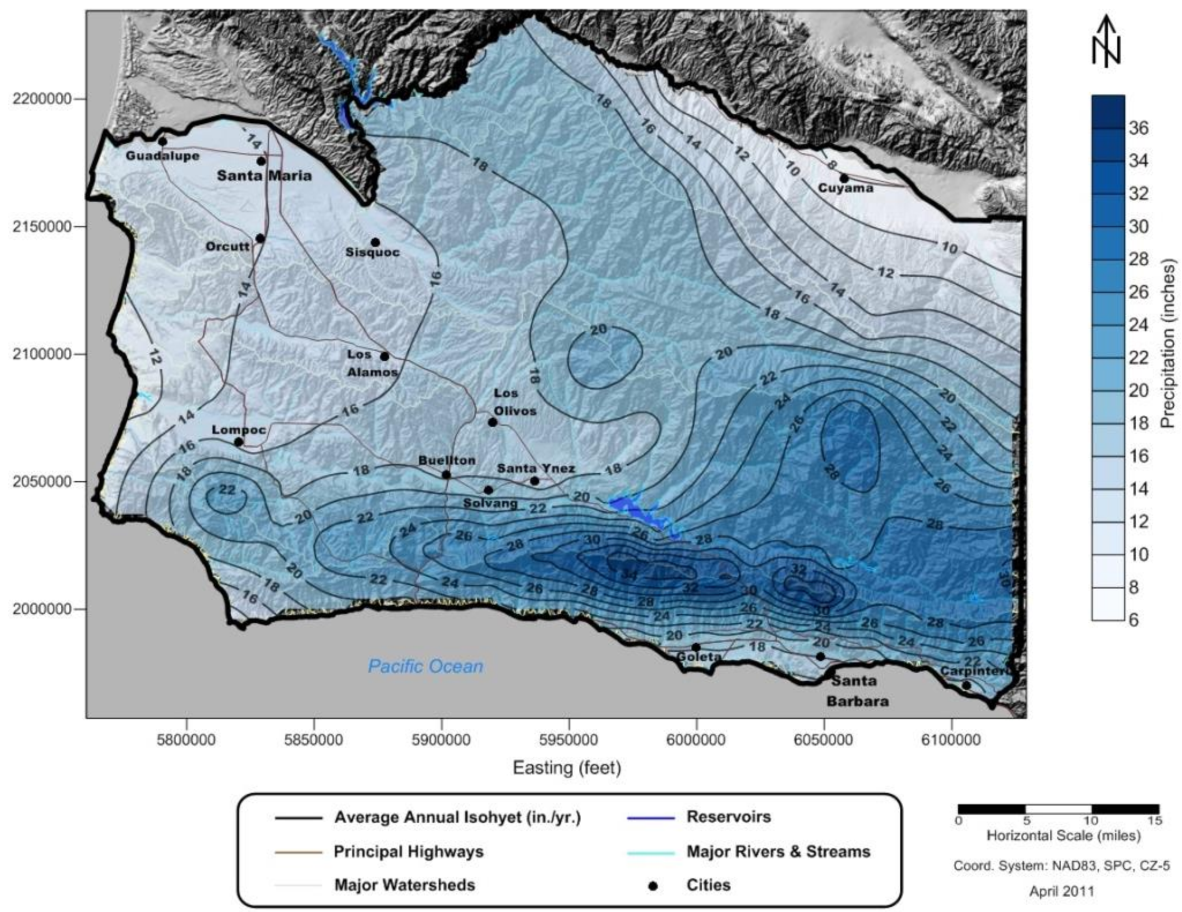
SIEMENS
Ingenuity for life

- Isocontours/lines in continuous data fields

Geographic map with isolines

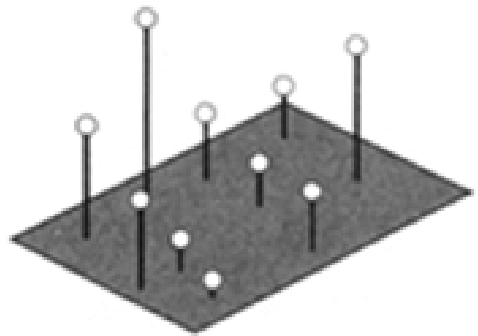


County of Santa Barbara
Hydrology Section
Average Annual Isohyetal Map (precipitation contours)

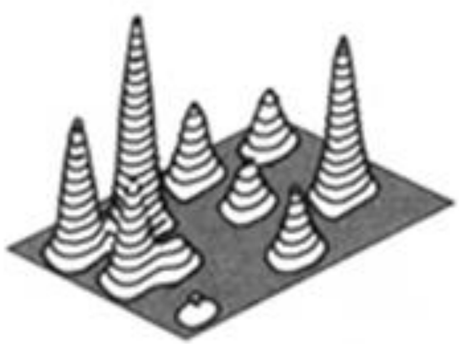


Data reconstruction

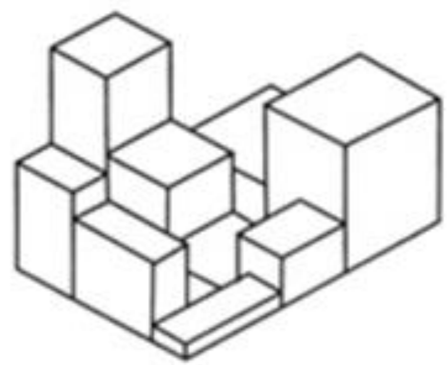
- Data reconstruction from scattered points



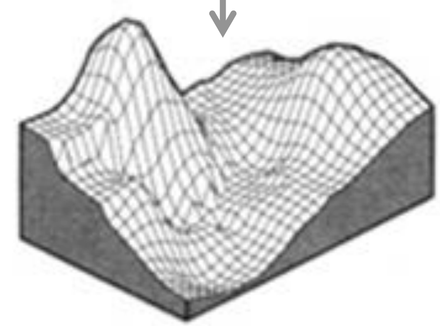
Assumes some similarity between data values inversely proportional to distance



data interpolation



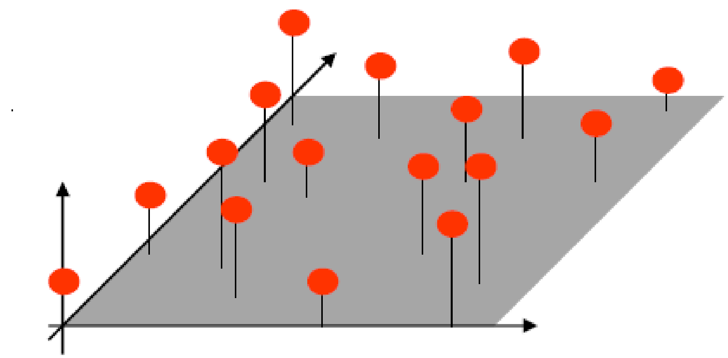
piece-wise constant interpolation



continuous interpolation

Data reconstruction

- Data reconstruction from scattered points

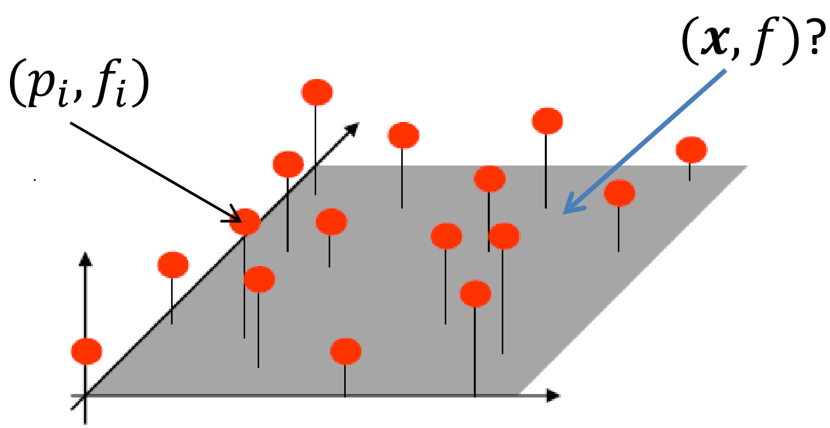


- Strategies to obtain values in-between data points…
  1. from a continuous function which interpolates the given values and varies smoothly in-between
  2. from a grid which is constructed from the given points, i.e., a triangulation

Continuous representation

- Given a set of scattered points p_i in a 2D parameter domain with scalar values f_i
  – The principles are applicable to arbitrary parameter domain dimensions (1D/2D/3D)

- **Goal:** Construct a continuous function f from given set of p_i, f_i which approximates ("follows") the given values



Continuous representation

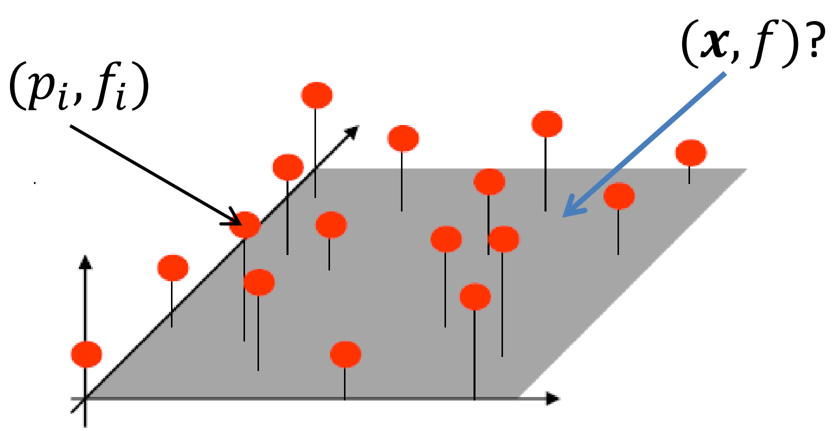
- Radial Basis Functions
  - Independent of dimension of parameter domain (1D/2D/3D)
  - Function *f* represented as weighted sum of *N* radial functions φ

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} f_i \, \varphi(\|p_i - \boldsymbol{x}\|)$$

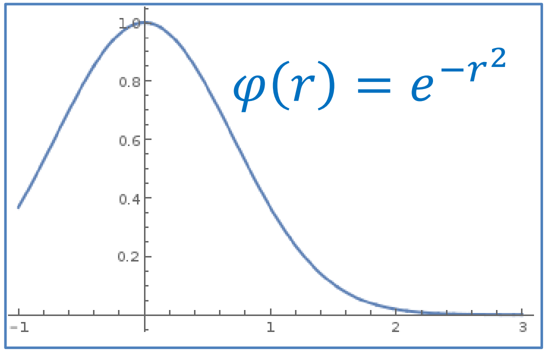
  - Each (p_i, f_i) influences $f(\boldsymbol{x})$ based on Euclidean distance

$$r = \|p_i - \boldsymbol{x}\|$$

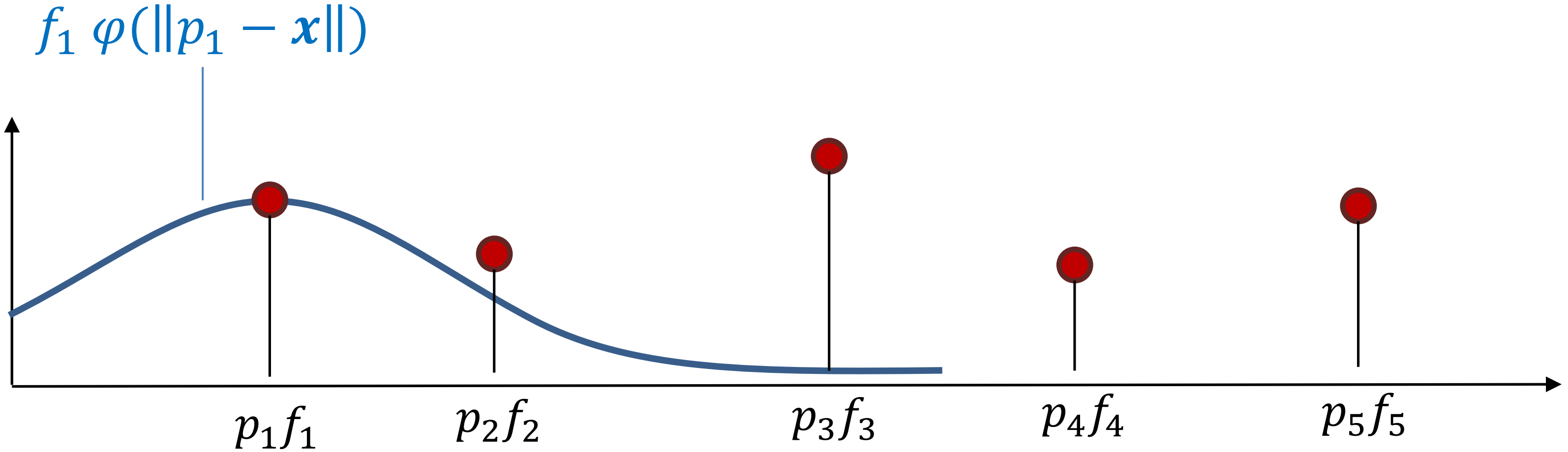
  - Nearby points have higher influence than far-away points



Continuous representation

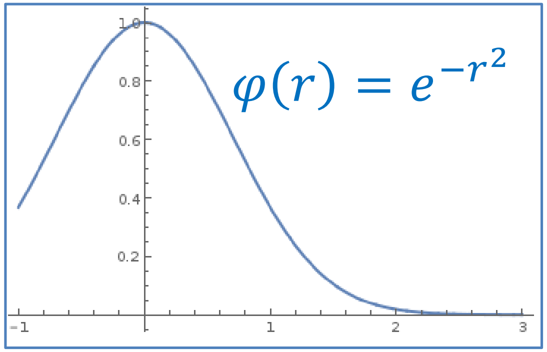


- Radial Basis Functions
  - Each radial function $\varphi(r)$ is centered around a data point p_i
  - Value of $\varphi(r)$ decreases quickly with increasing distance $r = \|p_i - \boldsymbol{x}\|$ to the function's center p_i

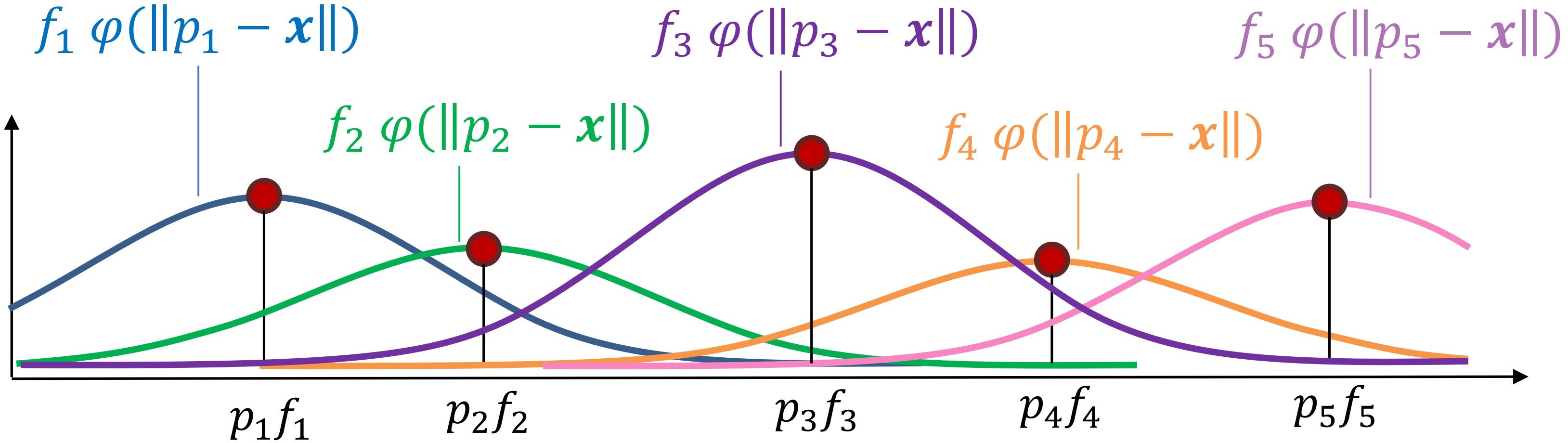


multivis.net/lecture/radial-basis.html

Continuous representation

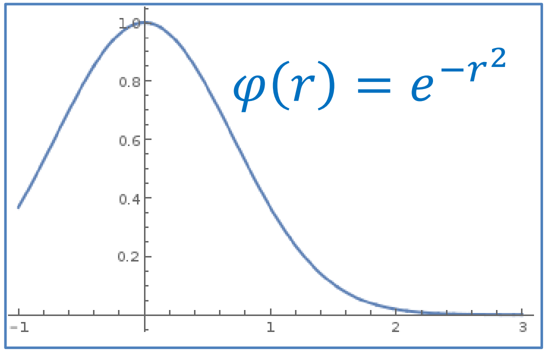


- Radial Basis Functions
  - Each radial function $\varphi(r)$ is centered around a data point p_i
  - Value of $\varphi(r)$ decreases quickly with increasing distance $r = \|p_i - \boldsymbol{x}\|$ to the function's center p_i



multivis.net/lecture/radial-basis.html

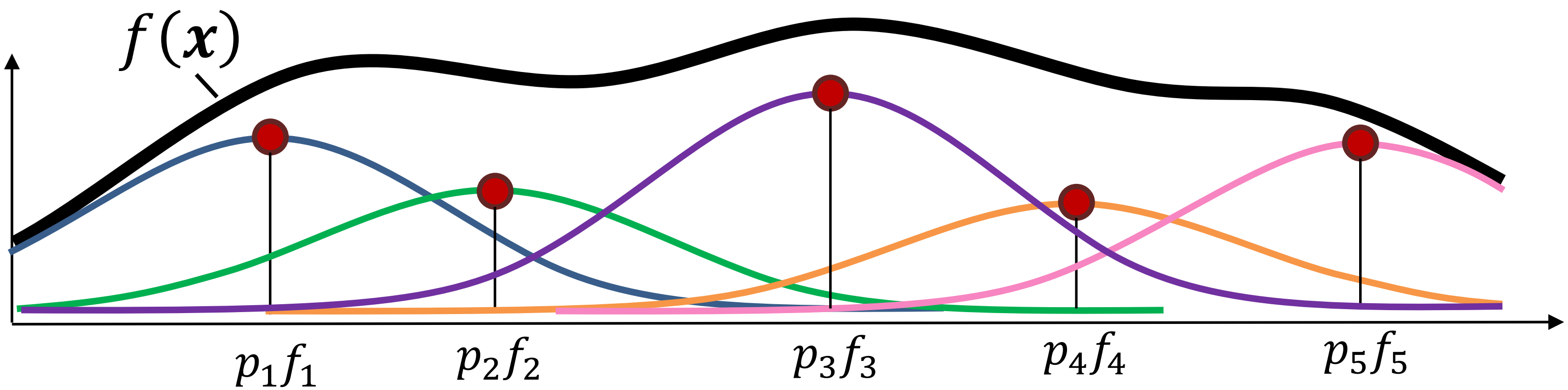
Continuous representation



- Radial Basis Functions
  - Each radial function $\varphi(r)$ is centered around a data point p_i
  - Value of $\varphi(r)$ decreases quickly with increasing distance $r = \|p_i - \boldsymbol{x}\|$ to the function's center p_i
  - Function *f* represented as weighted sum of *N* radial functions φ

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} f_i \, \varphi(\|p_i - \boldsymbol{x}\|)$$

where $\|\cdot\|$ is the Euclidean distance (length of a vector)



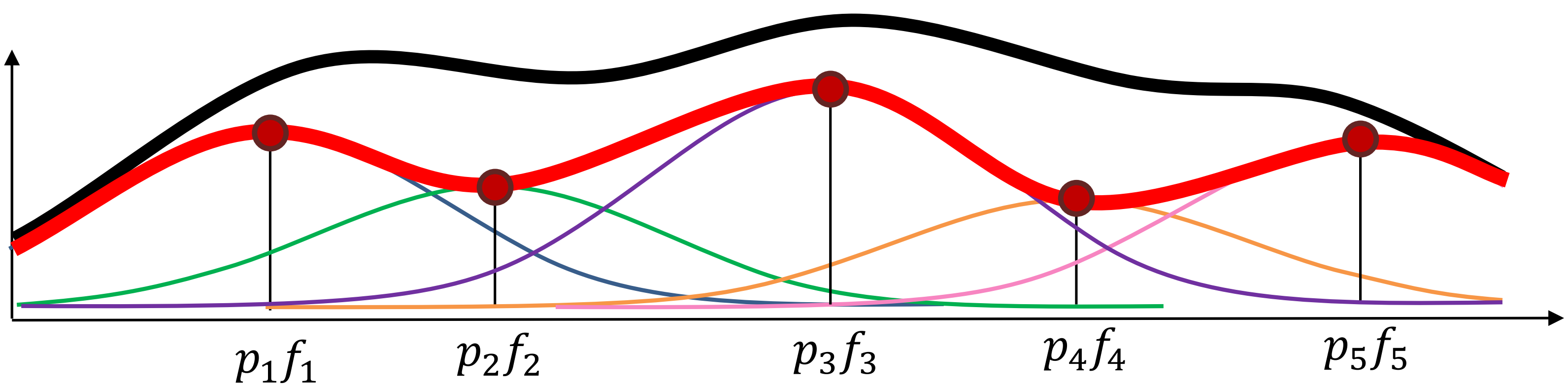
multivis.net/lecture/radial-basis.html

Continuous representation

SIEMENS
Ingenuity for life

- Radial Basis Functions
  - Instead of the **black** curve we want the **red** one, i.e., a curve which is going through the initial data points
  - This is called an interpolation
  - **Question:** How do we have to select the weights w_i so that the red curve is obtained?

$$f(\boldsymbol{x}) = \sum_i w_i\, \varphi(\|p_i - \boldsymbol{x}\|)$$



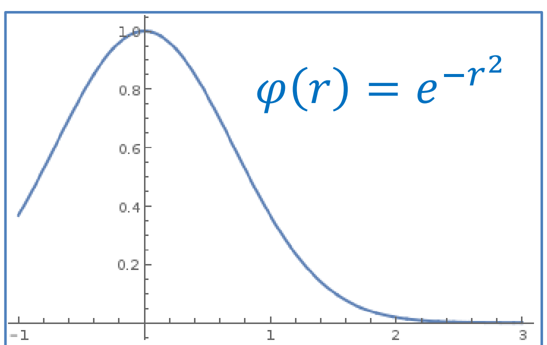
multivis.net/lecture/radial-basis.html

Continuous representation

SIEMENS
Ingenuity for life

- Example:
  - Data points: $p_1 = 1$, $p_2 = 3$, $p_3 = 4$
  - Data values: $f_1 = 1$, $f_2 = {}^3\!/_5$, $f_3 = 0$
- Find the weights w_i such that f interpolates all points

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} w_i\, \varphi(\|p_i - \boldsymbol{x}\|) \quad \text{where} \quad \varphi(r) = e^{-r^2}$$



r	0	0.5	1	1.5	2	2.5	3
$\varphi(r)$	1	$4/5$	$2/5$	$1/10$	0	0	0

Continuous representation

- Radial Basis Functions – finding the weights w_i
  - For $j = 1, \dots, N$,
    specify w_i such that $f(p_j)$ interpolates the value f_j

$$f(p_j) = \sum_{i=1}^{N} w_i \varphi(\|p_i - \boxed{p_j}\|) = \boxed{f_j}$$

Continuous representation

SIEMENS
Ingenuity for life

- Radial Basis Functions – finding the weights w_i
  - For $j = 1, \ldots, N$,
    specify w_i such that $f(p_j)$ interpolates the value f_j

$$f(p_j) = \sum_{i=1}^{N} w_i \varphi(\|p_i - p_j\|) = f_j$$

  - Yields a system of linear equations (per point) to be solved for w_i

$$\begin{bmatrix} \varphi(\|p_1 - p_1\|) & \varphi(\|p_2 - p_1\|) & \cdots & \varphi(\|p_N - p_1\|) \\ \varphi(\|p_1 - p_2\|) & \varphi(\|p_2 - p_2\|) & \cdots & \varphi(\|p_N - p_2\|) \\ \vdots & \vdots & \ddots & \vdots \\ \varphi(\|p_1 - p_N\|) & \varphi(\|p_2 - p_N\|) & \cdots & \varphi(\|p_N - p_N\|) \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_N \end{bmatrix}$$

N equations in N unkowns

Continuous representation

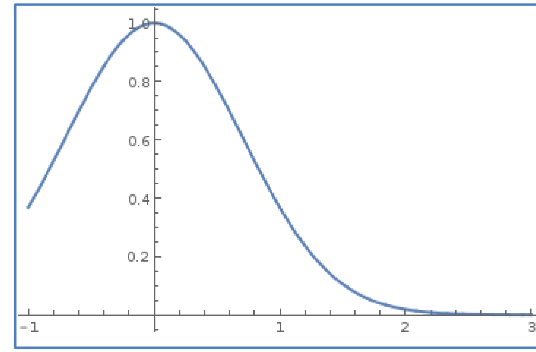
SIEMENS
Ingenuity for life

- Radial Basis Functions – finding the weights w_i
  - For $j = 1, \dots, N$,
    specify w_i such that $f(p_j)$ interpolates the value f_j
    $$f(p_j) = \sum_{i=1}^{N} w_i \varphi(\|p_i - p_j\|) = f_j$$
  - Yields a system of linear equations (per point) to be solved for w_i

$$\begin{bmatrix} \varphi(0) & \varphi(\|p_2 - p_1\|) & \cdots & \varphi(\|p_N - p_1\|) \\ \varphi(\|p_1 - p_2\|) & \varphi(0) & \cdots & \varphi(\|p_N - p_2\|) \\ \vdots & \vdots & \ddots & \vdots \\ \varphi(\|p_1 - p_N\|) & r(\|p_2 - p_N\|) & \cdots & \varphi(0) \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_N \end{bmatrix}$$

Continuous representation

- Example (cont.)
  - Data points: $p_1 = 1,\ p_2 = 3,\ p_3 = 4$
  - Data values: $f_1 = 1,\ f_2 = {}^{3}/_{5},\ f_3 = 0$
  - Radial function: $\varphi(r) = e^{-r^2}$



$$\Rightarrow \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & {}^{2}/_{5} \\ 0 & {}^{2}/_{5} & 1 \end{bmatrix}}_{\boldsymbol{R}} \underbrace{\begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix}}_{\boldsymbol{w}} = \underbrace{\begin{bmatrix} 1 \\ {}^{3}/_{5} \\ 0 \end{bmatrix}}_{\boldsymbol{f}}$$

$1 = w_1$
$3/5 = w_2 + \frac{2}{5} w_3$
$0 = 2/5\, w_2 + w_3$

If matrix $\boldsymbol{R}$ is invertible,

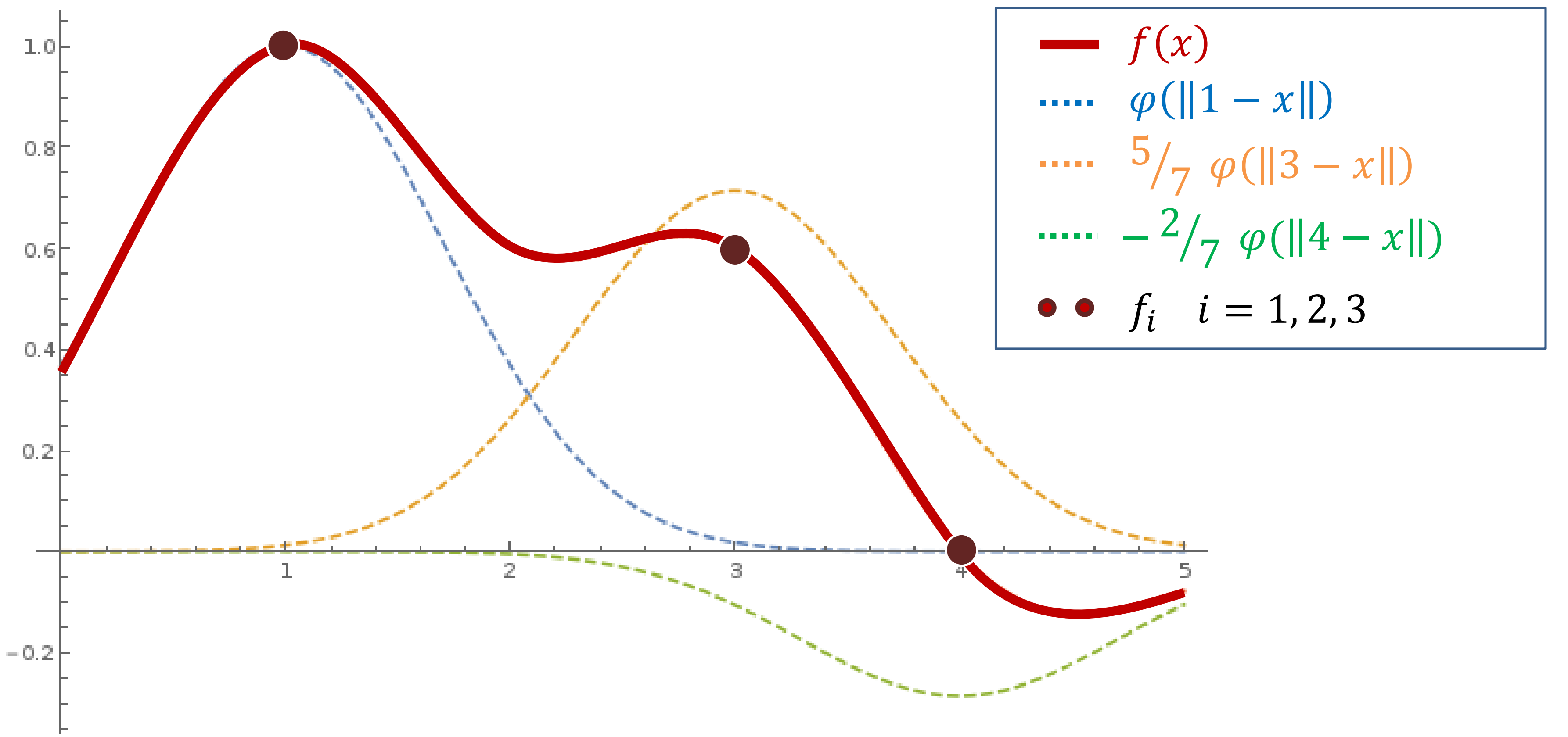
$\boldsymbol{w} = \boldsymbol{R}^{-1}\boldsymbol{f}$ with solution: $w_1 = 1, w_2 = {}^{5}/_{7}, w_3 = -{}^{2}/_{7}$

Continuous representation

SIEMENS
Ingenuity for life

- Example

$$f(x) = \varphi(\|1 - x\|) + {}^5\!/_7\, \varphi(\|3 - x\|) - {}^2\!/_7\, \varphi(\|4 - x\|)$$

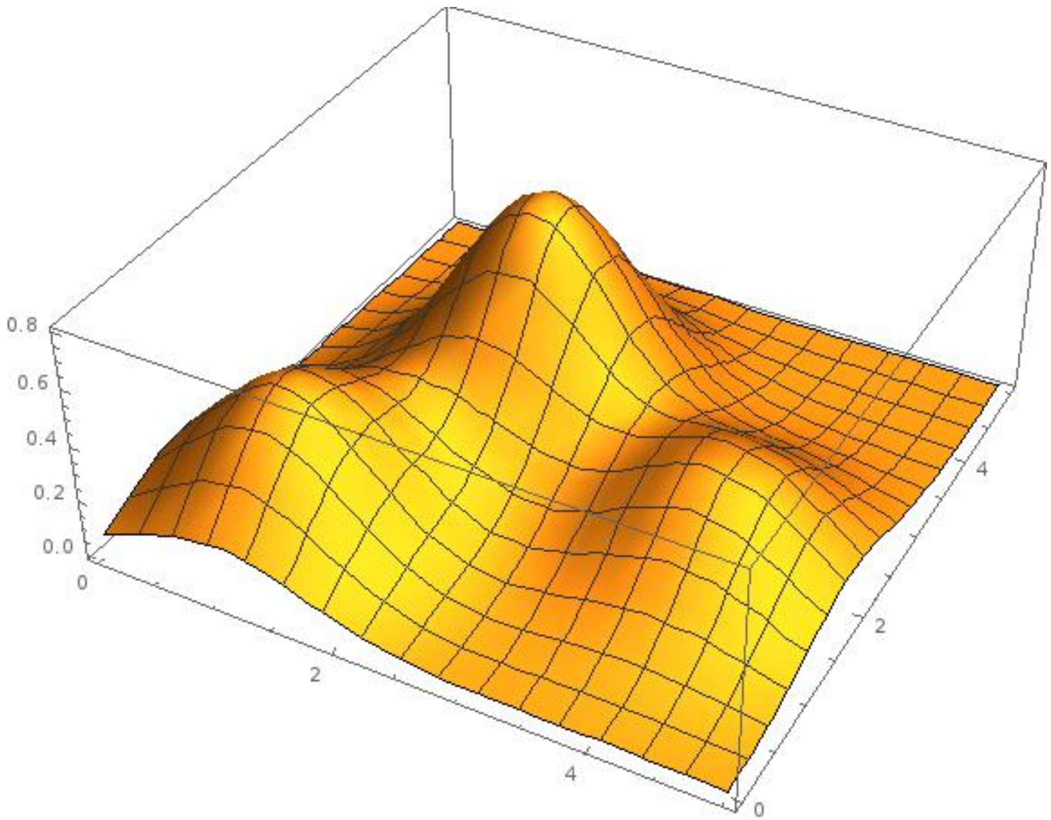


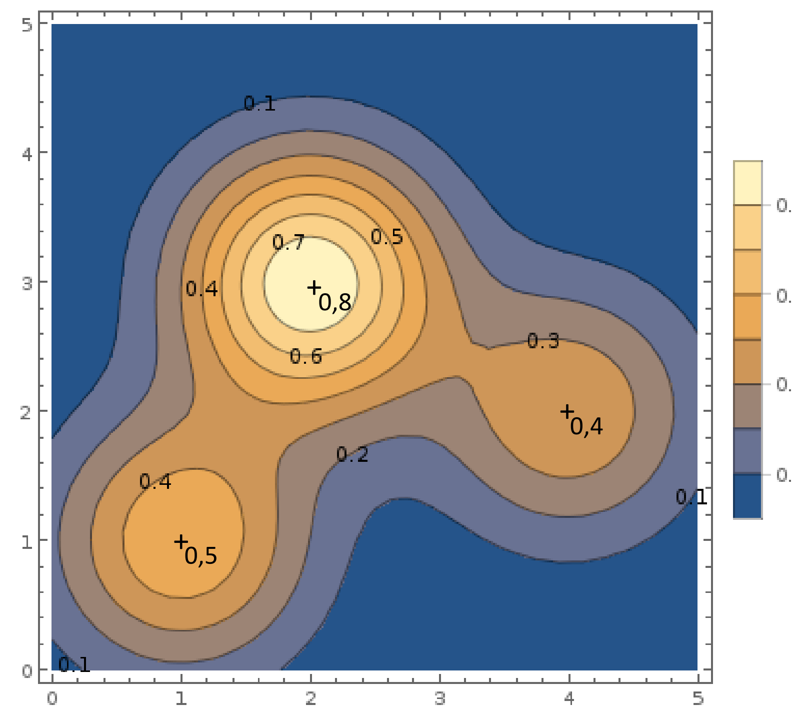
multivis.net/lecture/radial-basis.html

Continuous representation

- 2D Example
  - Data points: $p_1 = (1,1)^T$, $p_2 = (2,3)^T$, $p_3 = (4,2)^T$
  - Data values: $f_1 = 0.5$, $f_2 = 0.8$, $f_3 = 0.4$

$$f(\boldsymbol{x}) = 0.49\,\varphi(\|(1,1)^T - \boldsymbol{x}\|) + 0.79\,\varphi(\|(2,3)^T - \boldsymbol{x}\|) + 0.39\,\varphi(\|(4,2)^T - \boldsymbol{x}\|)$$





Continuous representation

- Drawbacks of radial basis functions
  - Every sample point has influence on whole domain
  - Adding a new sample requires re-solving the equation system
  - Computationally expensive (solving a system of linear equations)

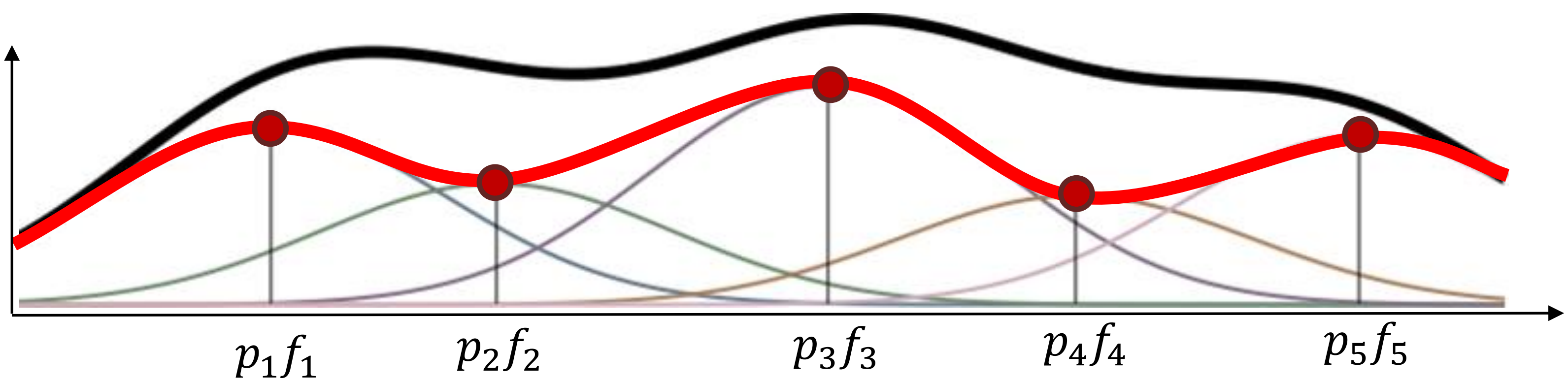
- What can we do?
  - Find a different radial function
  - Give up finding a smooth reconstruction
    - Try finding a piecewise (local) reconstruction function

Continuous representation

SIEMENS
Ingenuity for life

- Radial Basis Functions
  - Instead of the **black** curve we want the **red** one, i.e., the curve which is going through the initial data points
  - This is called an interpolation
  - **Question:** How do we have to select the radial function φ so that the red curve is obtained?

$$f(\boldsymbol{x}) = \sum_i f_i \varphi(\|p_i - \boldsymbol{x}\|)$$



multivis.net/lecture/radial-basis.html

Continuous representation

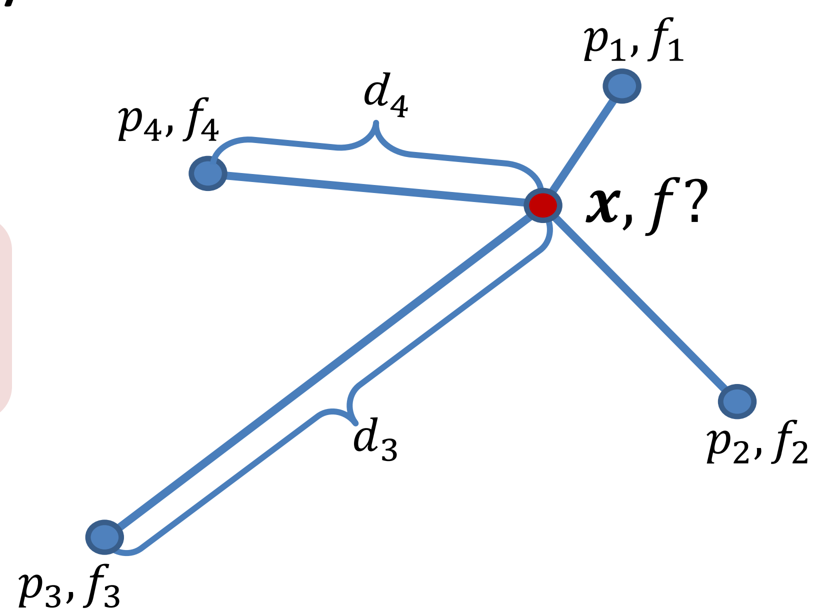
SIEMENS
Ingenuity for life

- Inverse distance weighting
  - Sample positions p_i and values f_i
  - Assumption: Nearby points are more similar than those further away → they have more influence

$$f(\boldsymbol{x}) = \sum_i f_i \, \varphi(\|p_i - \boldsymbol{x}\|)$$

$$d_i = \|p_i - \boldsymbol{x}\|, \qquad \boxed{\varphi(r) = \frac{1}{r^2} \Big/ \sum_{i=1}^{N} \frac{1}{d_i^2}}$$

Attention for $d_i < \epsilon$



multivis.net/lecture/radial-basis.html

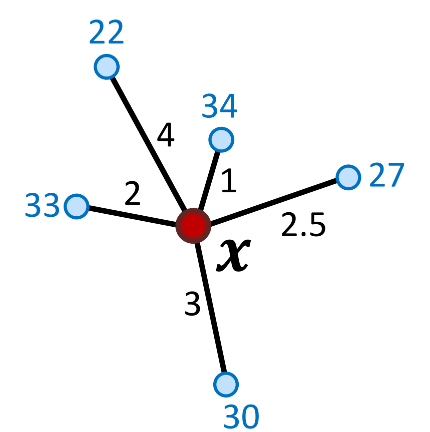
Continuous representation

SIEMENS
Ingenuity for life

- Inverse distance weighting
  - Sample positions p_i and values f_i
  - Assumption: Nearby points are more similar than those further away → they have more influence

$$f(\boldsymbol{x}) = \sum_i f_i \, \varphi(\|p_i - \boldsymbol{x}\|) =$$

$$= \sum_{i=1}^{N} \frac{f_i}{\|p_i - \boldsymbol{x}\|^2} \Big/ \sum_{i=1}^{N} \frac{1}{\|p_i - \boldsymbol{x}\|^2}$$



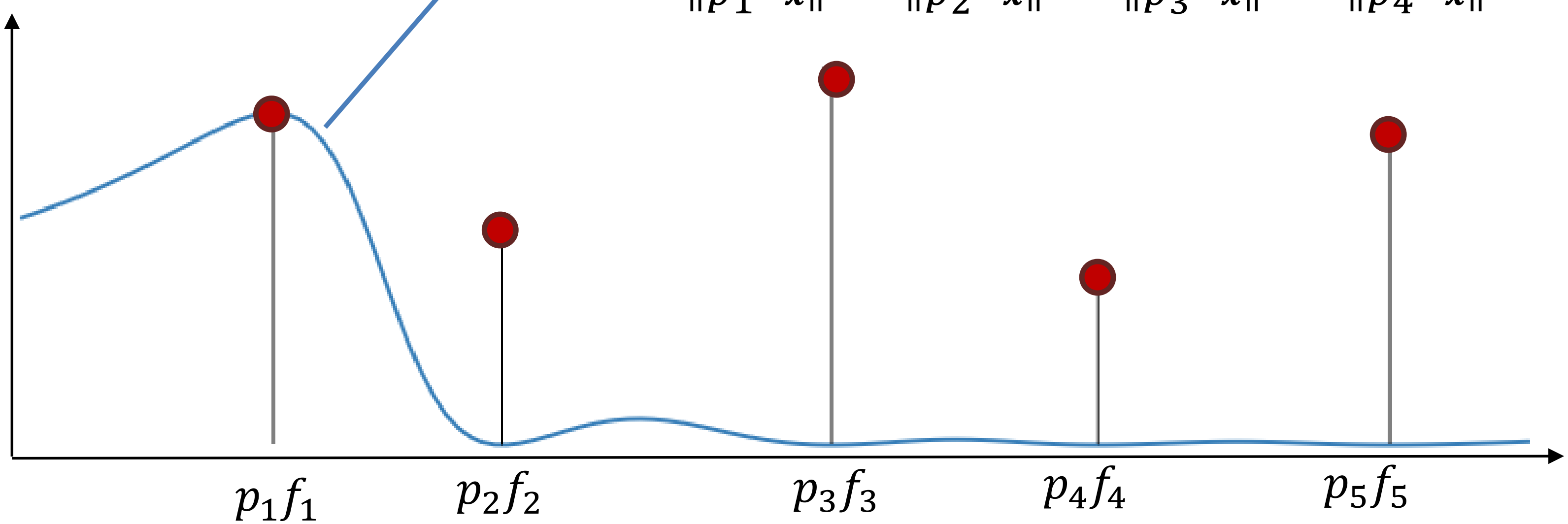
$$f(\boldsymbol{x}) = \frac{\frac{22}{4^2} + \frac{34}{1^2} + \frac{27}{2.5^2} + \frac{30}{3^2} + \frac{33}{2^2}}{\frac{1}{4^2} + \frac{1}{1^2} + \frac{1}{2.5^2} + \frac{1}{3^2} + \frac{1}{2^2}} = 32.38$$

Continuous representation

SIEMENS
Ingenuity for life

- Inverse distance weighting
  - Sample positions p_i and values f_i
  - Assumption: Nearby points are more similar than those further away

$$f_1\,\varphi(\|p_1 - \boldsymbol{x}\|) = \frac{\frac{f_1}{\|p_1-\boldsymbol{x}\|^2}}{\frac{1}{\|p_1-\boldsymbol{x}\|^2}+\frac{1}{\|p_2-\boldsymbol{x}\|^2}+\frac{1}{\|p_3-\boldsymbol{x}\|^2}+\frac{1}{\|p_4-\boldsymbol{x}\|^2}+\frac{1}{\|p_5-\boldsymbol{x}\|^2}}$$

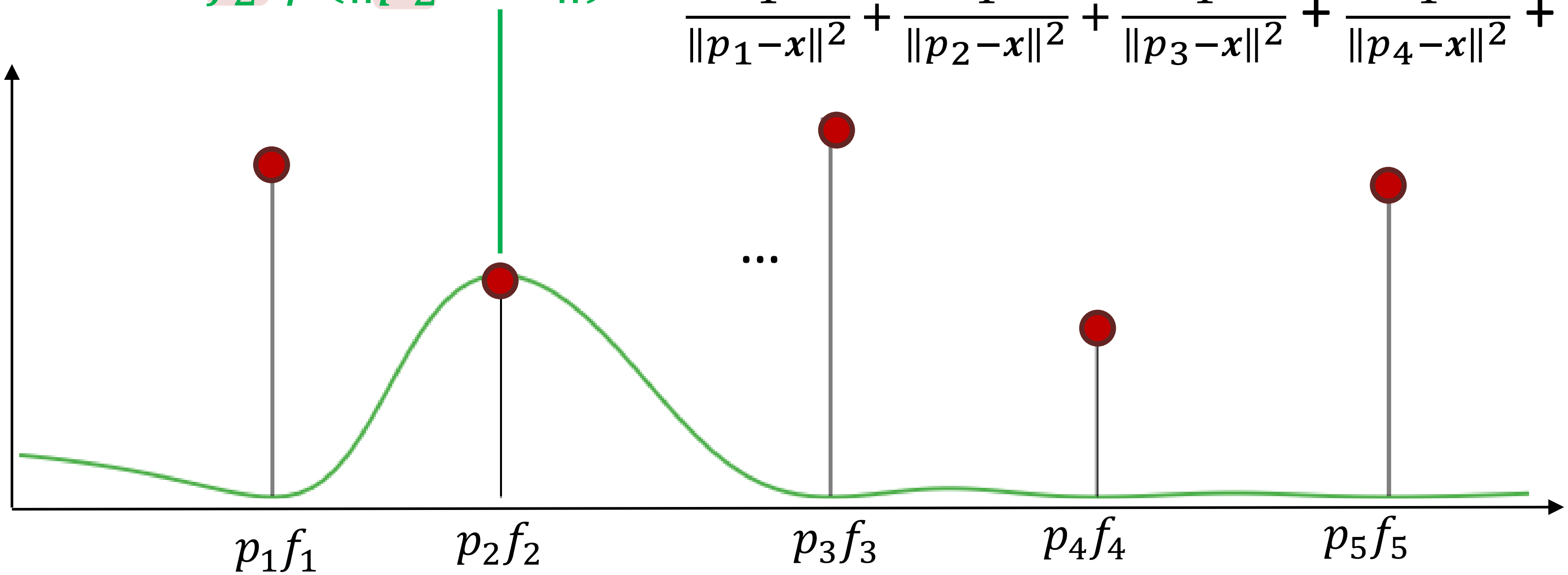


function is zero at $p_i, \quad i \neq 1$

Continuous representation

- Inverse distance weighting
  - Sample positions p_i and values f_i
  - Assumption: Nearby points are more similar than those further away

$$f_2\,\varphi(\|p_2 - \boldsymbol{x}\|) = \frac{\frac{f_2}{\|p_2-\boldsymbol{x}\|^2}}{\frac{1}{\|p_1-\boldsymbol{x}\|^2}+\frac{1}{\|p_2-\boldsymbol{x}\|^2}+\frac{1}{\|p_3-\boldsymbol{x}\|^2}+\frac{1}{\|p_4-\boldsymbol{x}\|^2}+\frac{1}{\|p_5-\boldsymbol{x}\|^2}}$$



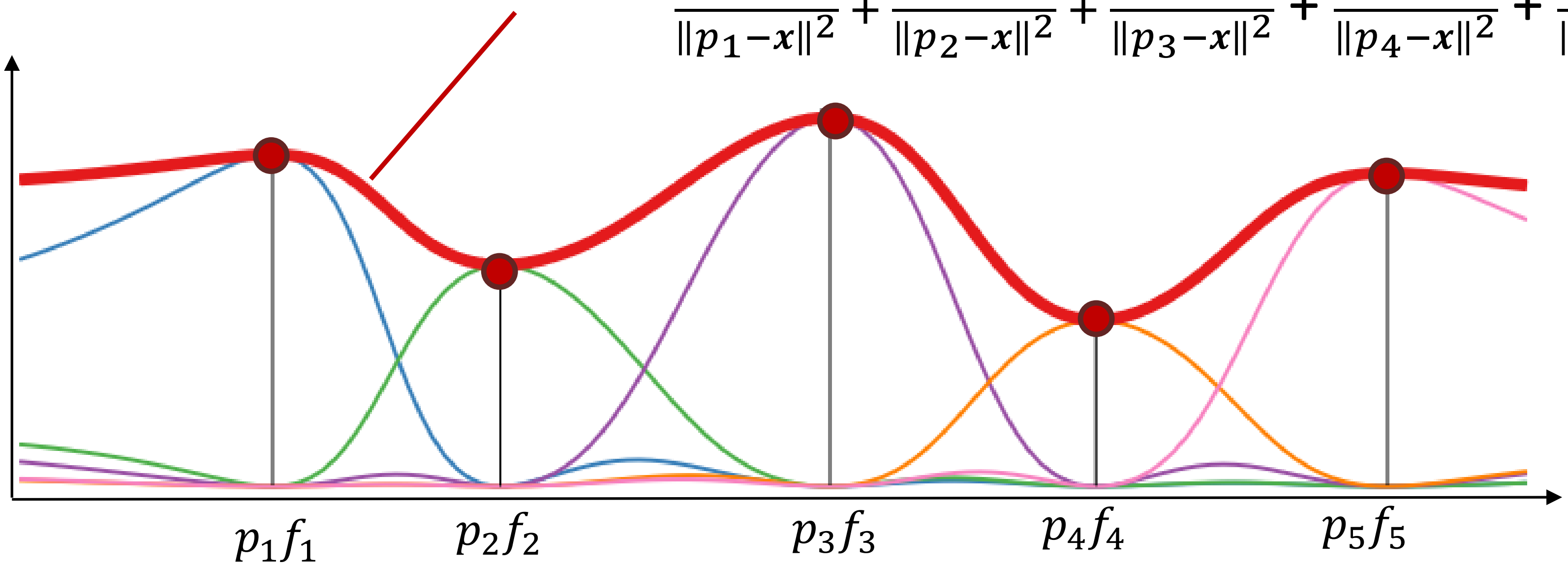
function is zero at p_i, $i \neq 2$

Continuous representation

SIEMENS
Ingenuity for life

- Inverse distance weighting
  - Sample positions p_i and values f_i
  - Assumption: Nearby points are more similar than those further away

$$f(x) = \frac{\frac{f_1}{\|p_1 - x\|^2} + \frac{f_2}{\|p_2 - x\|^2} + \frac{f_3}{\|p_3 - x\|^2} + \frac{f_4}{\|p_4 - x\|^2} + \frac{f_5}{\|p_5 - x\|^2}}{\frac{1}{\|p_1 - x\|^2} + \frac{1}{\|p_2 - x\|^2} + \frac{1}{\|p_3 - x\|^2} + \frac{1}{\|p_4 - x\|^2} + \frac{1}{\|p_5 - x\|^2}}$$



multivis.net/lecture/radial-basis.html

Continuous representation

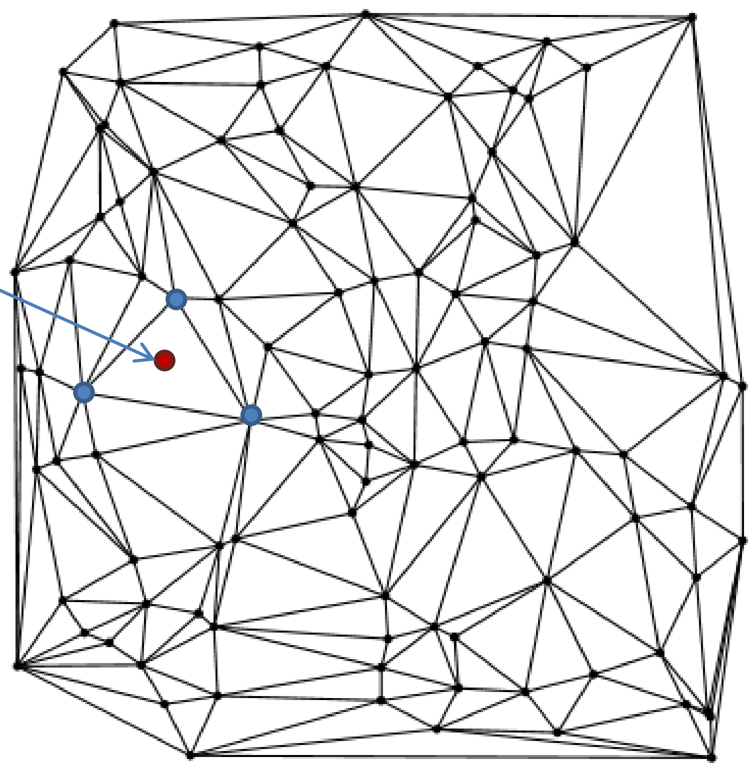
- Inverse distance weighting
  - We no longer have to solve a system of linear equations to find the weights
  - However, every sample point still has global influence

- What can we do?
  - Give up smooth reconstruction by constructing a grid from the given points, i.e., a triangulation

Generating a triangulation

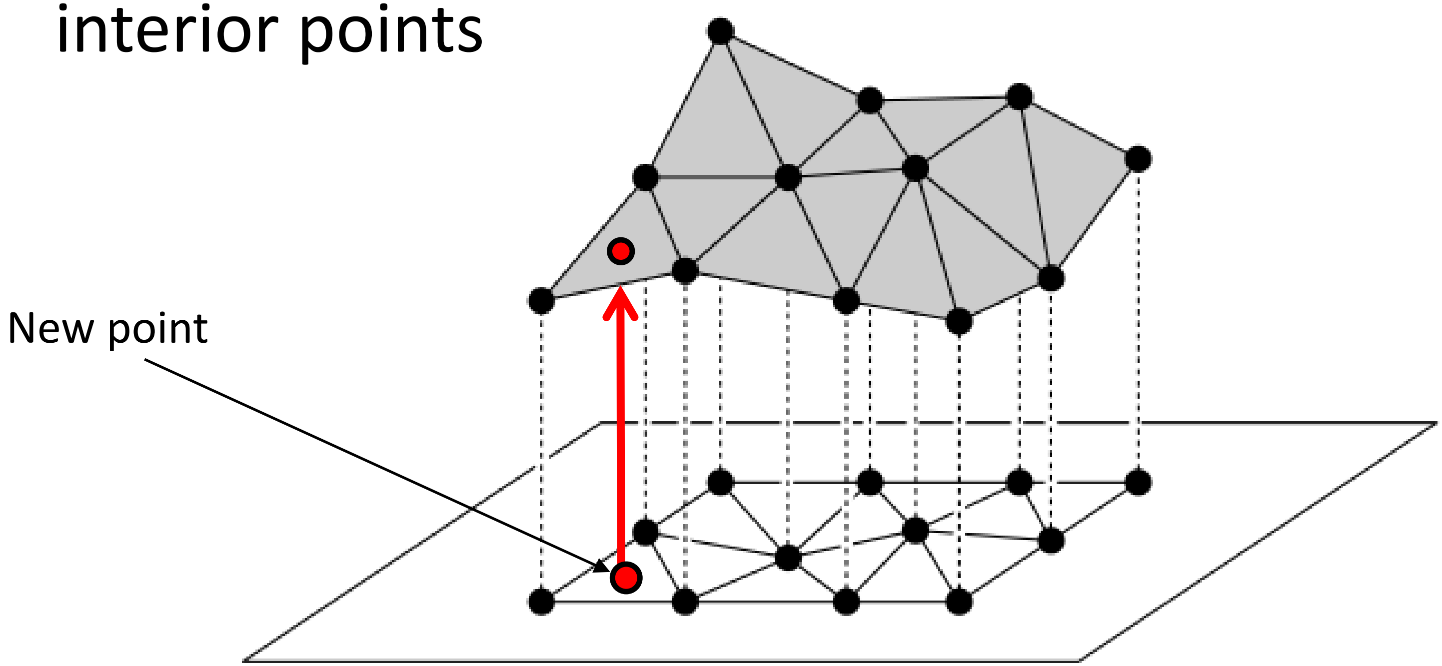
- Try finding a piecewise (local) reconstruction function
  - Connect the points so that a triangulation is obtained
  - Interpolate locally within the triangles

Value obtained by only considering values at triangle corners



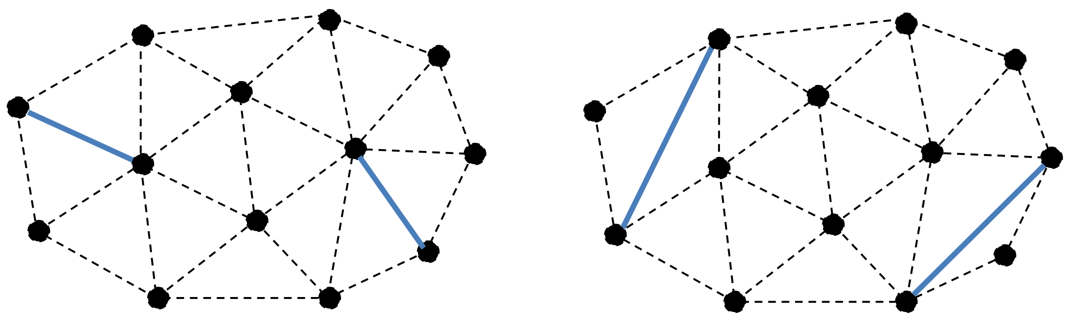
Generating a triangulation

- Once a triangulation is given
  - Let the scalar values at vertices be interpolated across the triangles
  - I.e., piecewise linear interpolation of values at interior points



Generating a triangulation

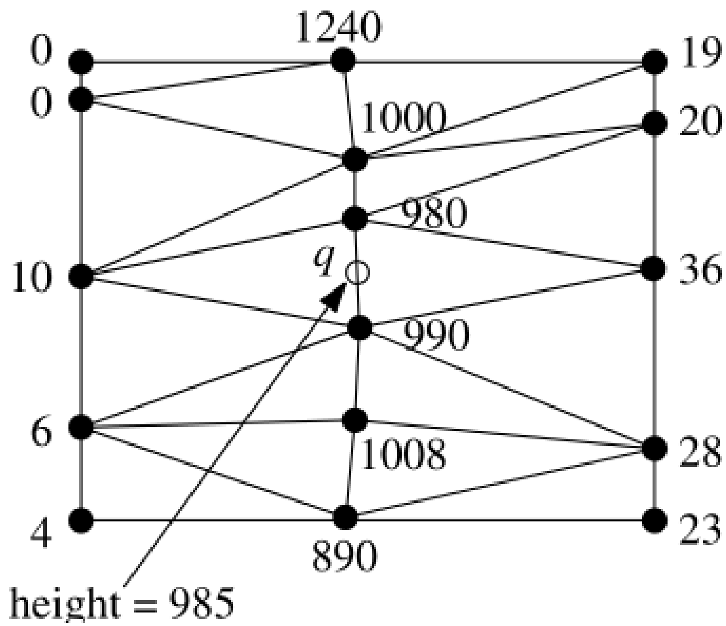
- Given irregularly distributed positions without connectivity information
- For a set of points many triangulations exist
- The challenge is to find the connectivity so that a "good" triangulation is generated

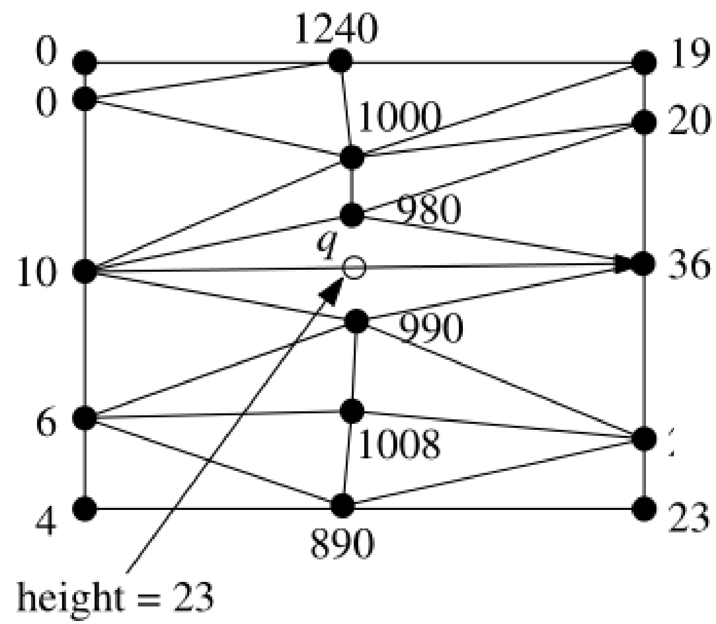


Generating a triangulation

SIEMENS
Ingenuity for life

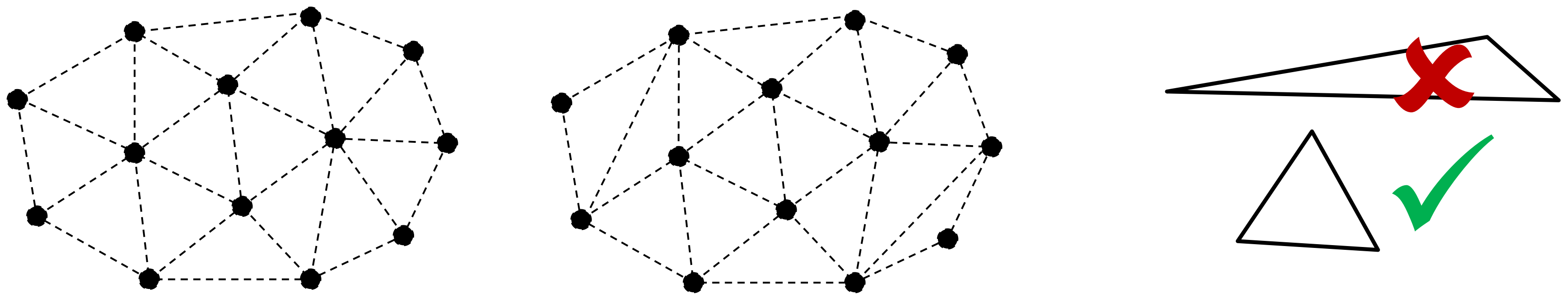
- What is a good triangulation?





Generating a triangulation

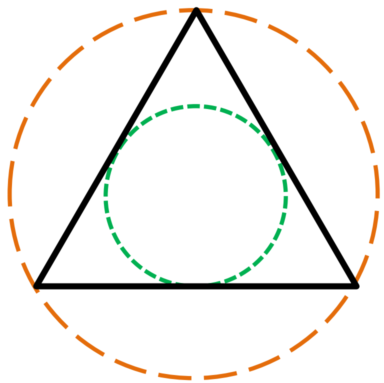
- What is a good triangulation?
  - A measure for the quality of a triangulation is the aspect ratio of the so-defined triangles
  - Avoid long, thin triangles
  - Make triangles as “round” as possible

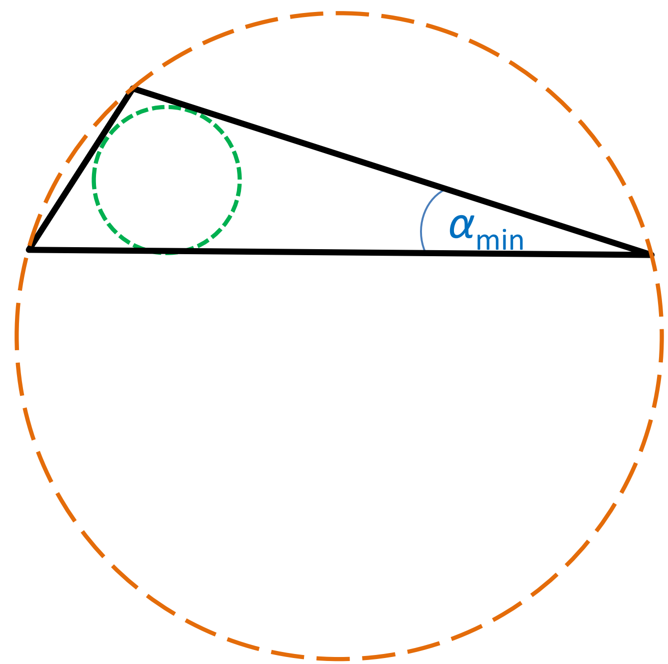


Generating a triangulation

SIEMENS
Ingenuity for life

- An “optimal” triangulation
  - Makes triangles as “round” as possible
  - Maximizes the minimum angle in the triangulation
  - Maximizes $\dfrac{\text{radius of in-circle}}{\text{radius of circumcircle}}$

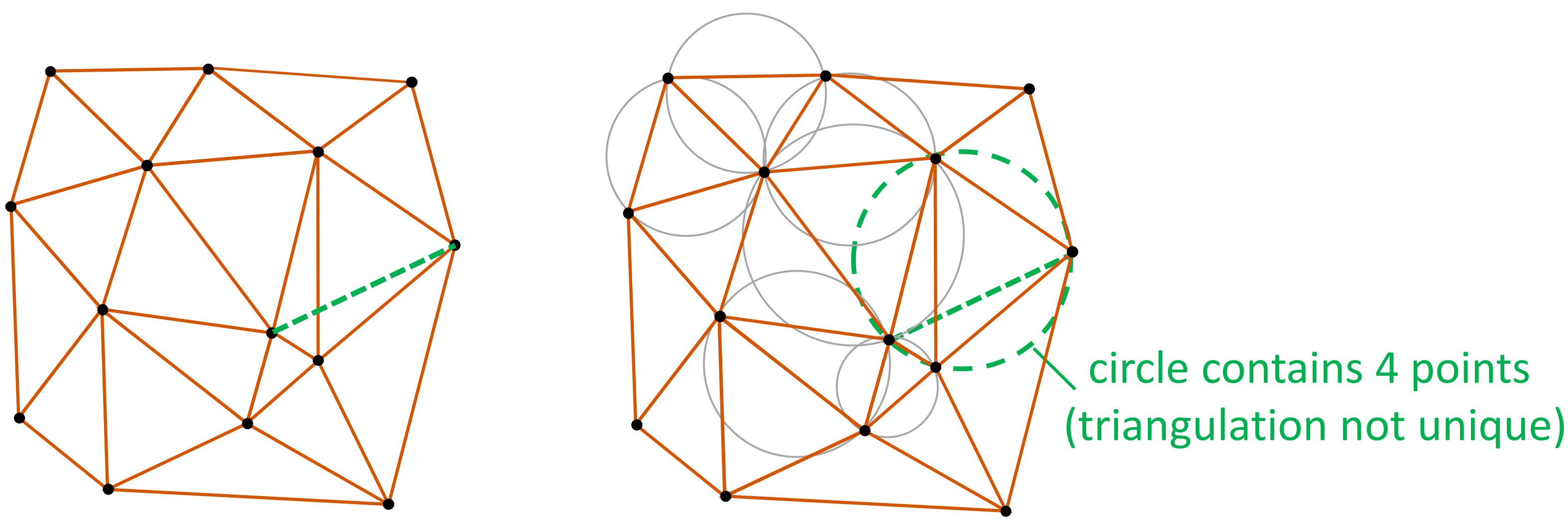




- A Delaunay triangulation is an optimal triangulation

Delaunay triangulation

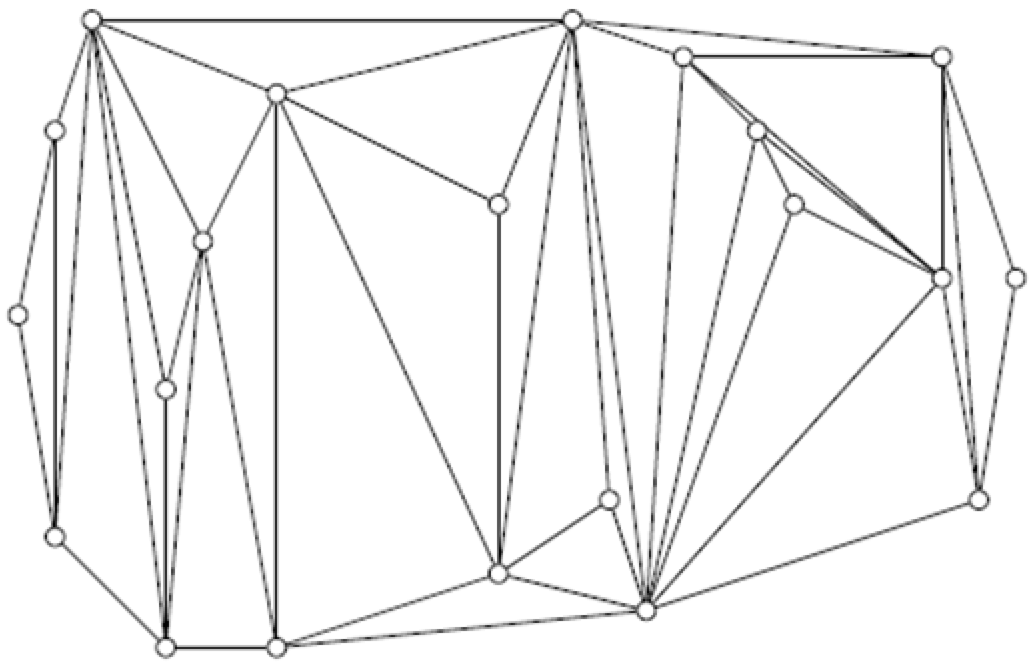
- Delaunay triangulation
  - The circumcircle of any triangle does not contain another point of the set
    - Maximizes the minimum angle in the triangulation
    - Such a triangulation is unique (independent of the order of samples) for all but trivial cases



Delaunay triangulation

SIEMENS
Ingenuity for life

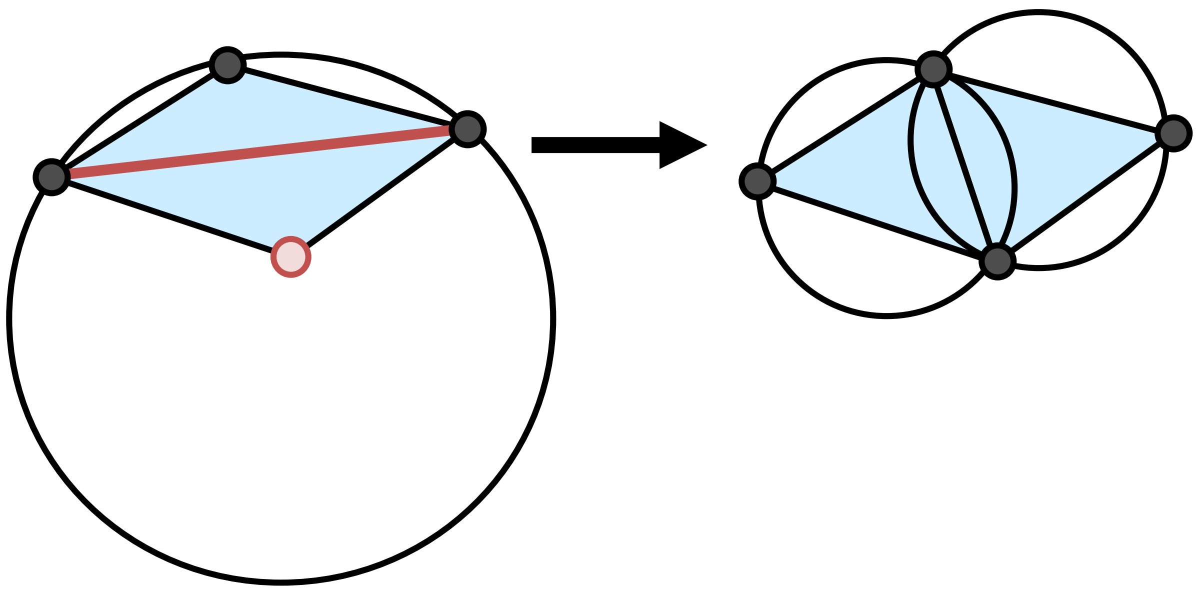
- How to build a Delaunay triangulation from an initial, non-optimal triangulation?
  - Can be performed by successively improving the initial triangulation via local operations



Delaunay triangulation

SIEMENS Ingenuity for life

- Edge flip operation



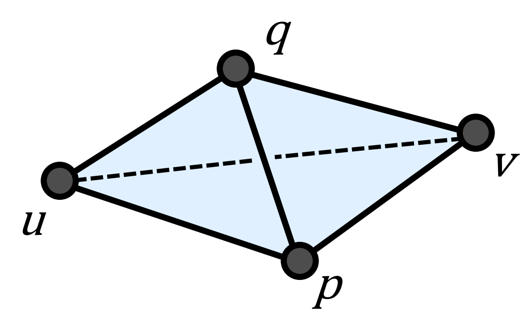
  - An edge is local Delaunay if there exists an empty circumcircle
  - If an edge shared by two triangles is illegal, a flip operation generates a new edge that is legal!

Delaunay triangulation

SIEMENS
Ingenuity for life

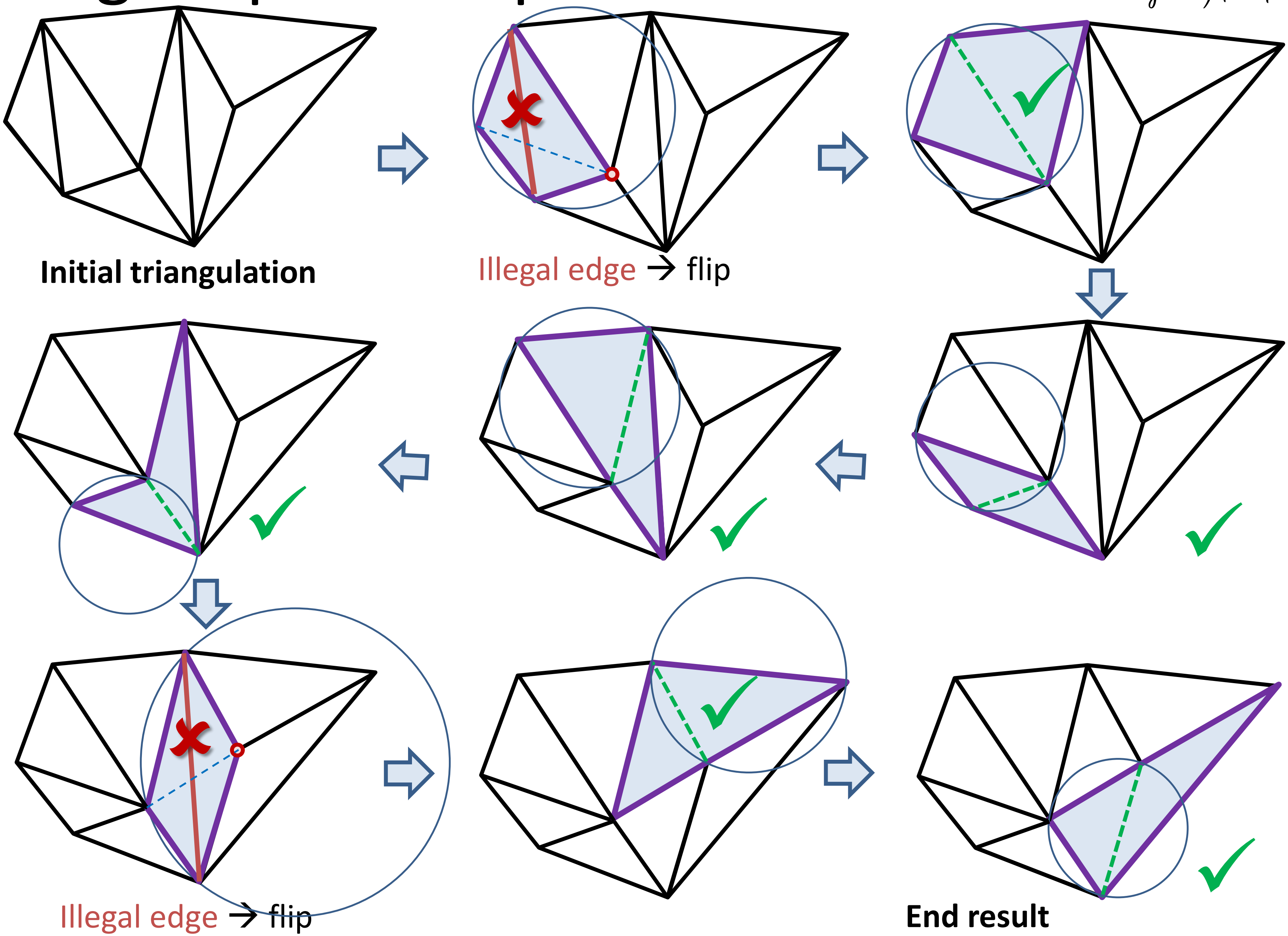
Edge flip algorithm:

```
put all edges in stack and mark them
while stack is non-empty do
  pop edge uv from stack and unmark it
  if uv is illegal then
    substitute pq for uv    //edge flip
    for ab ∈ {up, pv, vq, qu} do
      if ab is unmarked then
        push ab on the stack and mark it
      endif
    endfor
  endif
endwhile
```



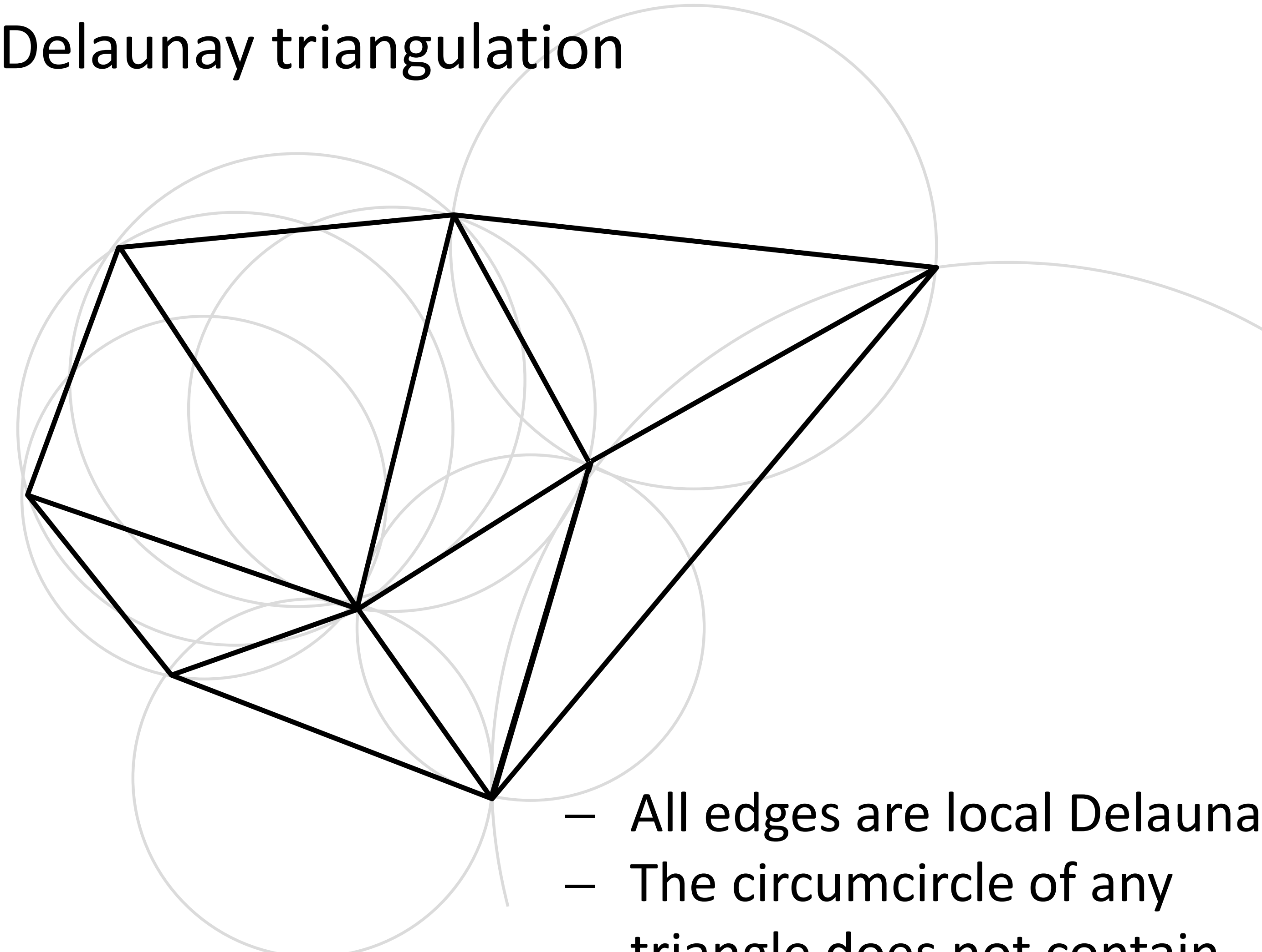
Edge flip - example

SIEMENS
Ingenuity for life



Edge flip - example

- **Result:** Delaunay triangulation

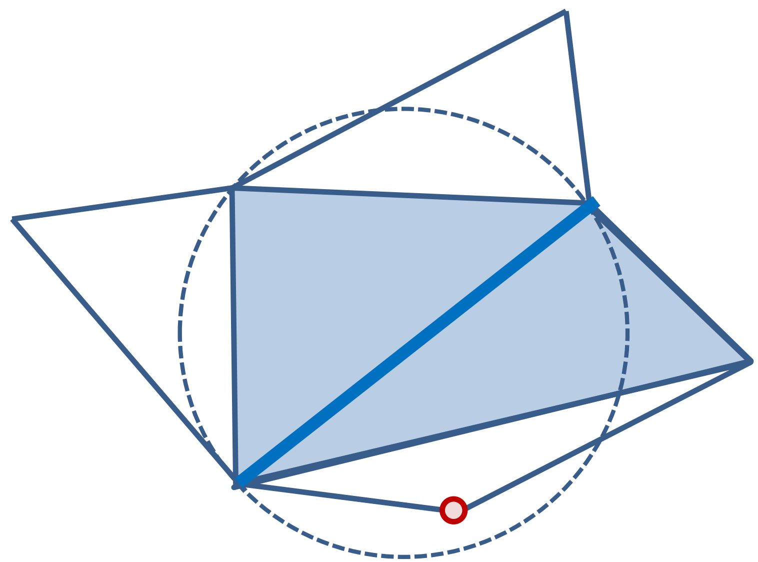


– All edges are local Delaunay
– The circumcircle of any triangle does not contain another point of the set

http://multivis.net/lecture/delaunay.html

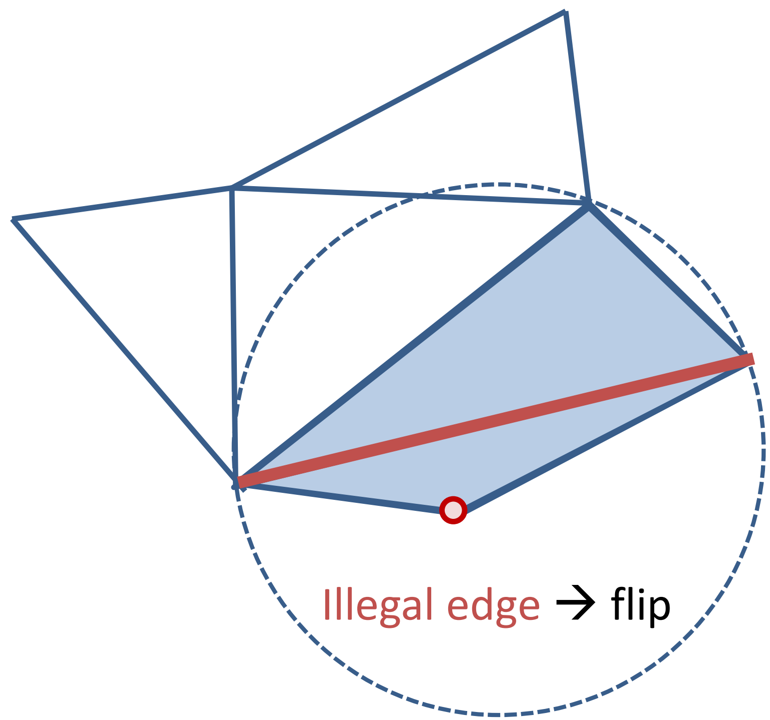
Delaunay triangulation

- Local vs. global optimality
  - Edge is locally Delaunay … but not globally



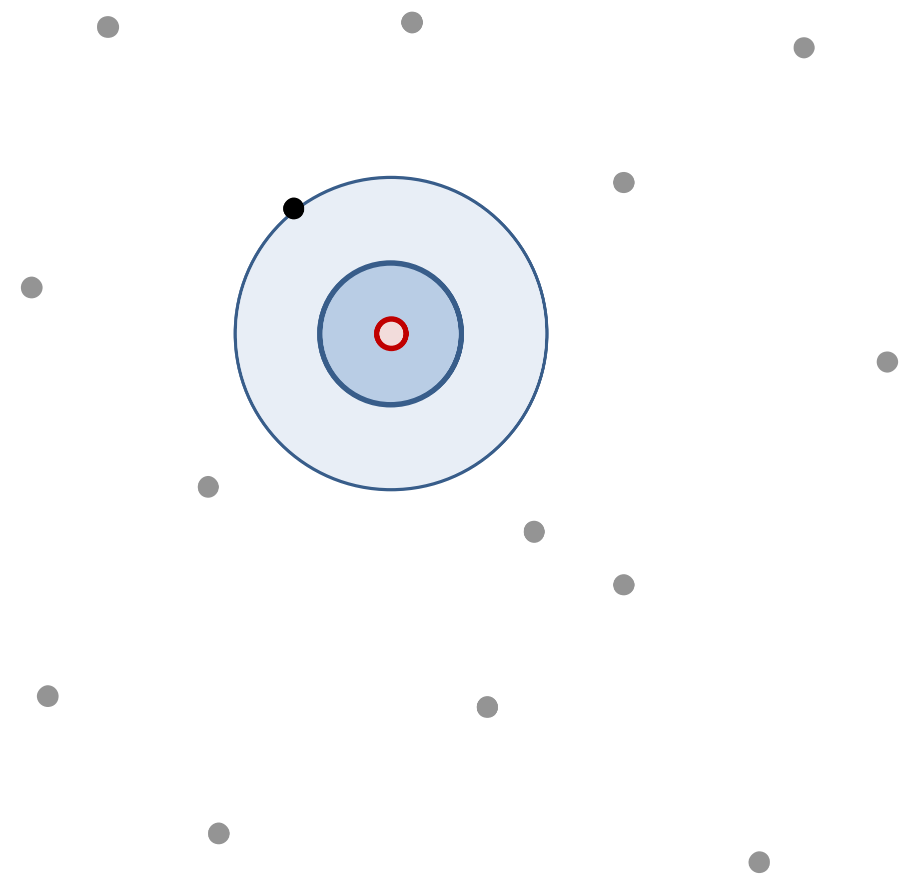
Delaunay triangulation

- Local vs. global optimality
  – If a triangulation is locally Delaunay everywhere → globally Delaunay



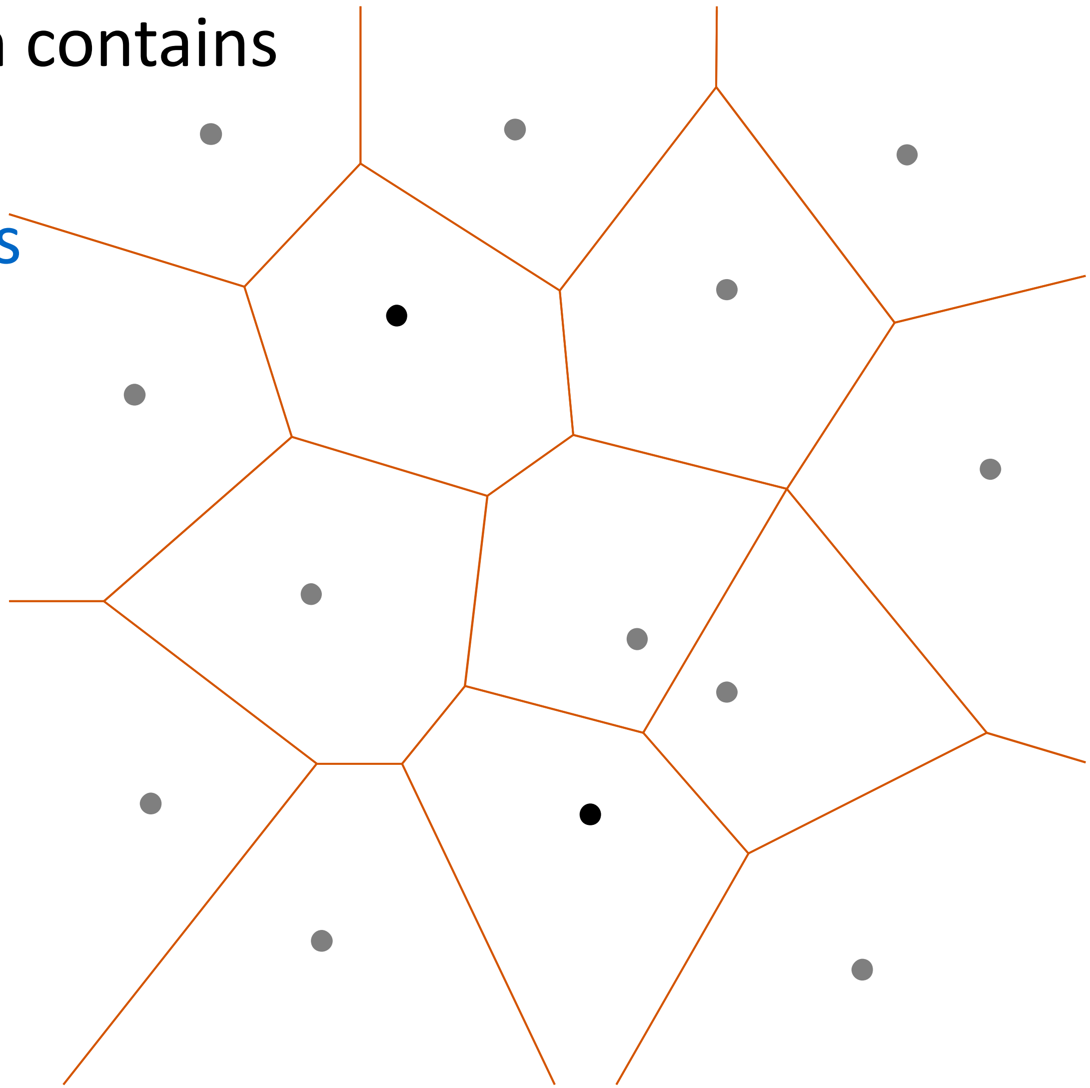
Voronoi diagram

- **Problem:** Looking for nearest neighbor



Voronoi diagram

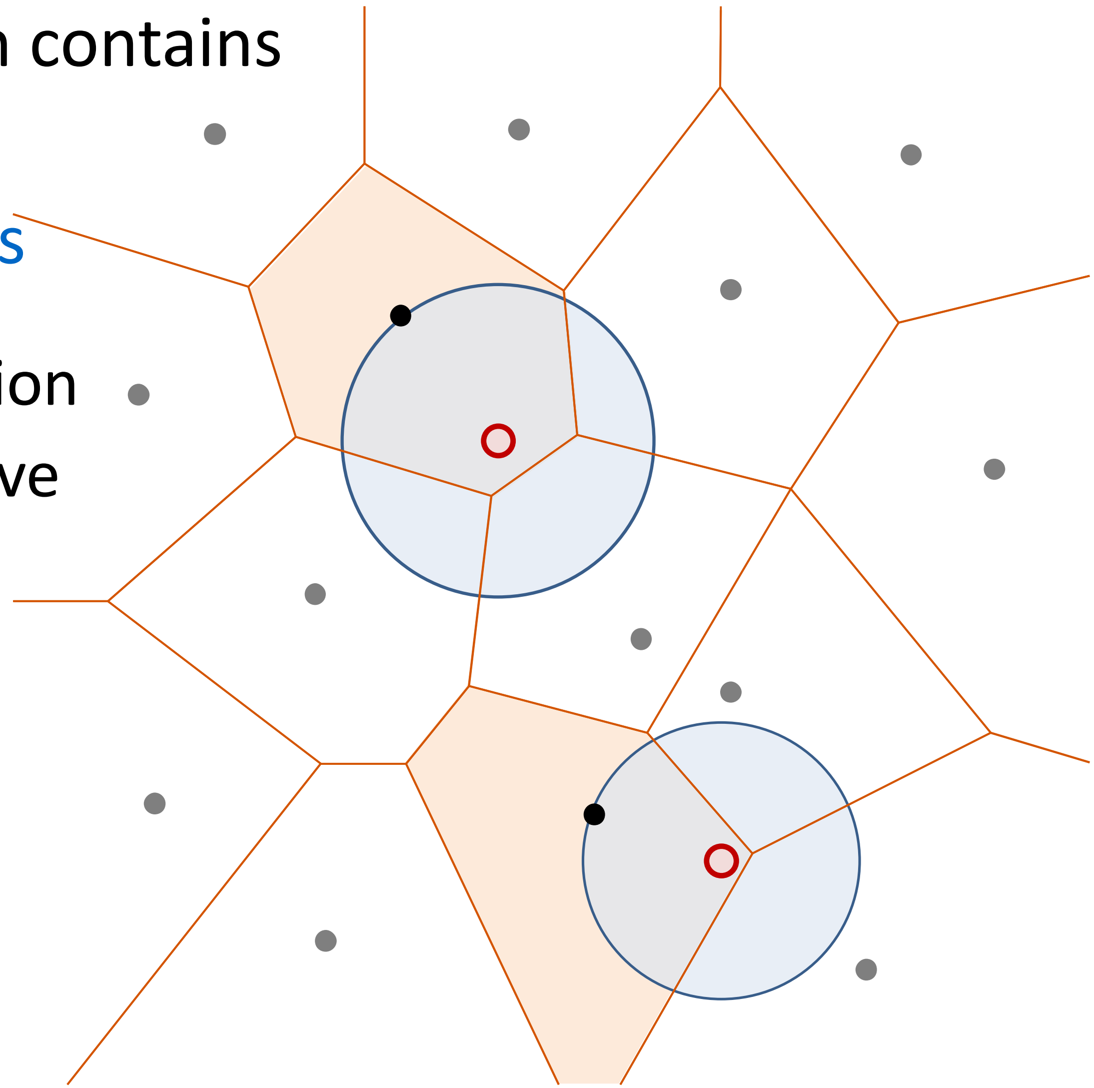
- Partitions domain into Voronoi regions
  - Each Voronoi region contains one initial sample – the Voronoi samples



Voronoi diagram

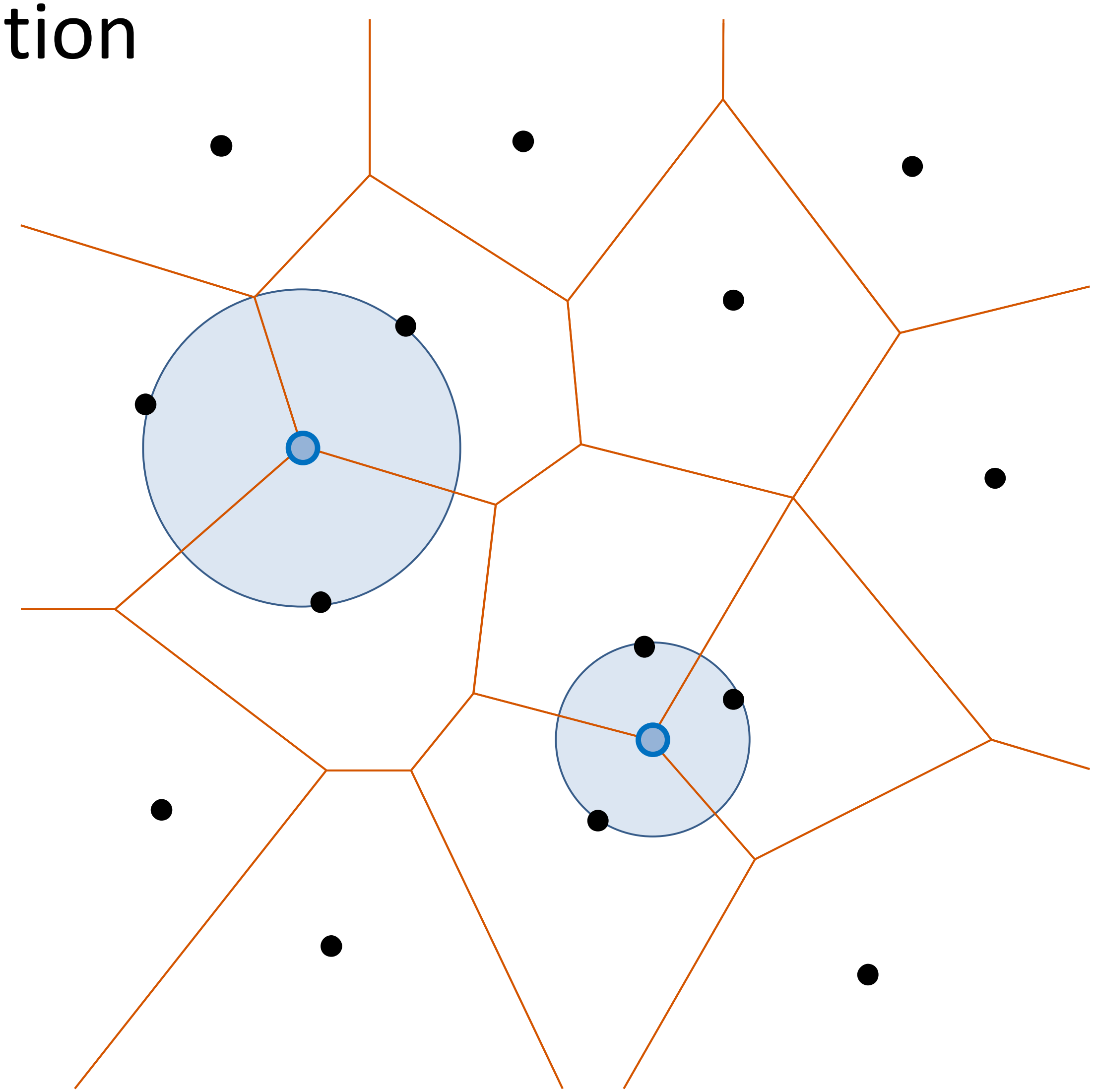
SIEMENS
Ingenuity for life

- Partitions domain into Voronoi regions
  - Each Voronoi region contains one initial sample – the Voronoi samples
  - Points in Voronoi region are closer to respective sample than to any other sample



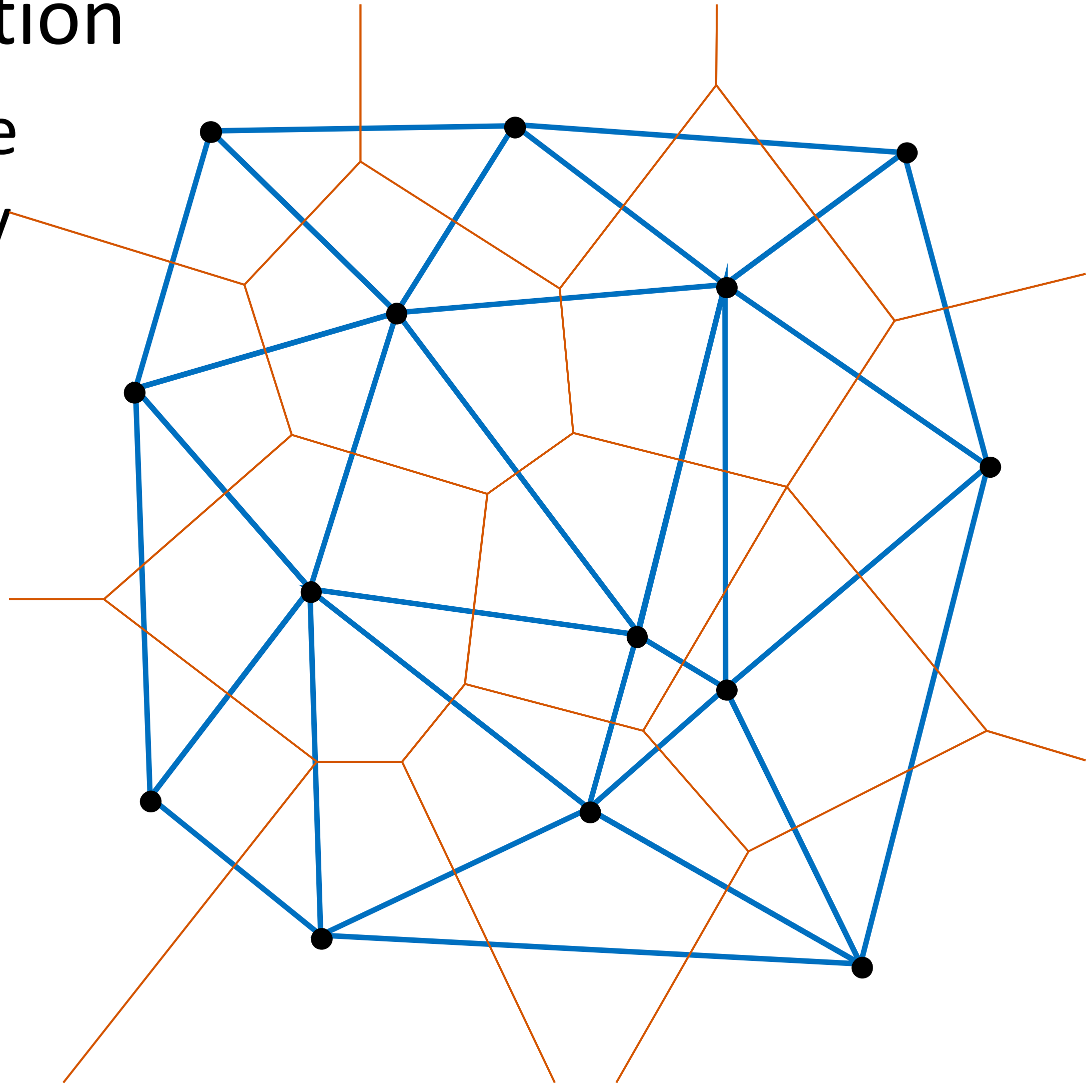
Voronoi diagram

- Centers of circumcircles of Delauney triangulation



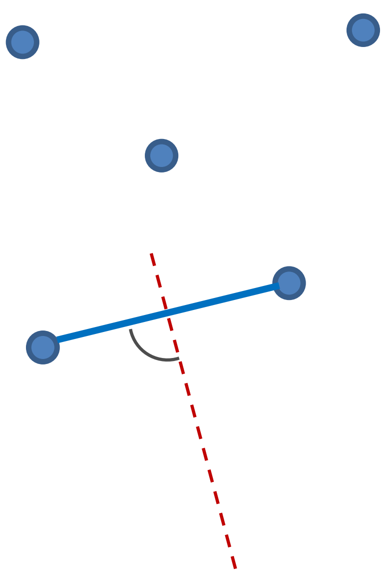
Voronoi diagram

- The geometric dual (topologically equal) of Delaunay triangulation
  - Voronoi samples are vertices in Delauney triangulation



Voronoi diagram

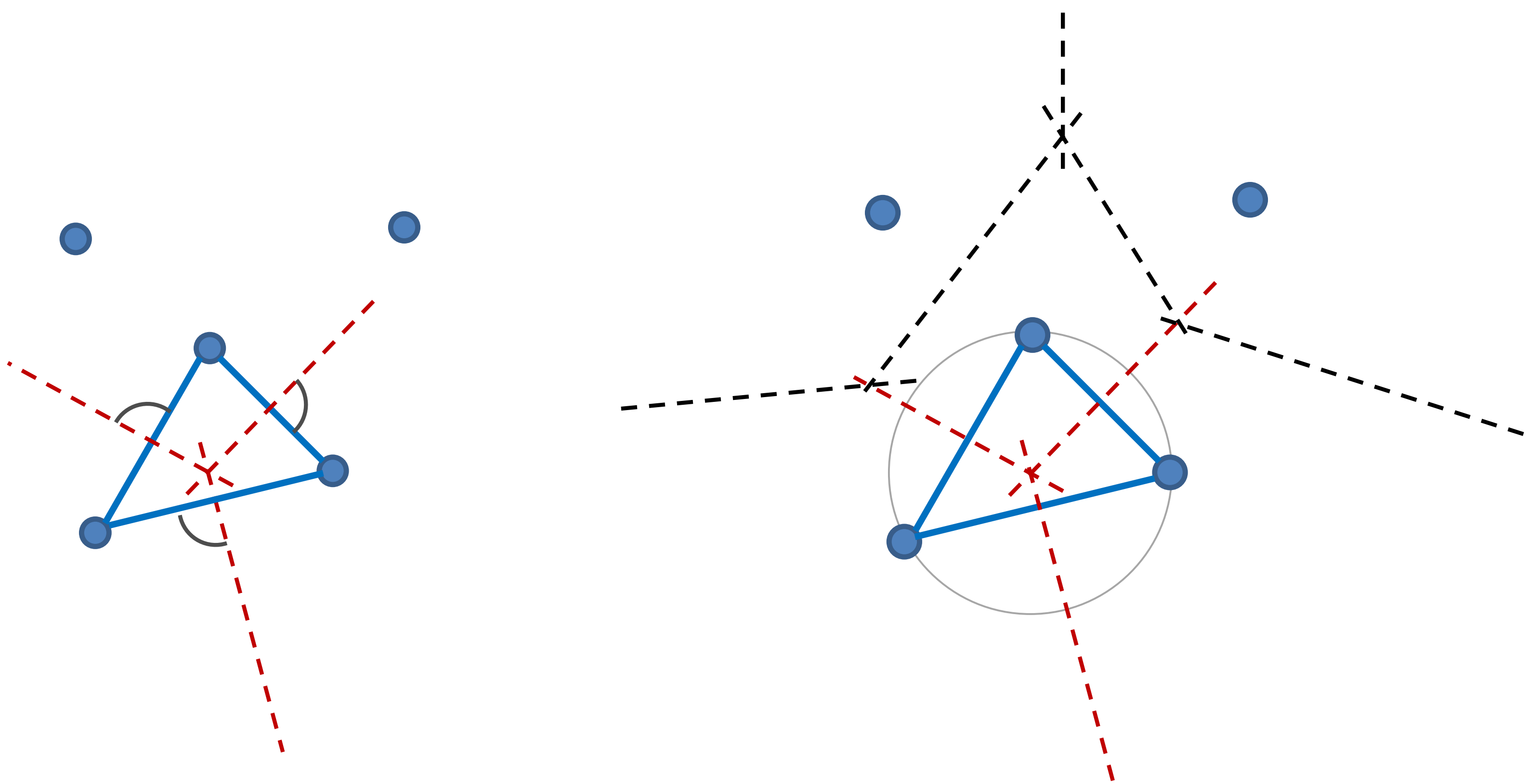
- Construction
  - Points in a Voronoi region are closer to the respective sample than to any other sample



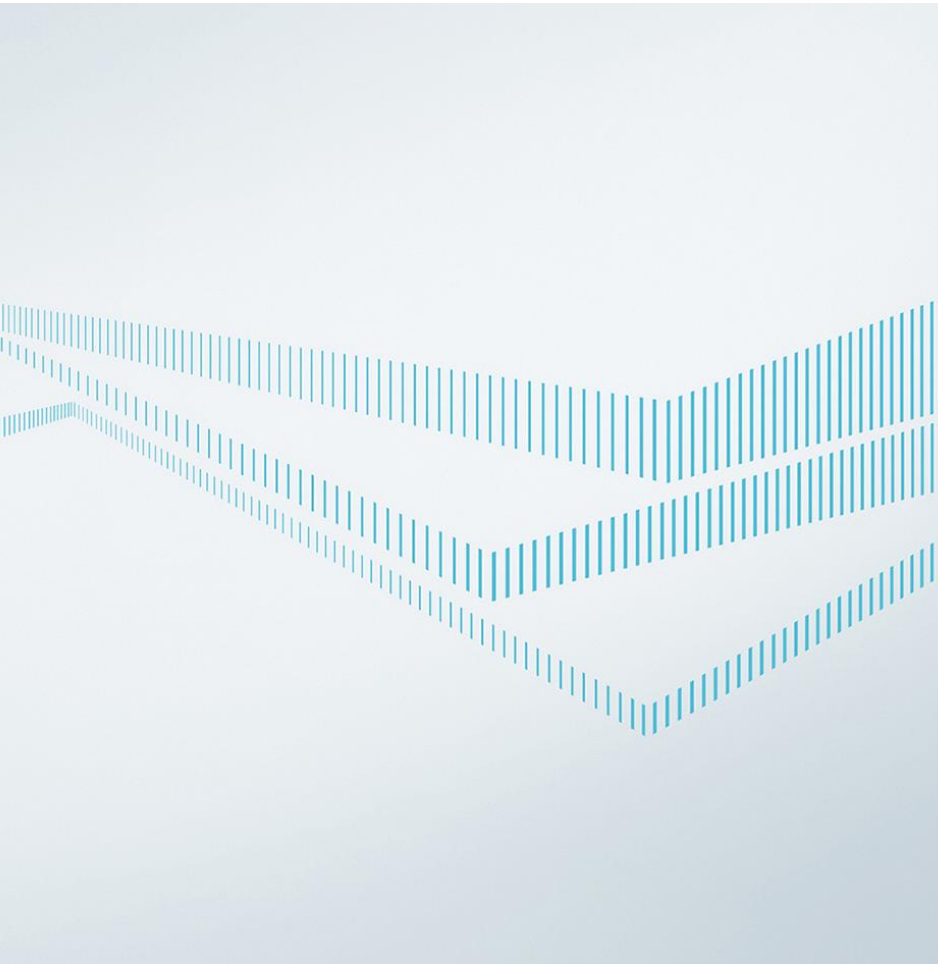
Voronoi diagram

SIEMENS
Ingenuity for life

- Construction
  - Points in a Voronoi region are closer to the respective sample than to any other sample



Contact information



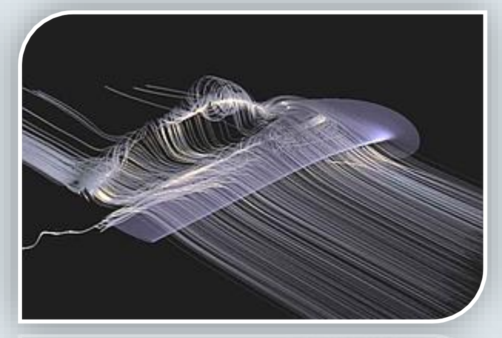
Dr. Johannes Kehrer

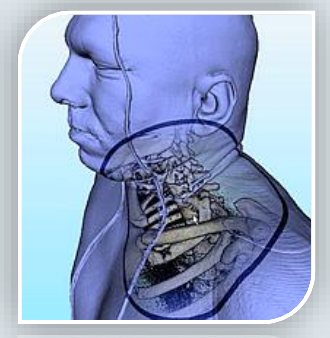
Siemens Technology
T RDA BAM IBI-DE
Otto-Hahn-Ring 6
81739 München, Deutschland

E-mail:
kehrer.johannes@siemens.com

Internet
siemens.com/innovation

Intranet
intranet.ct.siemens.com





Visual Data Analytics

Data Reconstruction and Interpolation II

Dr. Johannes Kehrer

Disclaimer

SIEMENS
Ingenuity for life

These lecture slides and materials are for personal use by students and are intended for educational purposes only. Distribution or reproduction of the material is prohibited.

The views and opinions expressed by any individual during the course of these lecturers are their own and do not necessarily represent the views, opinions, or positions of Siemens or the Technical University of Munich.

Although reasonable efforts have been made to ensure the information in these lecture slides is current and accurate, we do not assume any responsibility for the content of the posted material.

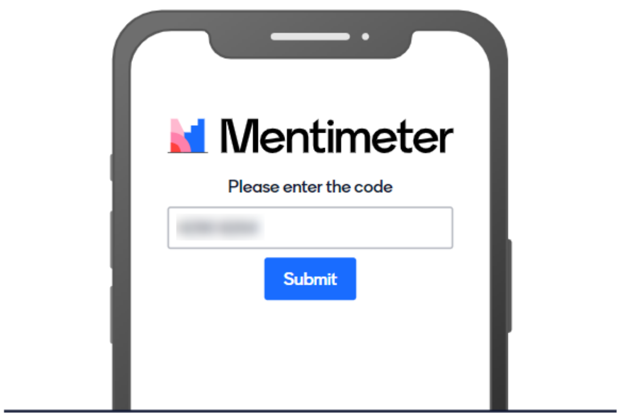
These slides also contain links to websites of third parties. As the content of these websites is not under our control, we cannot assume any liability for such external content.

Any comments/questions?

SIEMENS
Ingenuity for life

Go to

www.menti.com



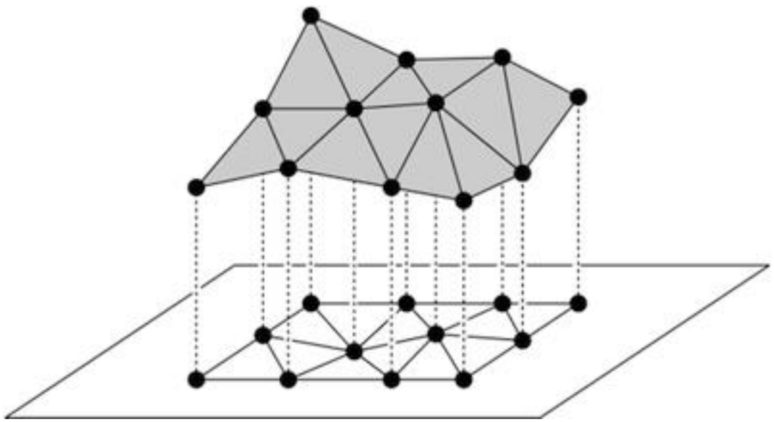
Enter the code

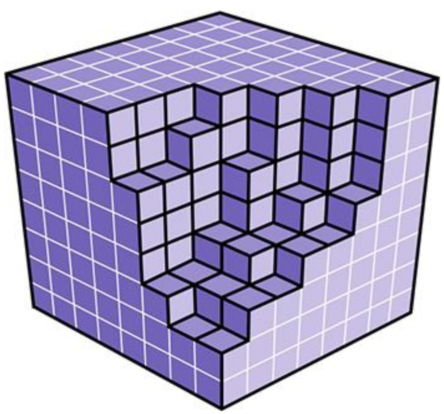
22 07 85 2

Or use QR code

Overview

- Interpolation on grids / meshes
  - Barycentric interpolation
  - Bi-/trilinear interpolation
  - High-quality reconstruction



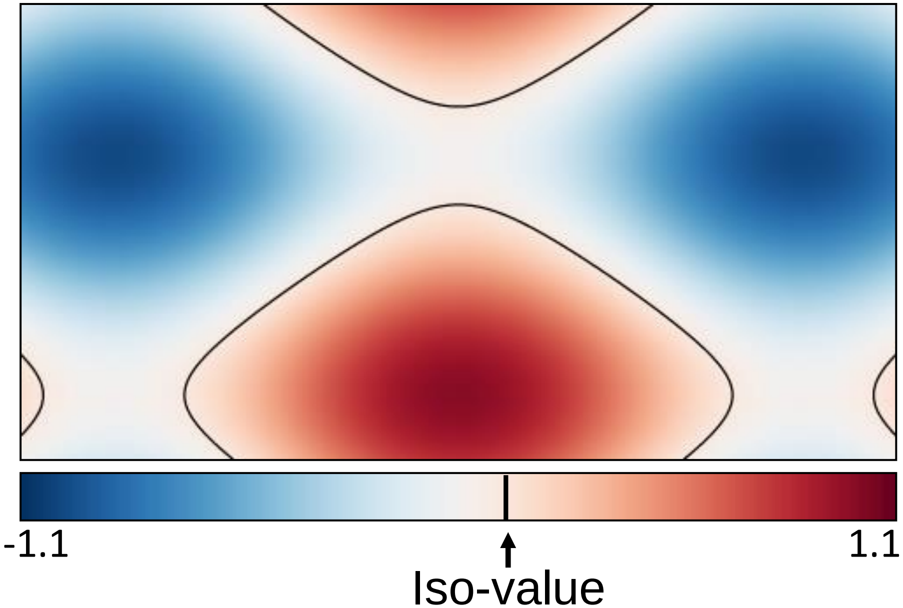


Tricubic vs. trilinear interpolation

Data interpolation

SIEMENS
Ingenuity for life

- Why do we need a continuous representation of data given on grids / meshes
  - Better communication of spatial data distribution
  - Some techniques require a continuous representation (e.g., color mapping, iso-lines)



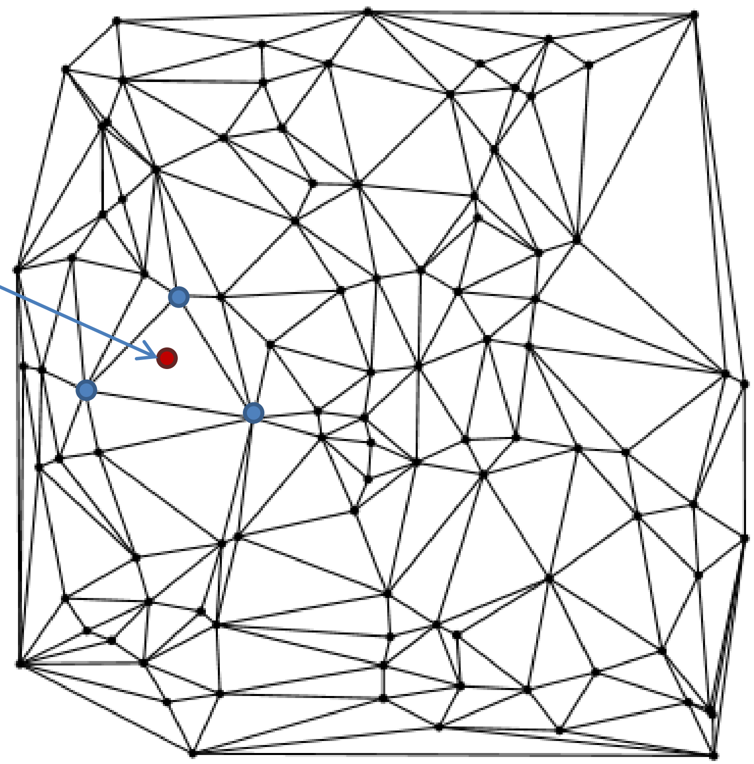
Isoline = all points having a certain value

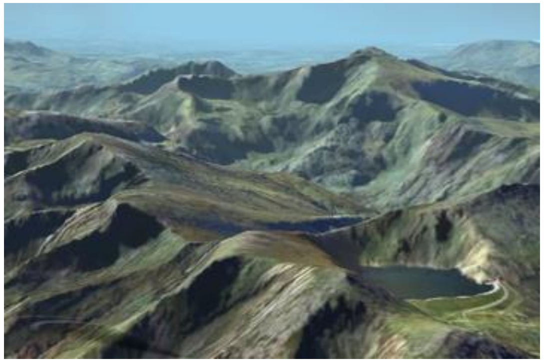
Data interpolation

SIEMENS
Ingenuity for life

- Try finding a piecewise (local) reconstruction function
  – Connect the points so that a triangulation is obtained
  – Interpolate locally within the triangles

Value obtained by only considering values at triangle corners





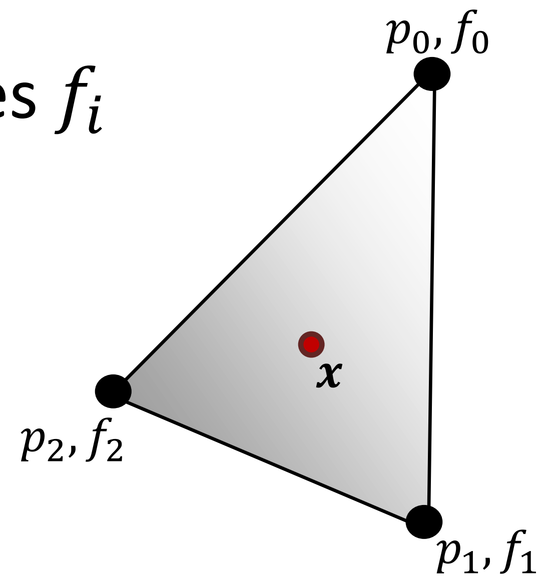
Data interpolation

- How to interpolate inside a triangle
  - The triangle lives in a (N = 2)D plane; it has $N+1$ points (x_i, y_i) with values f_i
  - Can we find a function f that interpolates f_i at the points p_i, *i.e.*,

$$f(p_i) = f_i, \qquad i = 0, \ldots, N$$

(interpolation constraint)

  - If so, then the value at any point $\boldsymbol{x}$ can be interpolated by evaluating $f(\boldsymbol{x})$



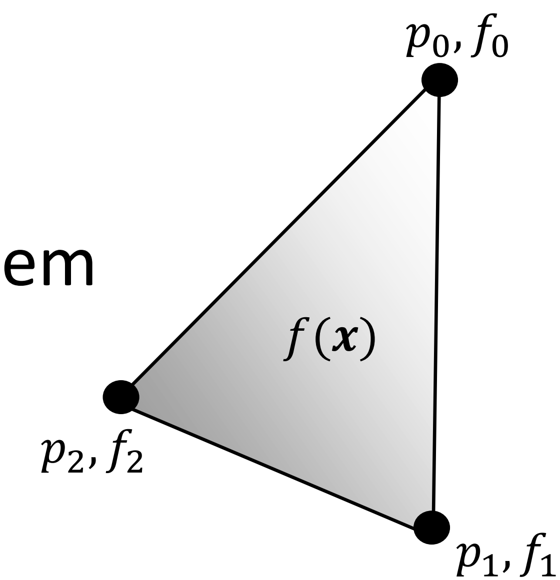
Data interpolation

- There is a unique linear function that satisfies the interpolation constraint
- A linear function can be written as

$$f(\boldsymbol{x}) = a + bx + cy$$

  – The unknown coefficients a, b, c can be obtained by solving the system

$$\begin{array}{l} p_0 \rightarrow \\ p_1 \rightarrow \\ p_2 \rightarrow \end{array} \quad \begin{bmatrix} 1 & x_0 & y_0 \\ 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} f_0 \\ f_1 \\ f_2 \end{bmatrix}$$



Data interpolation

SIEMENS
Ingenuity for life

- Example

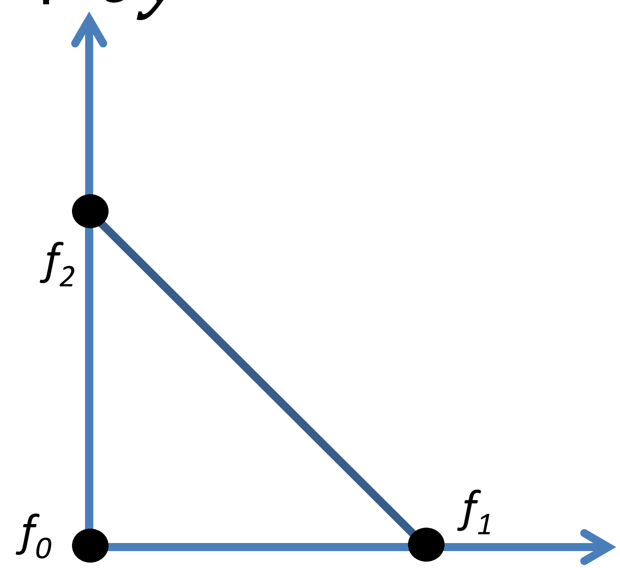
  $p_0 = (0,0),\ p_1 = (1,0),\ p_2 = (0,1)$ with
  $f_0 = 1,\ f_1 = 8,\ f_2 = 2$

- Obtain a, b, c in interpolation function

$$f(\boldsymbol{x}) = a + bx + cy$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} 1 \\ 8 \\ 2 \end{bmatrix}$$

$a = 1$
$b = 7$
$c = 1$



Data interpolation

SIEMENS
Ingenuity for life

- Example

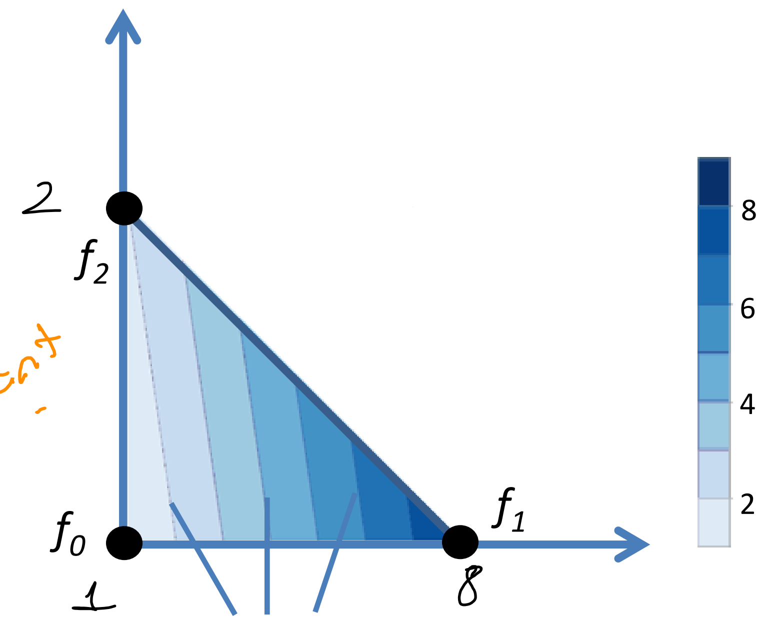
  $p_0 = (0,0),\ p_1 = (1,0),\ p_2 = (0,1)$ with
  $f_0 = 1,\ f_1 = 8,\ f_2 = 2$

- Obtain a, b, c by solving the system

$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} 1 \\ 8 \\ 2 \end{bmatrix}$$

$$a = 1, b = 7, c = 1$$
$$f(x,y) = 1 + 7x + 1y$$

important

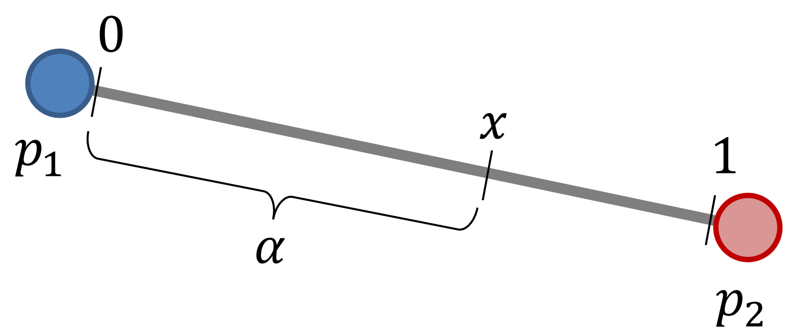


Notice that the isolines are straight lines

Data interpolation

SIEMENS
Ingenuity for life

- Barycentric interpolation
  - Another way to interpolate inside a triangle, which yields the same linear interpolation as before
  - But let's solve a simpler problem first

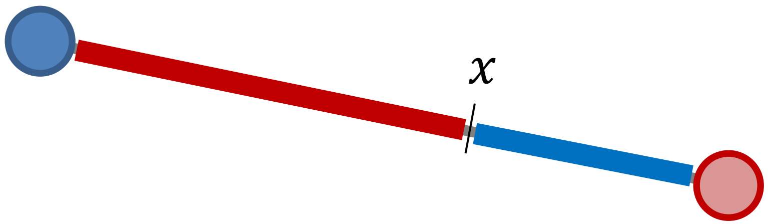


  - We want to define a color for every $\alpha \in [0, 1]$

Data interpolation

SIEMENS
Ingenuity for life

- How do we come up with an equation?



The closer x is to the red point, the more red we want

The closer x is to the blue point, the more blue we want

↓

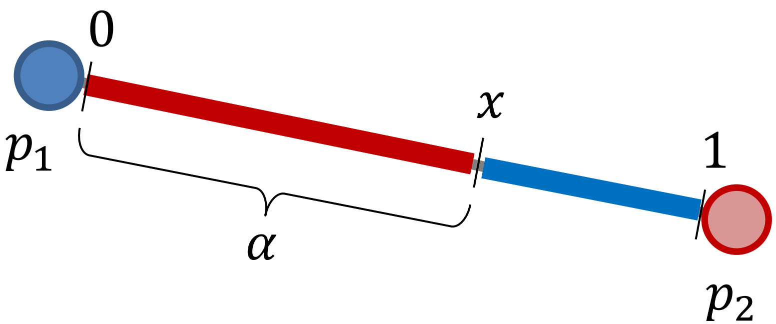
Percentage red = (length of red segment) / (total length)

Percentage blue = (length of blue segment) / (total length)

Data interpolation

SIEMENS
Ingenuity for life

- How do we come up with an equation?



The closer x is to the red point, the more red we want

The closer x is to the blue point, the more blue we want

↓

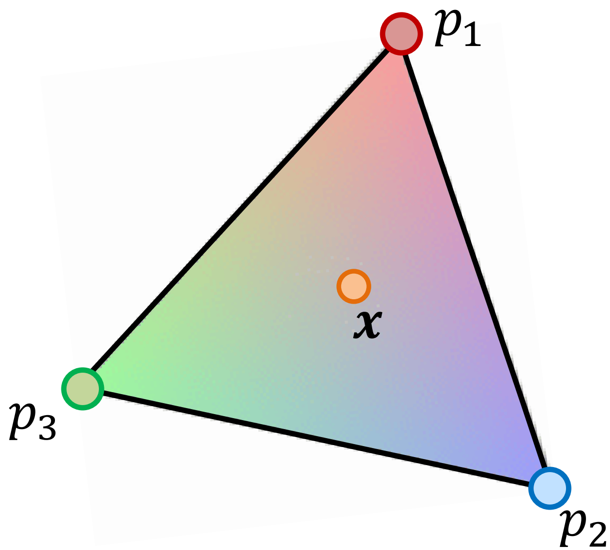
Percentage red $= \alpha$ → Normalized.

Percentage blue $= 1 - \alpha$

$$f(\alpha) = (1 - \alpha) \cdot p_1 + \alpha \cdot p_2$$

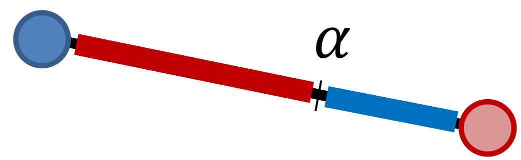
Data interpolation

- Barycentric interpolation
  - Now what about triangles?



What's the interpolated value at the point $\boldsymbol{x}$?

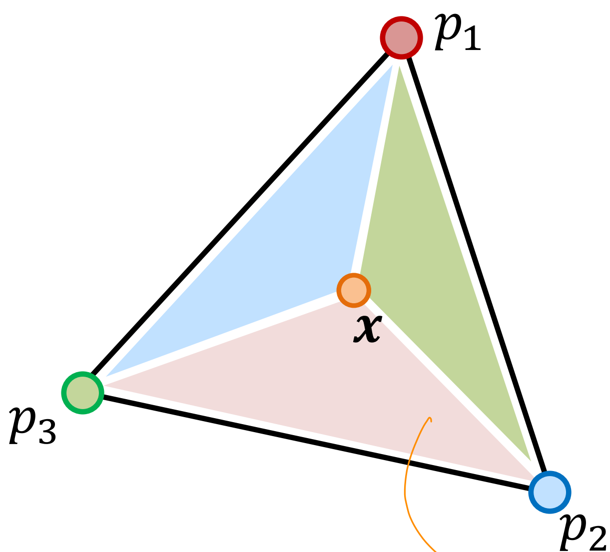
In 1D we used ratios of lengths



Data interpolation

SIEMENS
Ingenuity for life

- Barycentric interpolation
  - Now what about triangles?

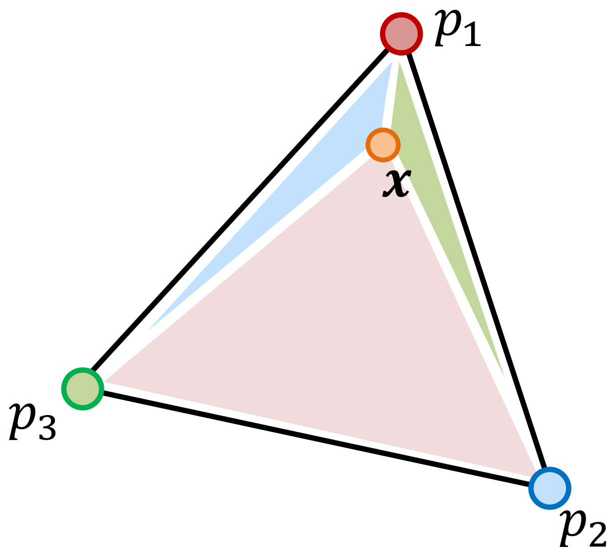


What about ratios of 2D areas?

Sizes of this triangle represent how much red I want.
percentage of red.

Data interpolation

- Barycentric interpolation
  - Now what about triangles?

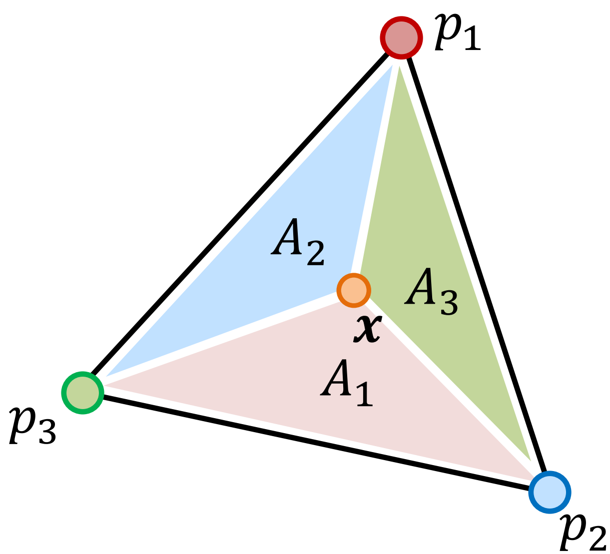


As $\boldsymbol{x}$ approaches the red point, the red area (for example) covers more of the triangle

Data interpolation

SIEMENS
Ingenuity for life

- Barycentric interpolation
  - Just like before:

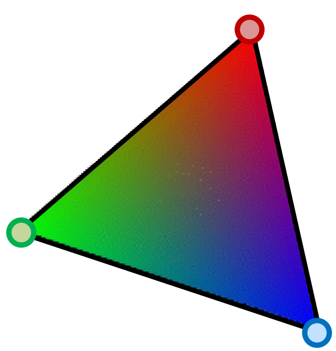


$\alpha_1 = {}^{A_1}/_{A}$ percentage red

$\alpha_2 = {}^{A_2}/_{A}$ percentage blue

$\alpha_3 = {}^{A_3}/_{A}$ percentage green

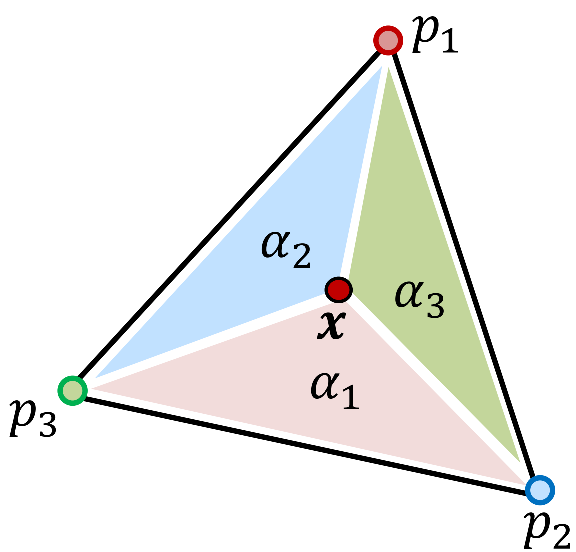
A … area of whole triangle



$$\boldsymbol{x} = \alpha_1 p_1 + \alpha_2 p_2 + \alpha_3 p_3$$

$$\alpha_1 + \alpha_2 + \alpha_3 = 1$$

Data interpolation



$$\alpha_1 = \frac{\text{area}(\Delta \boldsymbol{x} p_2 p_3)}{\text{area}(\nabla p_1 p_2 p_3)}$$

$$\alpha_2 = \frac{\text{area}(\Delta p_1 \boldsymbol{x} p_3)}{\text{area}(\nabla p_1 p_2 p_3)}$$

$$\alpha_3 = \frac{\text{area}(\Delta p_1 p_2 \boldsymbol{x})}{\text{area}(\nabla p_1 p_2 p_3)}$$

Barycentric interpolation

$$\boldsymbol{x} = \alpha_1 p_1 + \alpha_2 p_2 + \alpha_3 p_3$$

$$\alpha_1 + \alpha_2 + \alpha_3 = 1$$

Inside triangle criteria

$$0 \le \alpha_1 \le 1$$

$$0 \le \alpha_2 \le 1$$

$$0 \le \alpha_3 \le 1$$

Data interpolation

SIEMENS
Ingenuity for life

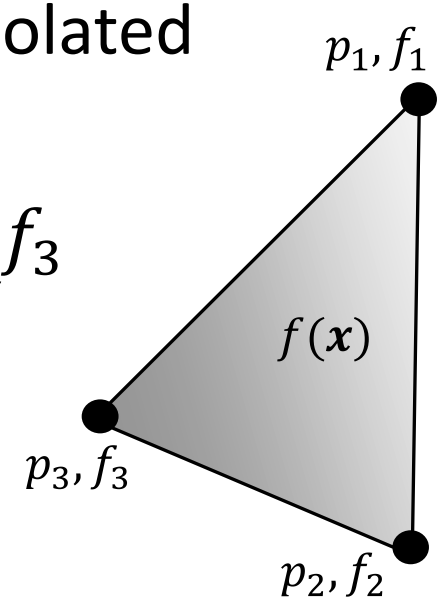
- Barycentric interpolation
  - Every point $\boldsymbol{x}$ in a triangle can be written as a barycentric combination of the vertices p_i:

$$\boldsymbol{x} = \sum_i \alpha_i p_i \quad \text{with} \quad \sum_i \alpha_i = 1$$

(α_i … barycentric coordinates)

  - If α_i are known, then $f(\boldsymbol{x})$ can be interpolated from values f_i at the vertices via

$$f(\boldsymbol{x}) = \alpha_1 f_1 + \alpha_2 f_2 + \underbrace{(1 - \alpha_1 - \alpha_2)}_{\alpha_3} f_3$$



Data interpolation

SIEMENS
Ingenuity for life

Example: Given a triangle with vertices $p_1 = (0.5, 2.5)$, $p_2 = (1.5, 4.5)$ and $p_3 = (2.5, 2.5)$. Compute the barycentric coordinates of the points $P = (1.5, 2.5)$ and $Q = (1.5, 0.5)$ with respect to the triangle.

The area of p2 is 0 because P in the bottom of the triangle, $a_2 = 0$

$a_1 = 0.5$ $a_3 = 0.5$

So for the Q value we need to calculate areas

Firstly, we put x values

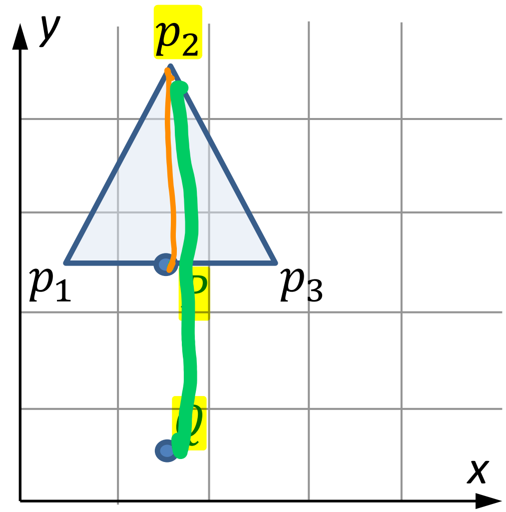
$1,5 = 0,5\, a_1 + 1.5\, a_2 + 2.5\, a_3$ (0.5)

$0,5 = 1,5\, a_1 + 4.5\, a_2 + 2.5\, a_3$

$1 = a_1 + a_2 + a_3$ (2.5)

$-1 = 2a_1 + 3a_2$

$2a_2 = -2$ $a_2 = \boxed{-1}$ $a_1 = 1$ $a_3 = 1$



http://kehrer.bplaced.net/lecture/barycentric.html

Data interpolation

Example: Given a triangle with vertices $p_1 = (0.5, 2.5)$, $p_2 = (1.5, 4.5)$ and $p_3 = (2.5, 2.5)$. Compute the barycentric coordinates of the points $P = (1.5, 2.5)$ and $Q = (1.5, 0.5)$ with respect to the triangle.

Point P:

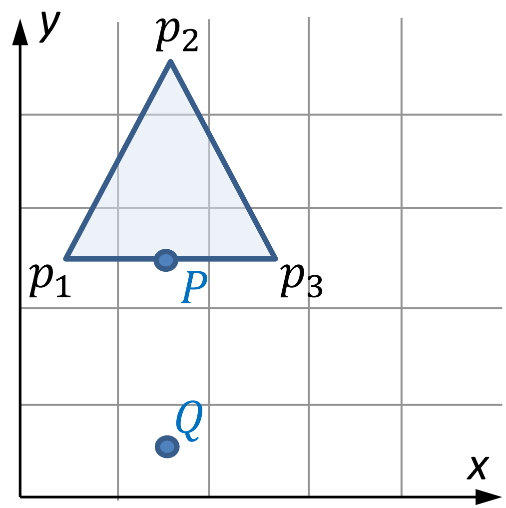
$$\alpha_2 = 0 \;\rightarrow \alpha_1 = \alpha_3 = 0.5$$

Point Q:

$$I\!: \; 1.5 = 0.5\,\alpha_1 + 1.5\,\alpha_2 + 2.5\,\alpha_3 \quad \leftarrow \text{x coordinates}$$
$$II\!: \; 0.5 = 2.5\,\alpha_1 + 4.5\,\alpha_2 + 2.5\,\alpha_3 \quad \leftarrow \text{y coordinates}$$
$$III\!: \; 1 = \alpha_1 + \alpha_2 + \alpha_3$$

$$I - II\!: \; 1 = -2\alpha_1 - 3\alpha_2$$
$$II - 2.5\,III\!: \; -2 = 2\alpha_2 \quad \rightarrow \alpha_2 = -1$$

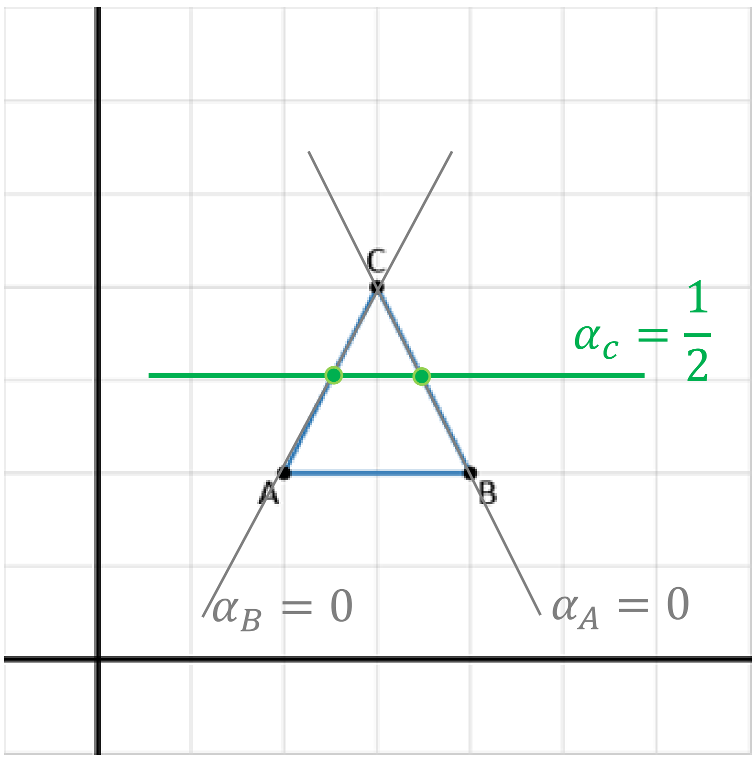
$$\alpha_2 \rightarrow I'\!: \; -2 = -2\alpha_1 \quad \rightarrow \alpha_1 = 1 \quad \rightarrow \alpha_3 = 1$$



http://kehrer.bplaced.net/lecture/barycentric.html

Barycentric coordinates

Example: Given a triangle with vertices (2, 2), (4, 2), and (3, 4). Draw the iso-contours along which the barycentric coordinate α_c corresponding to point C has the value $\frac{1}{2}$, $-\frac{1}{2}$, $\frac{3}{2}$, respectively.



$$\alpha_A + \alpha_B + \alpha_c = 1$$

$$\alpha_c = \frac{1}{2}, \qquad \alpha_A = 0 \rightarrow \alpha_B = \frac{1}{2}$$

$$\alpha_B = 0 \rightarrow \alpha_A = \frac{1}{2}$$

So we assume that $\alpha_A = 0$, the $\alpha_B = 1/2$

For finding exact cordinate,

$x = 4 \times \frac{1}{2} + 3 \cdot \frac{1}{2} = 3.5$

$y = 2 \cdot \frac{1}{2} + 4 \cdot \frac{1}{2} = 3$

Barycentric coordinates

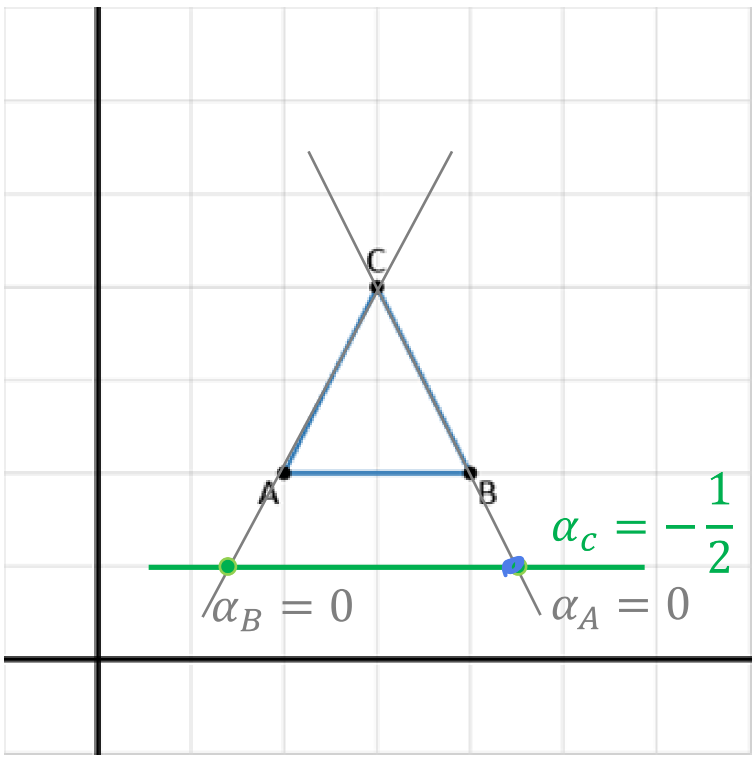
SIEMENS
Ingenuity for life

Example: Given a triangle with vertices (2, 2), (4, 2), and (3, 4). Draw the iso-contours along which the barycentric coordinate α_c corresponding to point C has the value $\frac{1}{2}$, $-\frac{1}{2}$, $\frac{3}{2}$, respectively.

$$\alpha_A + \alpha_B + \alpha_c = 1$$

$$\alpha_c = -\frac{1}{2}, \qquad \alpha_A = 0 \rightarrow \alpha_B = \frac{3}{2}$$

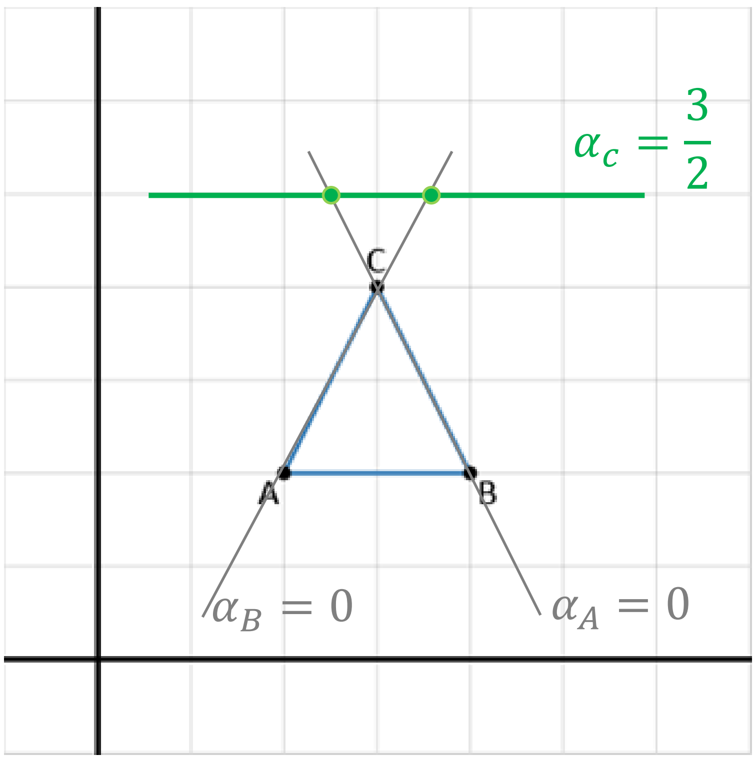
$$\alpha_B = 0 \rightarrow \alpha_A = \frac{3}{2}$$



Barycentric coordinates

Example: Given a triangle with vertices (2, 2), (4, 2), and (3, 4). Draw the iso-contours along which the barycentric coordinate α_c corresponding to point C has the value $\frac{1}{2}$, $-\frac{1}{2}$, $\frac{3}{2}$, respectively.

$$\alpha_A + \alpha_B + \alpha_c = 1$$



$$\alpha_c = \frac{3}{2}, \qquad \alpha_A = 0 \rightarrow \alpha_B = -\frac{1}{2}$$

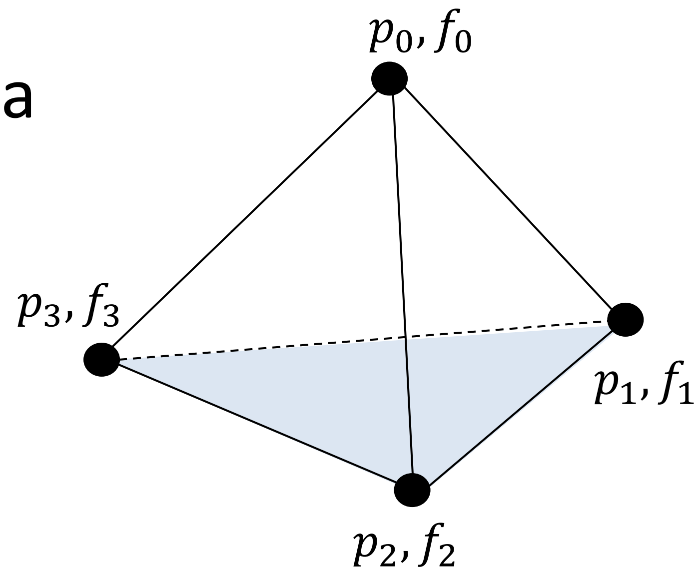
$$\alpha_B = 0 \rightarrow \alpha_A = -\frac{1}{2}$$

Data interpolation

SIEMENS
Ingenuity for life

- Interpolation of scalar values in a tetrahedron
  - A unique linear interpolation function $f(\boldsymbol{x}) = a + bx + cy + dz$ exists which interpolates the scalar values at the vertices
  - Solve for coefficients a, b, c, d via

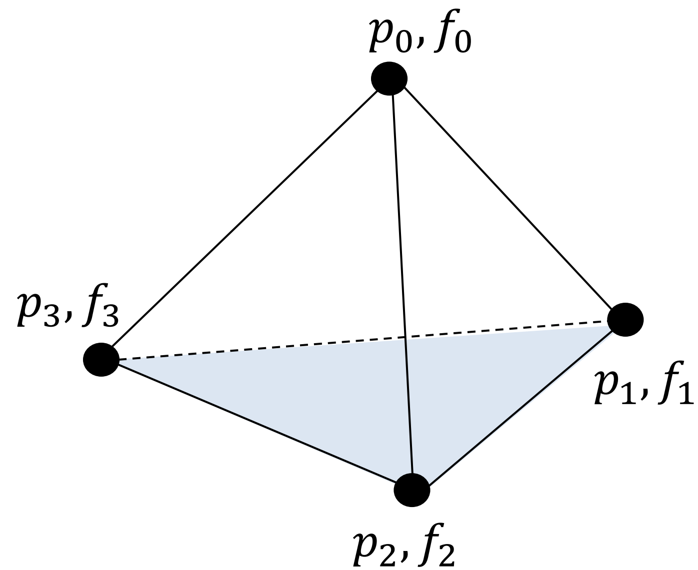
$$\begin{bmatrix} 1 & x_0 & y_0 & z_0 \\ 1 & x_1 & y_1 & z_1 \\ 1 & x_2 & y_2 & z_2 \\ 1 & x_3 & y_3 & z_3 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \end{bmatrix}$$



Data interpolation

SIEMENS
Ingenuity for life

- How to get the gradient inside the tetrahedron?
  - Given the linear interpolation function
$$f(\boldsymbol{x}) = a + bx + cy + dz$$
  - The gradient of the interpolated scalar field can be obtained by
$$\nabla f = \begin{pmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \\ \frac{\partial f}{\partial z} \end{pmatrix} = \begin{pmatrix} b \\ c \\ d \end{pmatrix}$$



  - The gradient is constant within the tetrahedron

Data interpolation

Example: For the tetrahedron with vertices A = (0,0,0), B = (1,0,0), C = (0,1,0), D = (0,0,1), compute the linear interpolation function $f(x, y, z) = a + b \cdot x + c \cdot y + d \cdot z$ which interpolates the scalar values $f_A = 1, f_B = 0, f_C = 0, f_D = 1$ at the corresponding vertices.

$f_A = 1 = a$

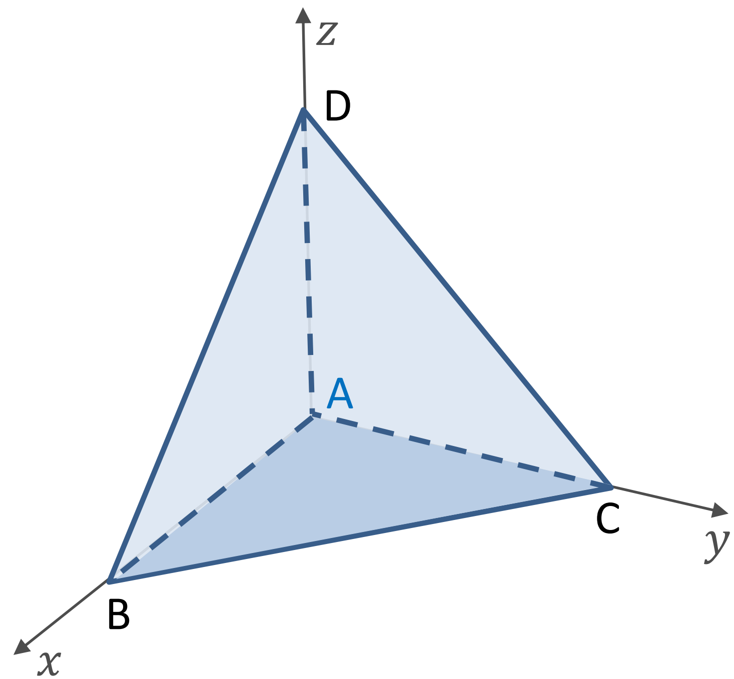
$f_B = 0 = a + b \rightarrow -1$

$f_C = 0 = a + c \rightarrow -1$

$f_d = 1 = a + d \rightarrow 0$

$f(x,y,z) = 1 - x - y$

$\nabla f = (-1, -1, 0)^T$



Compute the gradient of the interpolated scalar field at the center of the tetrahedron.

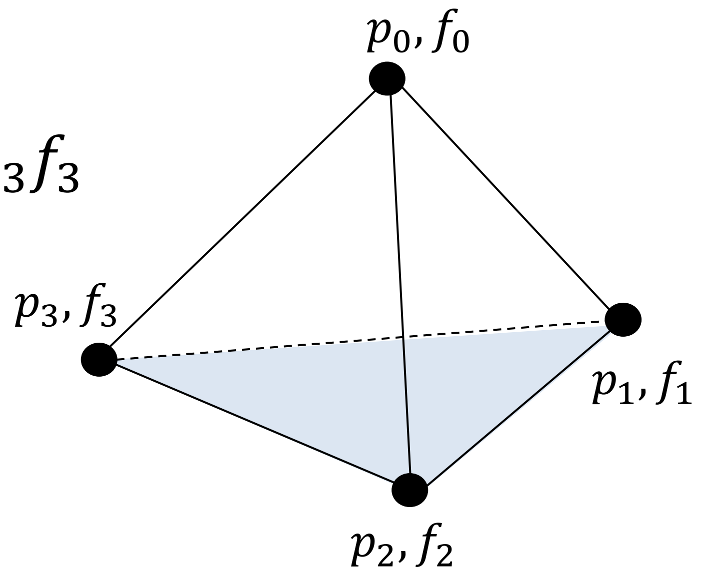
Data interpolation

SIEMENS
Ingenuity for life

- Barycentric interpolation in 3D
  - Scalar values can be interpolated by means of barycentric coordinates:

$$\boldsymbol{x} = \alpha_0 p_0 + \alpha_1 p_1 + \alpha_2 p_2 + \alpha_3 p_3$$
$$\alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 = 1$$

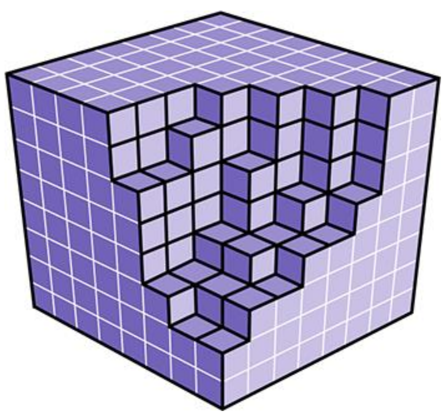
$$\rightarrow f(x, y, z) = \alpha_0 f_0 + \alpha_1 f_1 + \alpha_2 f_2 + \alpha_3 f_3$$



Overview

SIEMENS
Ingenuity for life

- Interpolation on grids / meshes
  - Barycentric interpolation
  - Bi-/trilinear interpolation
  - High-quality reconstruction

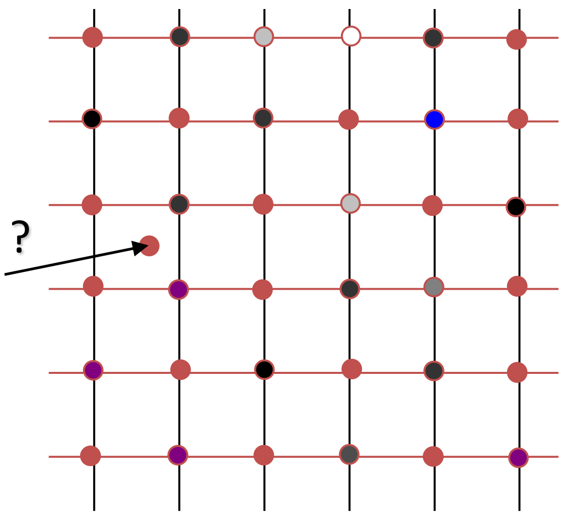


Tricubic vs. trilinear interpolation

Interpolation on grids

SIEMENS
Ingenuity for life

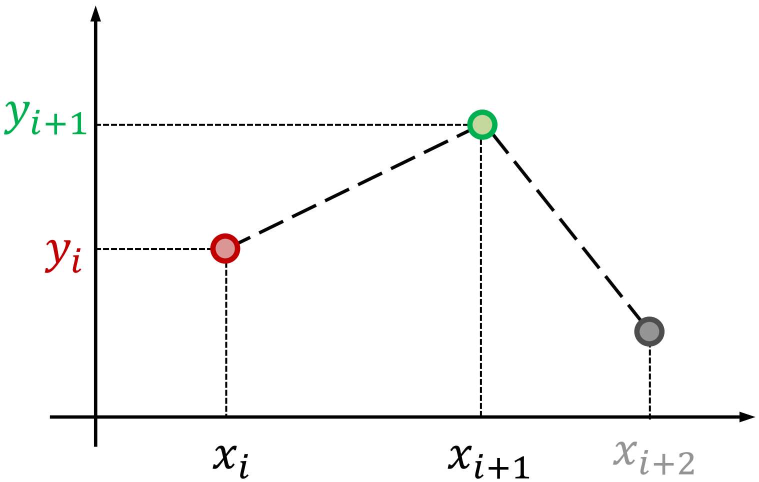
- **Problem**: assume data values are given only at vertices of a Cartesian grid
- How can we hide the underlying grid structure, i.e., how can we get a continuous data distribution over the spatial domain?
- This can be done via interpolation!



Interpolation on grids

SIEMENS
Ingenuity for life

- Piecewise linear interpolation
  - Simplest approach (except for piece-wise constant interpolation)
  - Data points: $(x_1,y_1),\ldots,(x_N,y_N)$



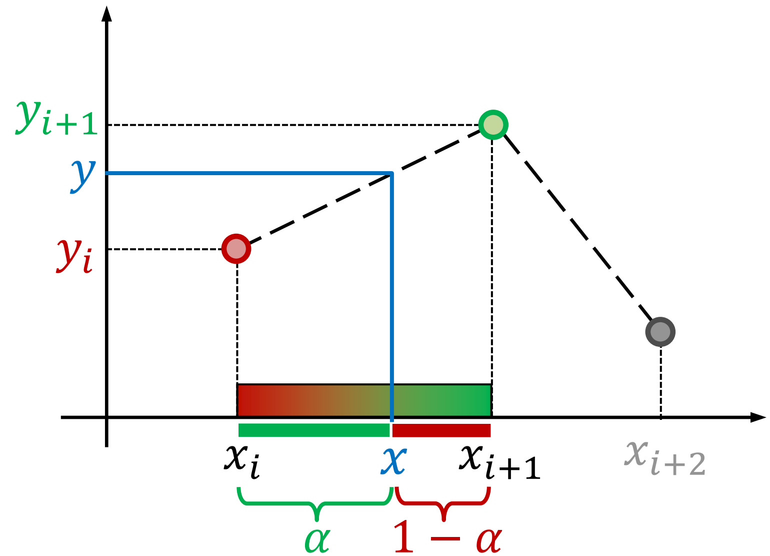
Interpolation on grids

- Piecewise linear interpolation
  - Simplest approach (except for piece-wise constant interpolation)
  - Data points: $(x_1,y_1),\dots,(x_N,y_N)$
  - For any point x with

$$x_i \le x \le x_{i+1}$$

evaluate $f(x) = (1-\alpha)y_i + \alpha y_{i+1}$

where $\alpha = \dfrac{x-x_i}{x_{i+1}-x_i} \in [0,1]$

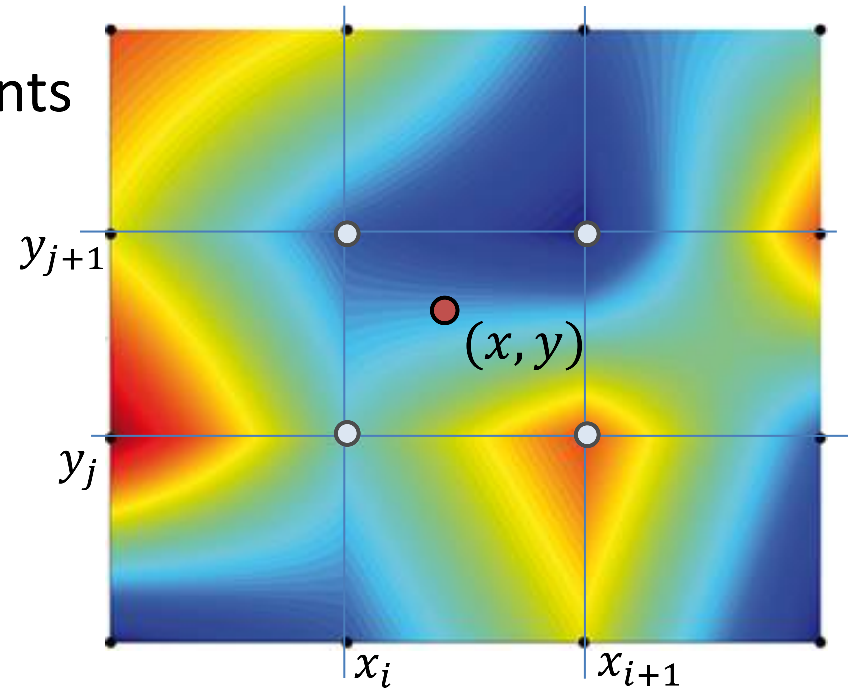


The closer x is to the green point, the more green we want

Interpolation on grids

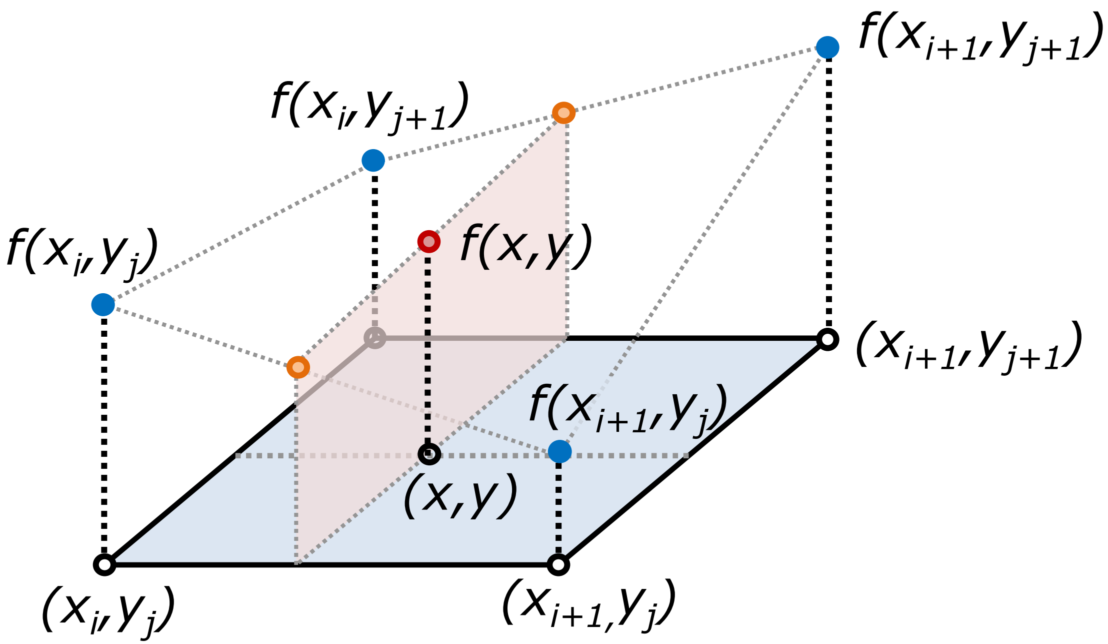
SIEMENS
Ingenuity for life

- Linear interpolation
  - C^0 continuity at segment boundaries
    - Tangents don't match at segment transition
  - Easily extendible to 2D
    - 2D cell consisting of 4 data points $(x_i, y_j), \ldots, (x_{i+1}, y_{j+1})$ with scalar values $f_{k,l} = f(x_k, y_l)$
    - Bilinear interpolation of points (x, y) with $x_i \leq x \leq x_{i+1}$ and $y_j \leq y \leq y_{j+1}$



Interpolation on grids

- Bilinear interpolation on a rectangle



Interpolation on grids

SIEMENS
Ingenuity for life

- Bilinear interpolation on a rectangle

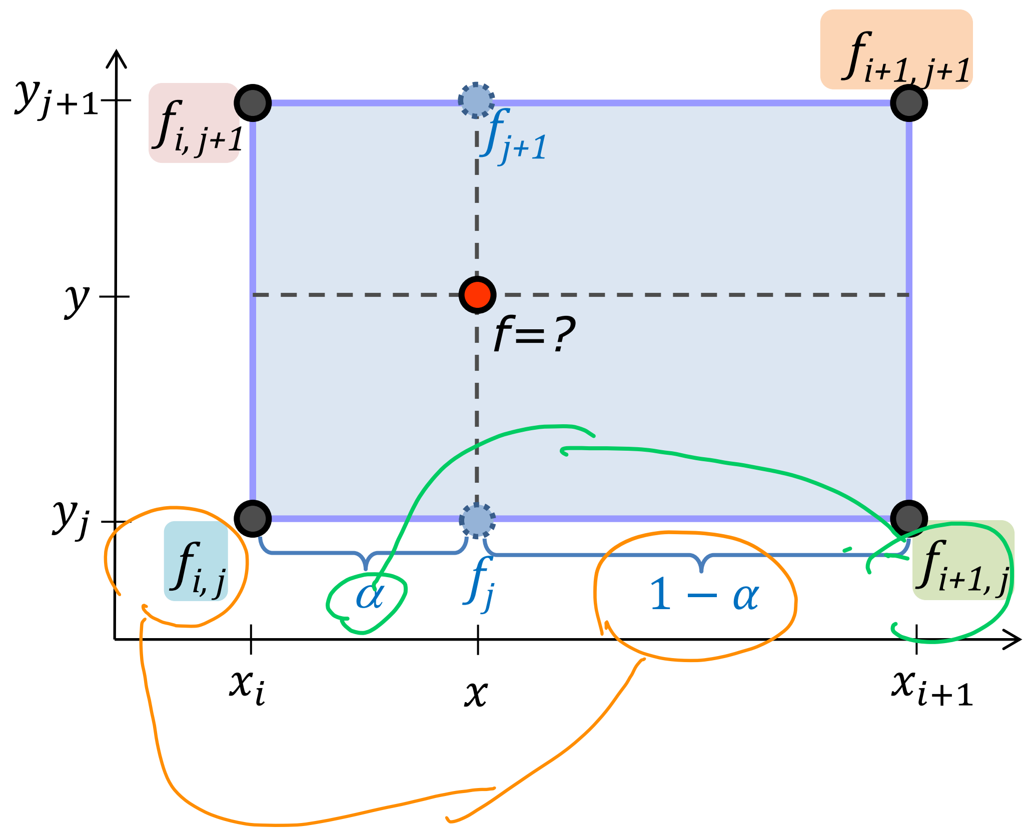
$$f(\alpha,\beta) = ?$$

Interpolate horizontally

$$f_j = (1-\alpha)\, f_{i,j} + \alpha\, f_{i+1,j}$$

$$f_{j+1} = (1-\alpha)\, f_{i,j+1} + \alpha\, f_{i+1,j+1}$$

$$\alpha = \frac{x - x_i}{x_{i+1} - x_i}$$



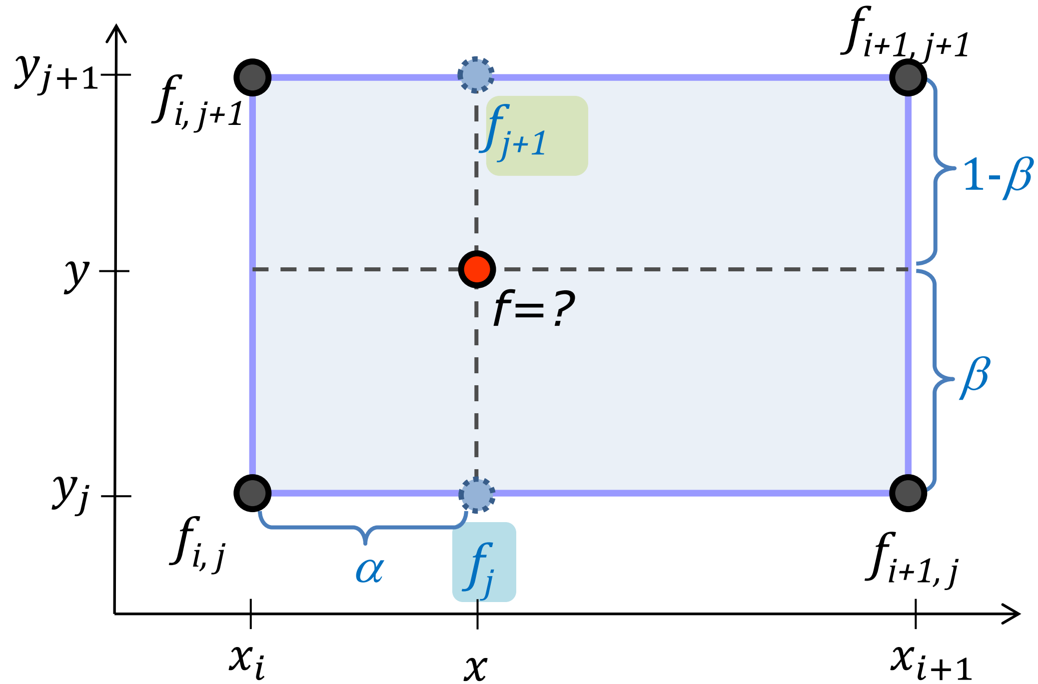
Interpolation on grids

SIEMENS
Ingenuity for life

- Bilinear interpolation on a rectangle

$$f(\alpha,\beta) = (1-\beta)[(1-\alpha)\, f_{i,j} + \alpha\, f_{i+1,j}] + \beta\, [(1-\alpha)\, f_{i,j+1} + \alpha\, f_{i+1,j+1}]$$

$$= (1-\beta)\, f_j + \beta\, f_{j+1}$$



with

$$f_j = (1-\alpha)\, f_{i,j} + \alpha\, f_{i+1,j}$$

$$f_{j+1} = (1-\alpha)\, f_{i,j+1} + \alpha\, f_{i+1,j+1}$$

and local coordinates

$$\alpha = \frac{x - x_i}{x_{i+1} - x_i}, \qquad \beta = \frac{y - y_i}{y_{i+1} - y_i}, \qquad \alpha, \beta \in [0,1]$$

Interpolation on grids

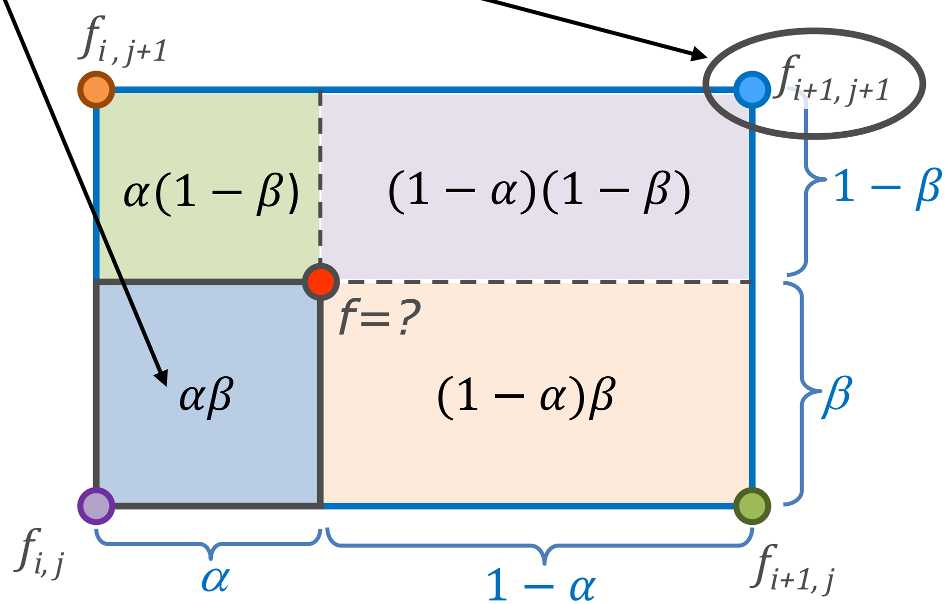
SIEMENS
Ingenuity for life

- Geometric interpretation of bilinear interpolation

$$f(\alpha,\beta) = (1-\beta)[(1-\alpha)f_{i,j} + \alpha f_{i+1,j}] + \beta[(1-\alpha)f_{i,j+1} + \alpha f_{i+1,j+1}]$$

$$= (1-\alpha)(1-\beta)f_{i,j} + \alpha(1-\beta)f_{i+1,j} + (1-\alpha)\beta f_{i,j+1} + \alpha\beta f_{i+1,j+1}$$

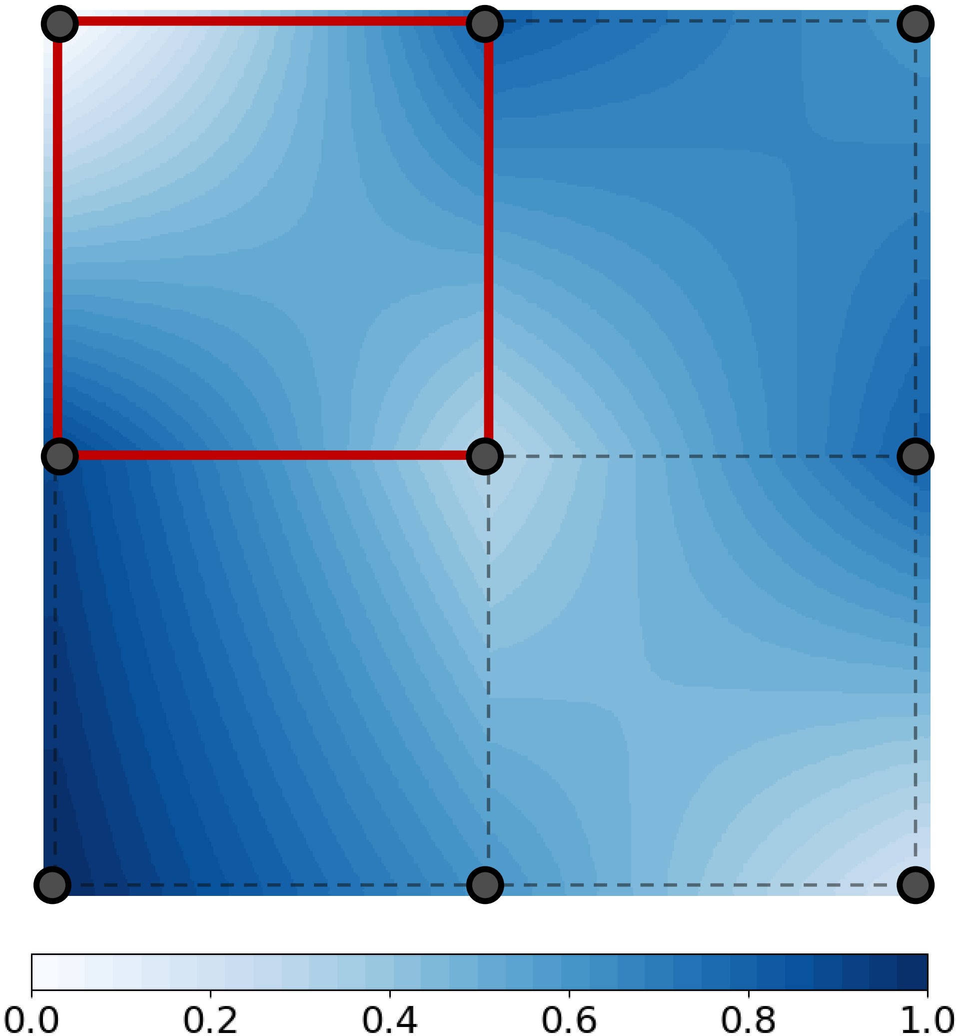
  – Opposite points are weighted by local areas



Interpolation on grids

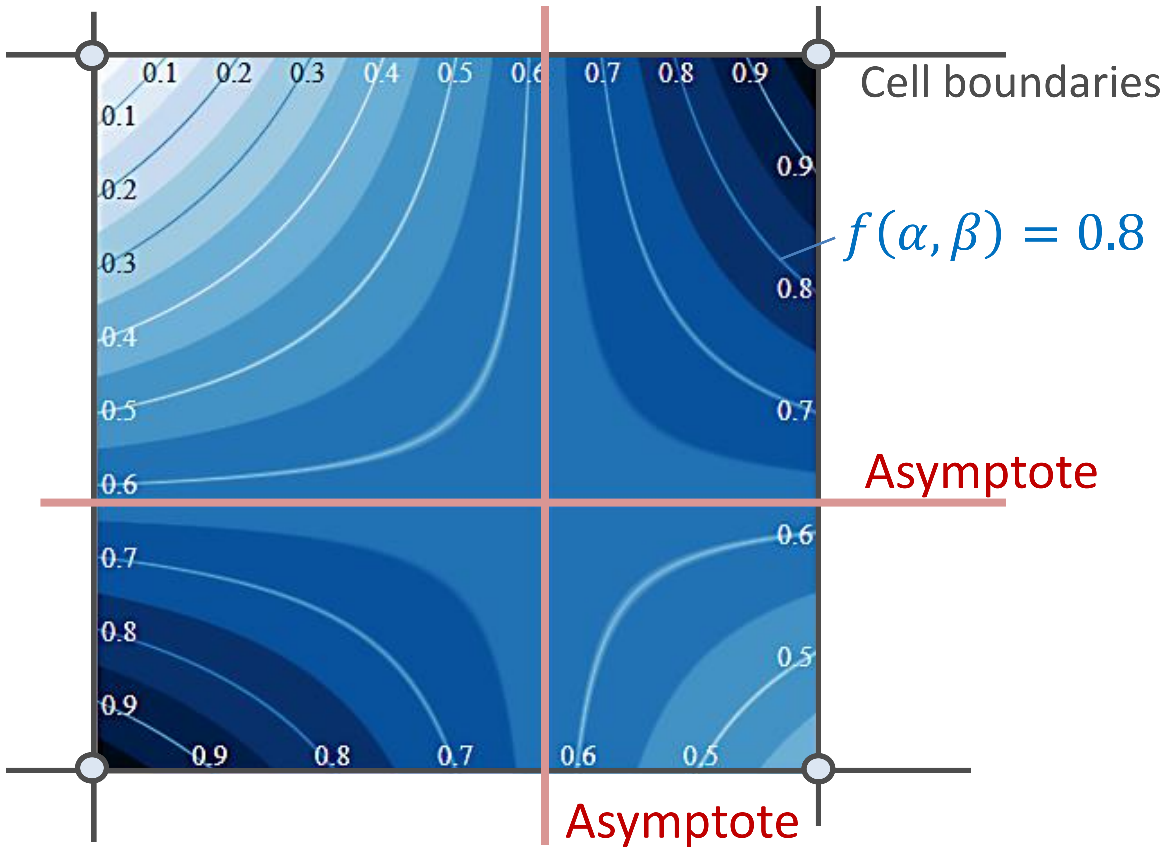
SIEMENS
Ingenuity for life

- Example
  - Data values given at 9 grid vertices
  - Bilinear interpolation used within grid cells



Interpolation on grids

- When bilinear interpolation is used, isolines within a cell are hyperbolas
  - **Isoline:** curve on which all points have a certain value

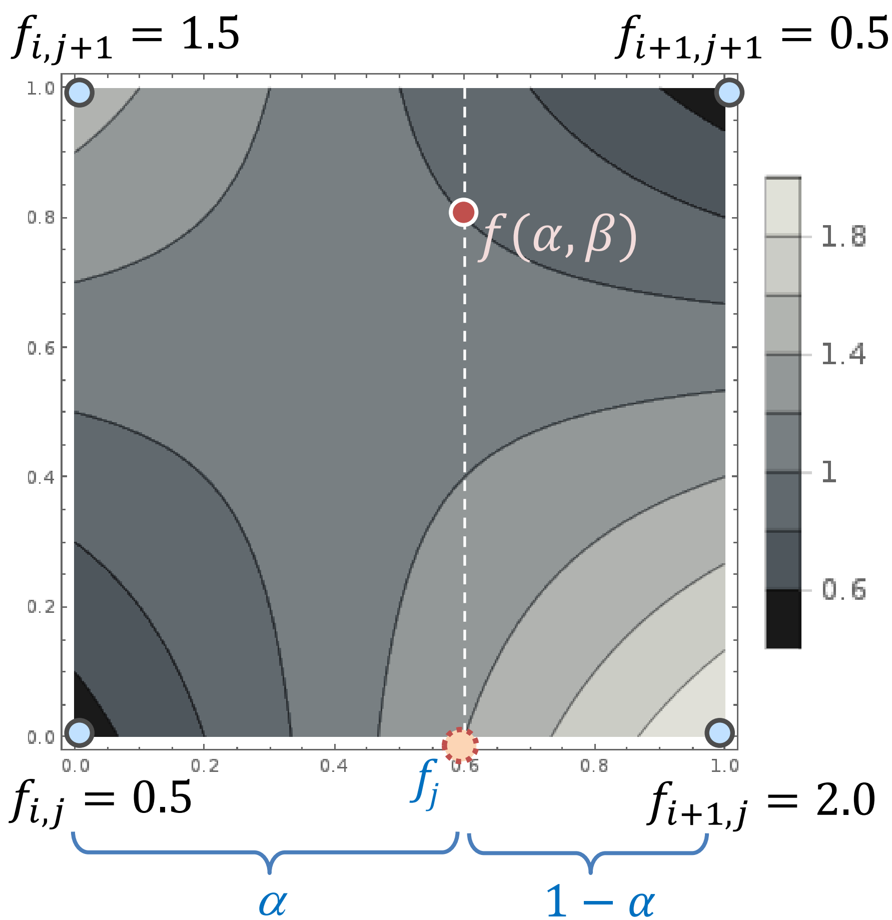


multivis.net/lecture/bilinear.html

Interpolation on grids

- How to evaluate the isolines?

$$
\begin{aligned}
f_j &= \alpha\, f_{i+1,j} + (1-\alpha) f_{i,j} \\
&= f_{i,j} + \left(f_{i+1,j} - f_{i,j}\right)\alpha \\
&= 0.5 + 1.5\alpha
\end{aligned}
$$



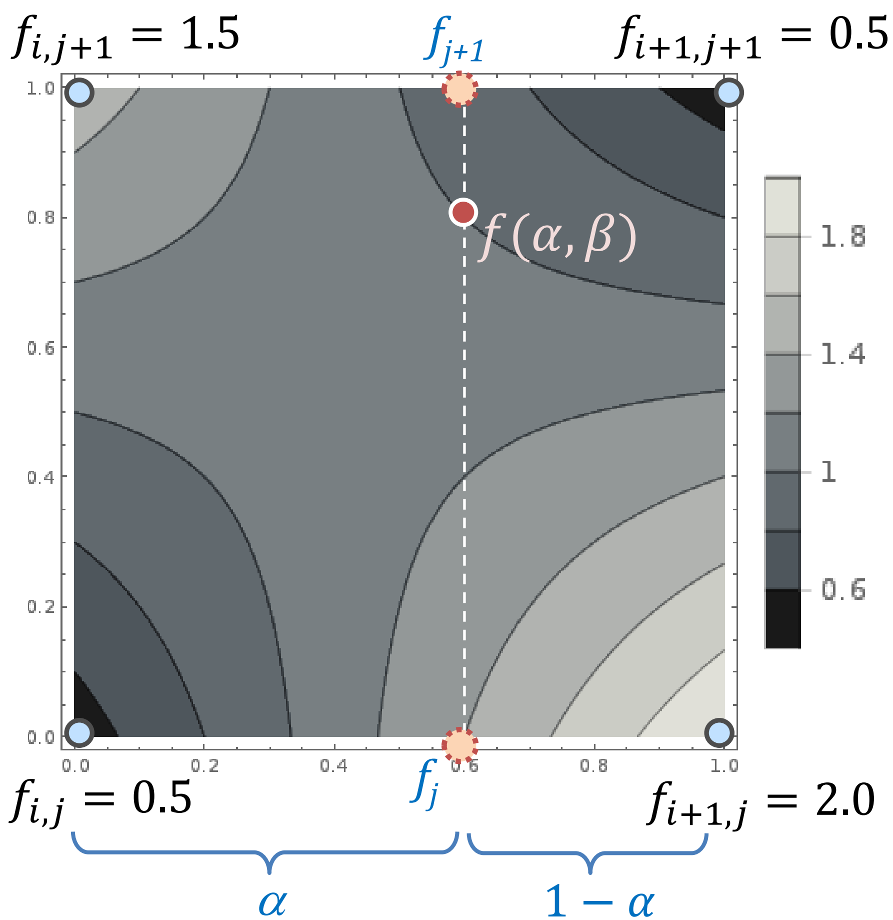
multivis.net/lecture/bilinear.html

Interpolation on grids

SIEMENS
Ingenuity for life

- How to evaluate the isolines?

$$f_j = \alpha\, f_{i+1,j} + (1-\alpha) f_{i,j}$$
$$= f_{i,j} + \left(f_{i+1,j} - f_{i,j}\right)\alpha$$
$$= 0.5 + 1.5\alpha$$
$$f_{j+1} = f_{i,j+1} + \left(f_{i+1,j+1} - f_{i,j+1}\right)\alpha$$
$$= 1.5 - \alpha$$



multivis.net/lecture/bilinear.html

Interpolation on grids

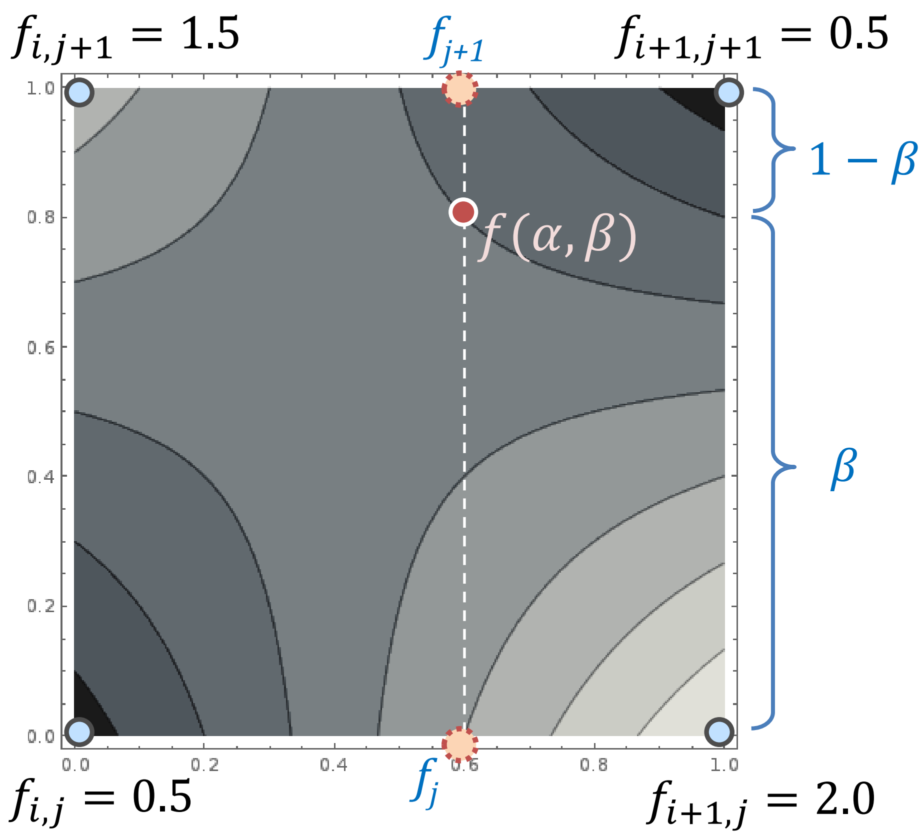
SIEMENS
Ingenuity for life

- How to evaluate the isolines?

$$f_j = \alpha\, f_{i+1,j} + (1-\alpha) f_{i,j}$$
$$= f_{i,j} + \left(f_{i+1,j} - f_{i,j}\right)\alpha$$
$$= 0.5 + 1.5\alpha$$
$$f_{j+1} = f_{i,j+1} + \left(f_{i+1,j+1} - f_{i,j+1}\right)\alpha$$
$$= 1.5 - \alpha$$
$$f(\alpha,\beta) = f_j + \left(f_{j+1} - f_j\right)\beta$$
$$= (0.5 + 1.5\alpha) + \left(1.5 - \alpha - (0.5 + 1.5\alpha)\right)\beta$$

$$f(\alpha,\beta) = 0.5 + 1.5\alpha + \beta - 2.5\alpha\beta$$

Bi-linear interpolation function defining the scalar value at each point (α, β) within the cell

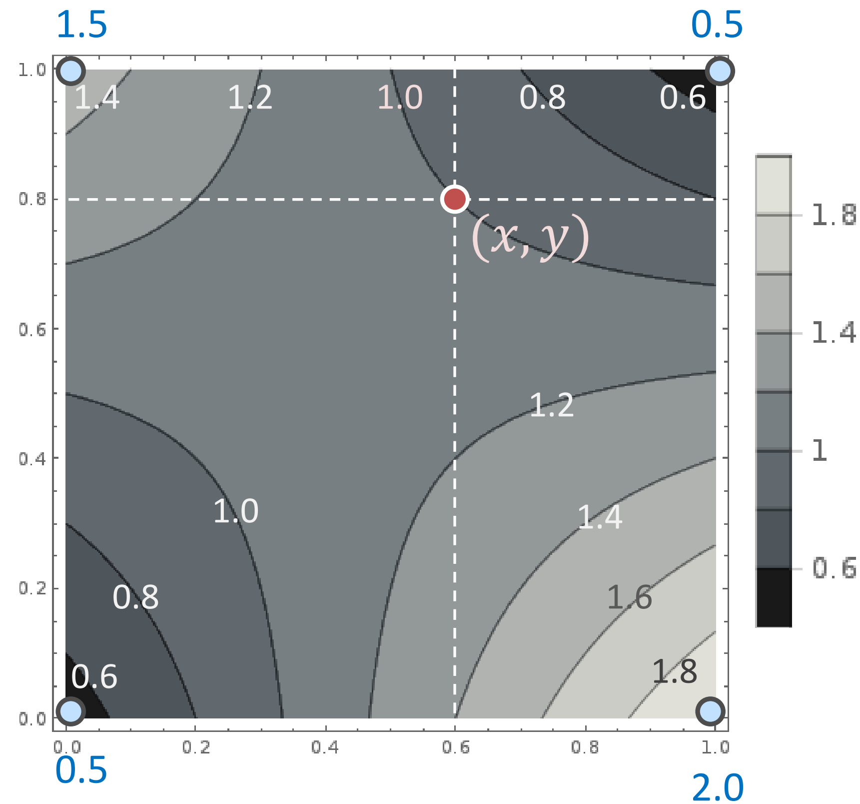


multivis.net/lecture/bilinear.html

Interpolation on grids

- How to evaluate the isolines?
  - Compute y-coordinate of a point (x, y) with $x = 0.6$ which is on the iso-contour $f(x, y) = 1$

$$f(x, y) = 0.5 + 1.5x + y - 2.5xy$$



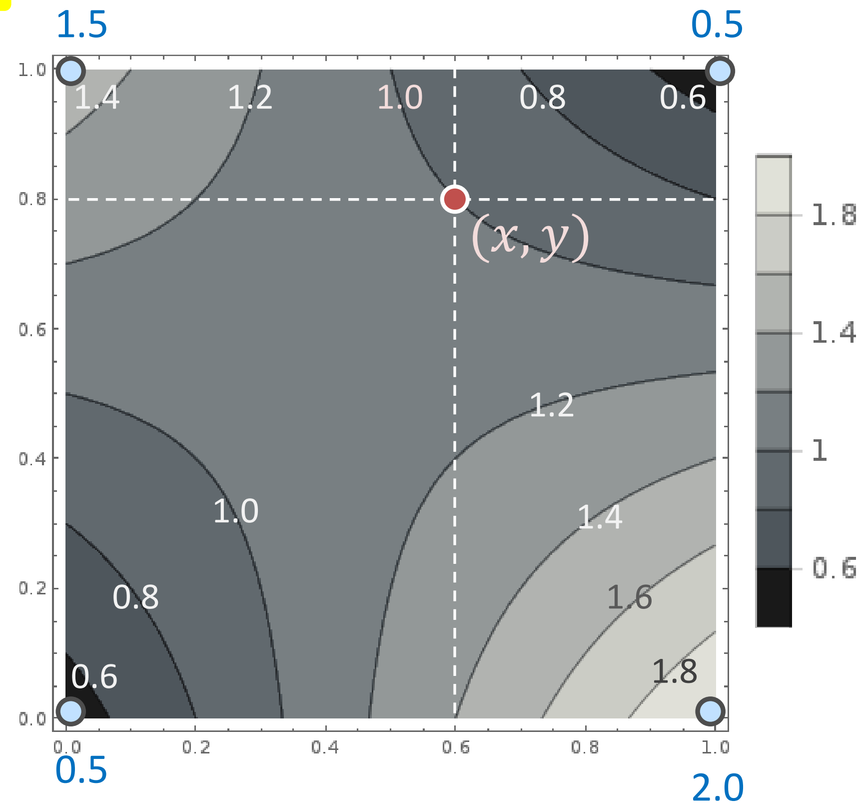
Interpolation on grids

- How to evaluate the isolines?
  - Compute y-coordinate of a point (x, y) with $x = 0.6$ which is on the iso-contour $f(x, y) = 1$

$$f(x, y) = 0.5 + 1.5x + y - 2.5xy$$

$$1 = 0.5 + 1.5 \cdot 0.6 + y - 2.5 \cdot 0.6 \cdot y$$

$$1 = 1.4 - 0.5y$$

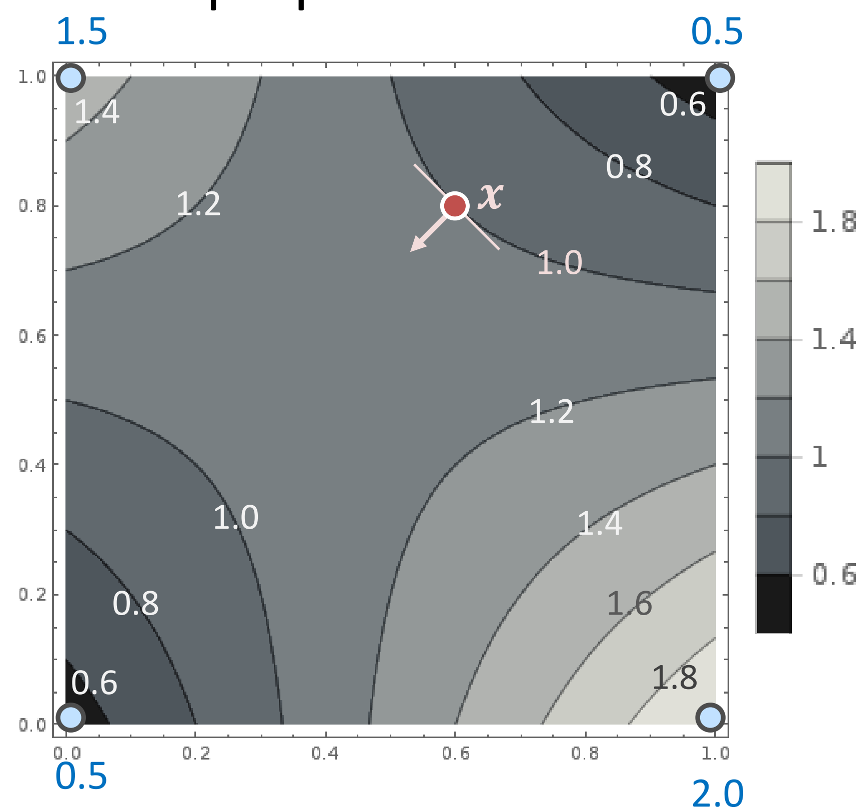


The coordinates of the point are $(0.6, 0.8)$

Interpolation on grids

- What is the normal at a point $\boldsymbol{x}$ on an iso-surface?
  - It is the gradient at this point, which is perpendicular to the tangent of the iso-surface
  - Gradient points into direction of steepest ascent of f

$$\nabla f(\boldsymbol{x}) = \left(\frac{\partial}{\partial x} f(\boldsymbol{x}),\ \frac{\partial}{\partial y} f(\boldsymbol{x})\right)$$



Interpolation on grids

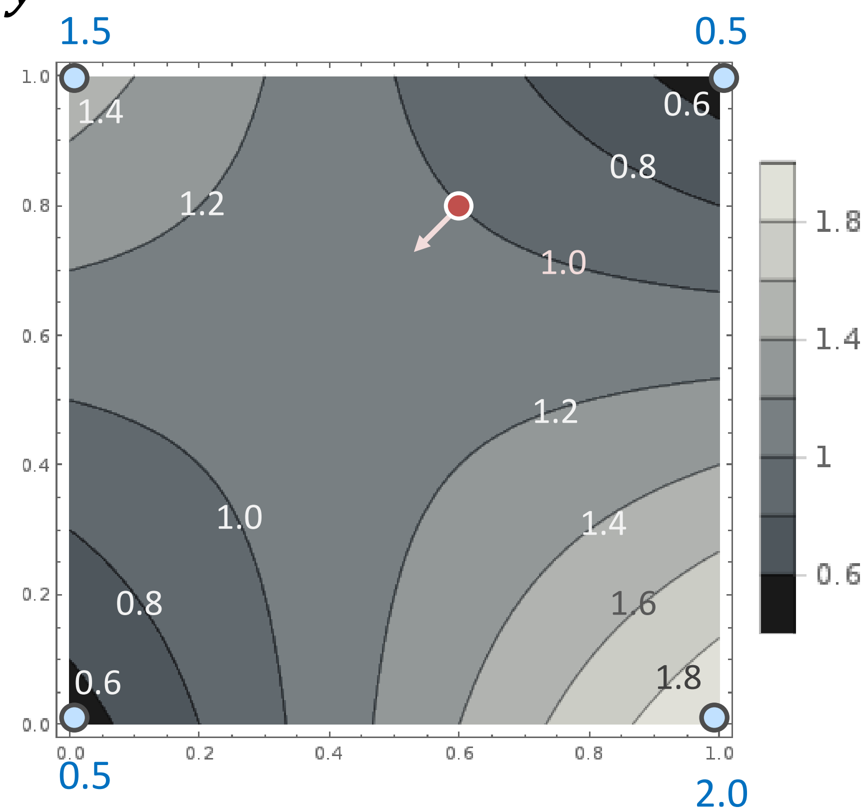
- What is the normal at point $(0.6, 0.8)$ on the iso-surface?

$$f(x, y) = 0.5 + 1.5x + y - 2.5xy$$

$$\frac{\partial f}{\partial x} = 1.5 - 2.5y$$

$$\frac{\partial f}{\partial y} = 1 - 2.5x$$

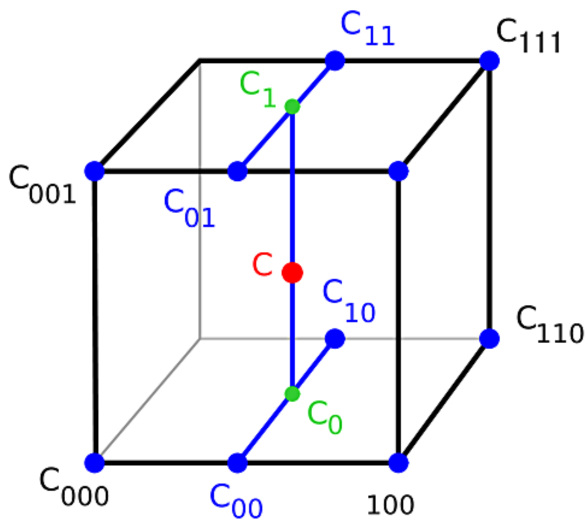
Gradient at $(0.6, 0.8)$: $\begin{pmatrix} -0.5 \\ -0.5 \end{pmatrix}$



Interpolation on grids

SIEMENS
Ingenuity for life

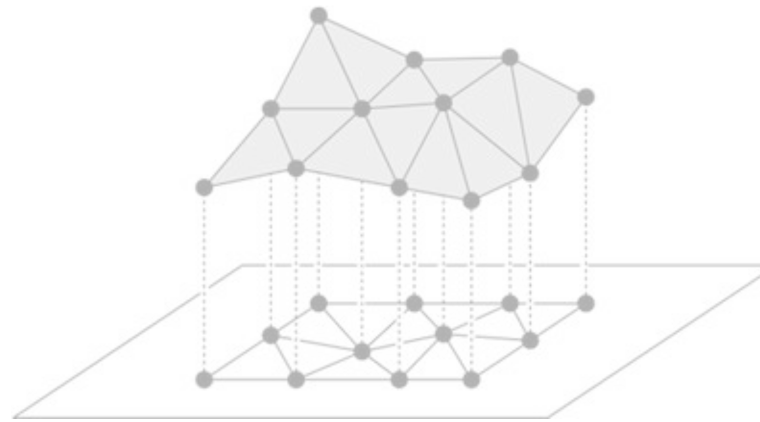
- In 3D we use trilinear interpolation:

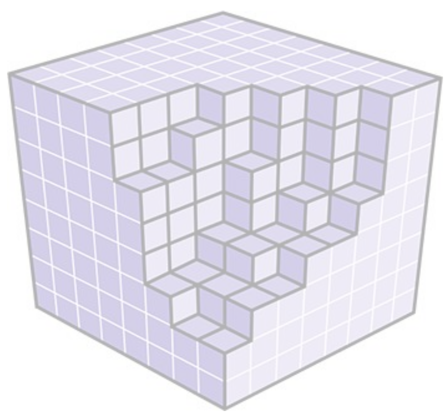


  - Apply linear interpolation of the initial data along the edges to obtain C_{00} , C_{01} , C_{10} , C_{11}
  - Interpolate linearly between C_{00} and C_{01} , and between C_{10} and C_{11} to obtain C_0 and C_1
  - Finally, interpolate between C_1 and C_0 to obtain C.

Overview

- Interpolation on grids / meshes
  - Barycentric interpolation
  - Bi-/trilinear interpolation
  - High-quality reconstruction

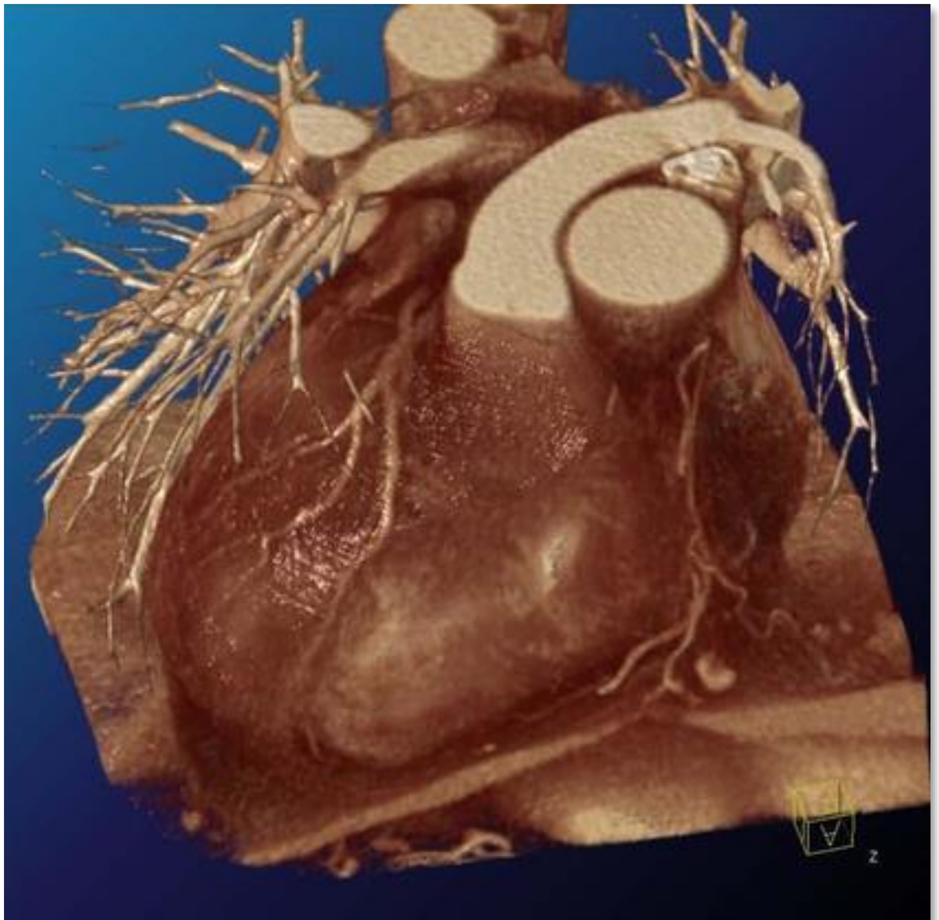


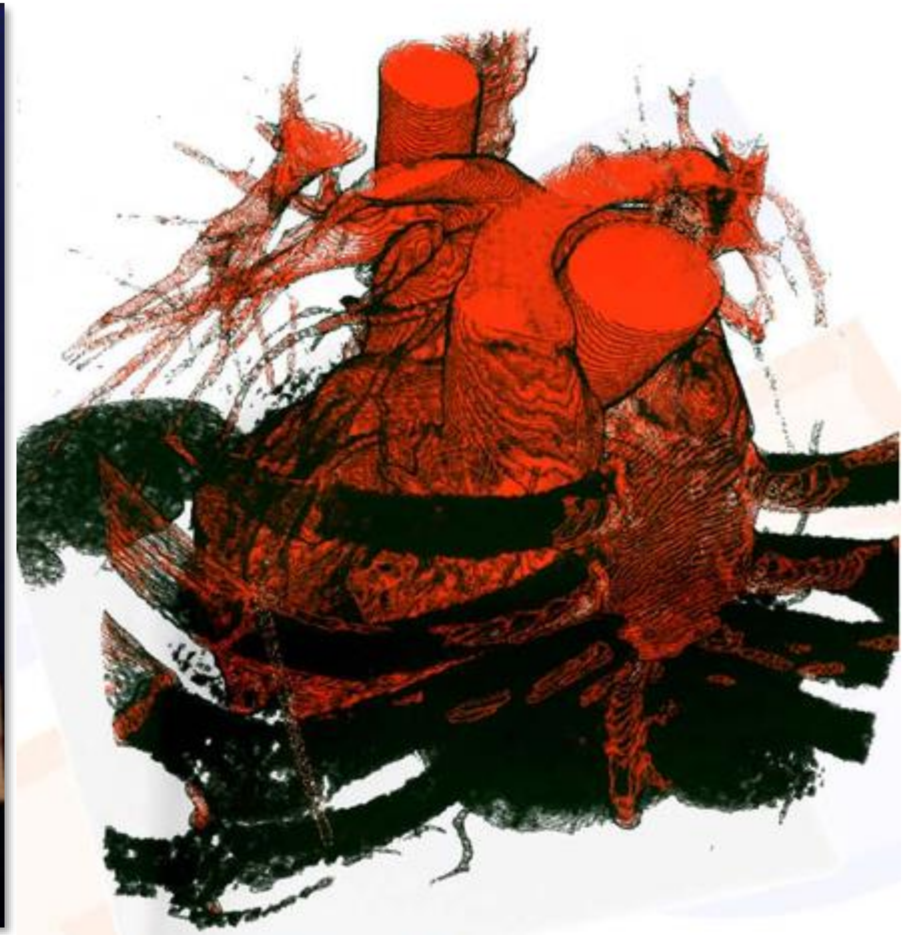


Tricubic vs. trilinear interpolation

Interpolation on grids

- This is what we want…



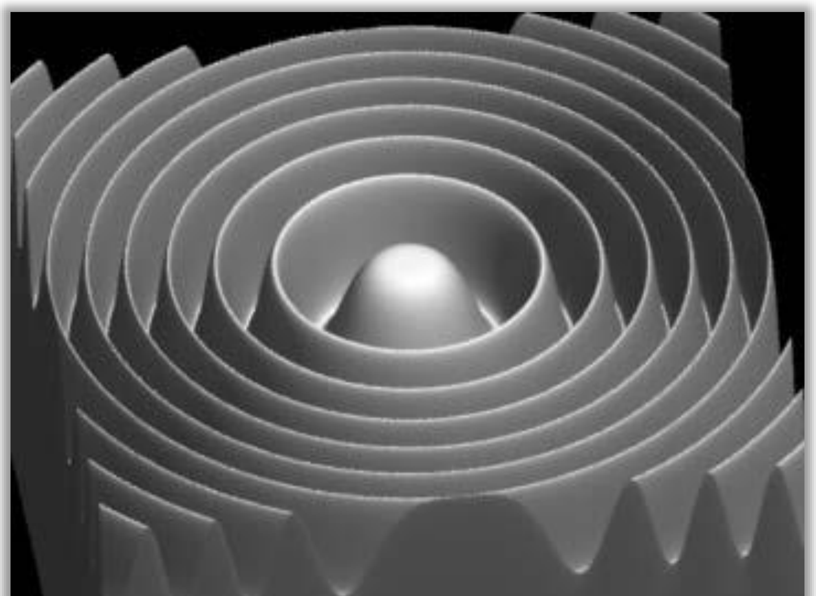


…but sometimes this is what we get

Interpolation on grids

SIEMENS
Ingenuity for life

- Higher-order reconstruction/interpolation
  - Required if very high quality is needed
  - Usually tested on Marschner-Lobb function
    - High amount of its energy is near its Nyquist frequency
    - Very demanding test for accurate reconstruction

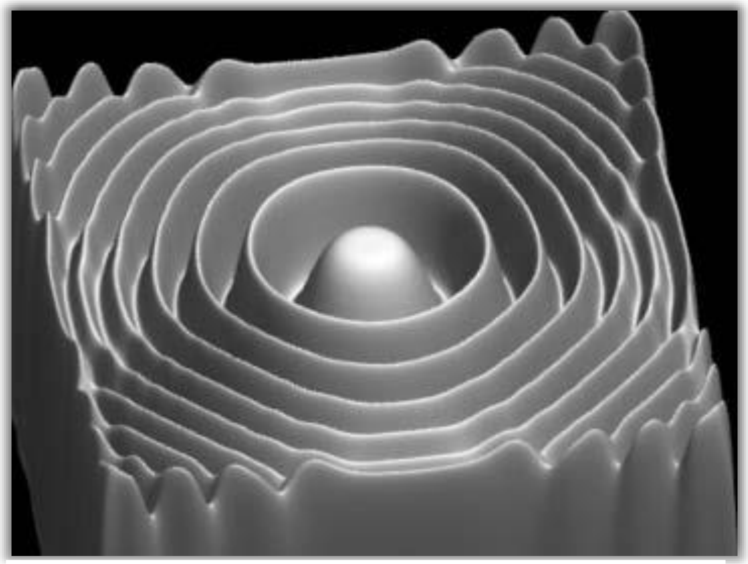


[Marschner & Lobb 94]

High-quality Reconstruction

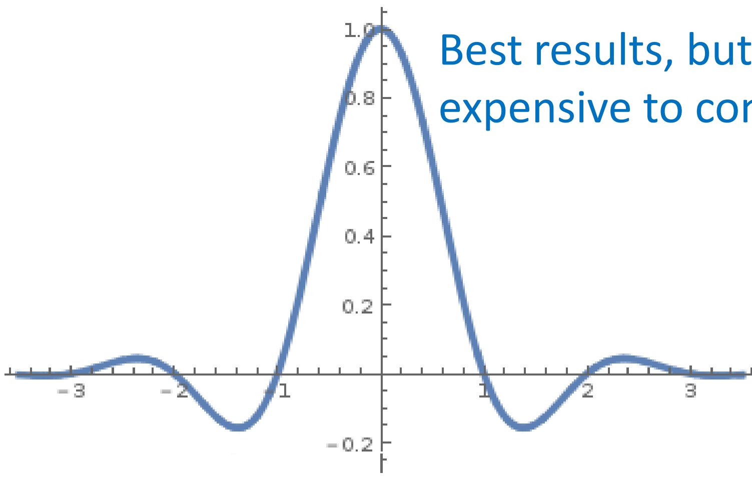
SIEMENS
Ingenuity for life

- Sinc function



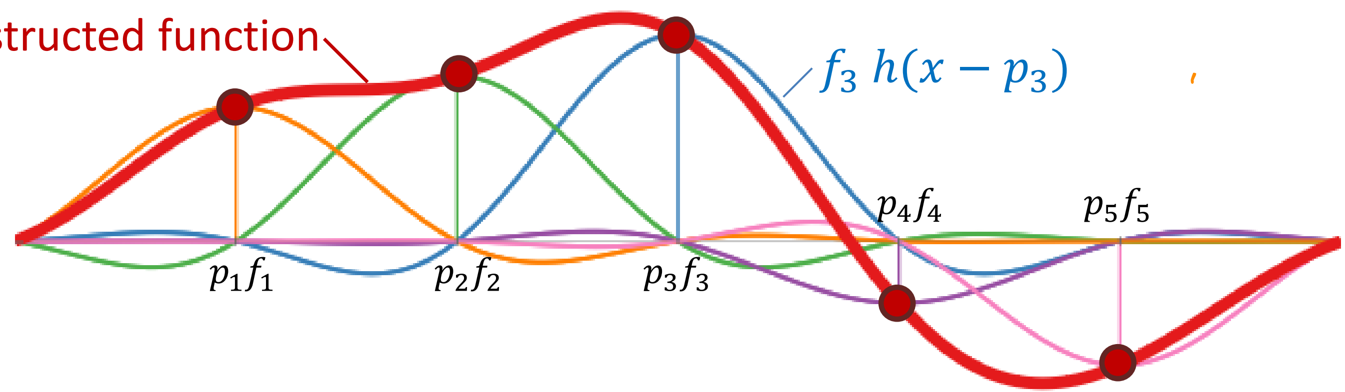
[Marschner & Lobb 94]

Windowed sinc (Hann window)



Best results, but very expensive to compute

"Optimal" reconstruction filter

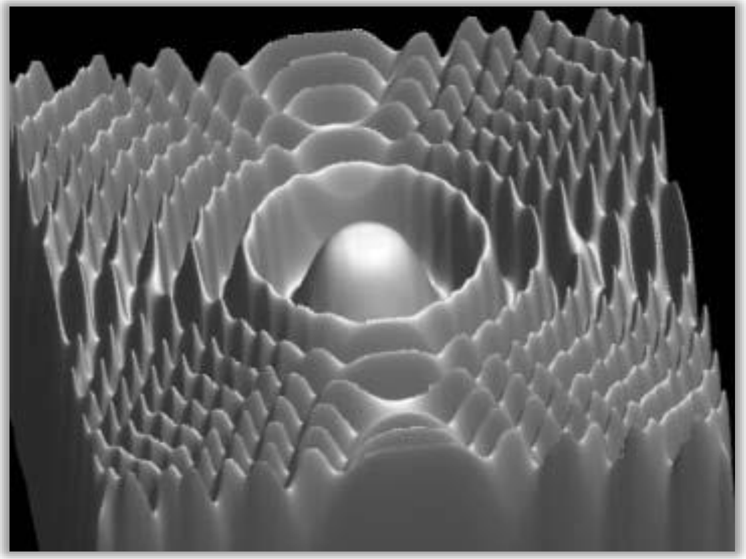


multivis.net/lecture/interpolation.html

High-quality Reconstruction

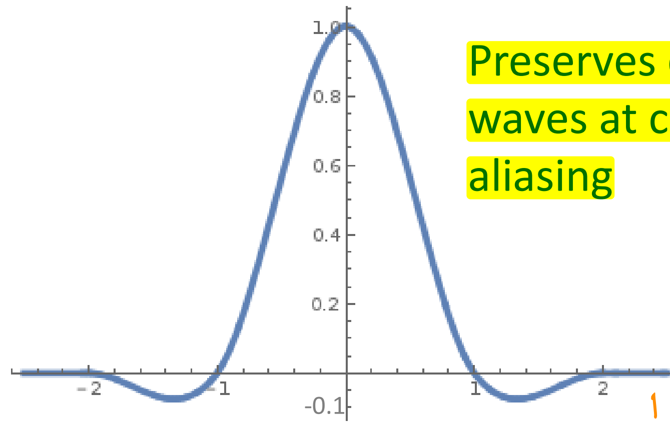
SIEMENS
Ingenuity for life

- Bicubic interpolation (Catmull-Rom spline)



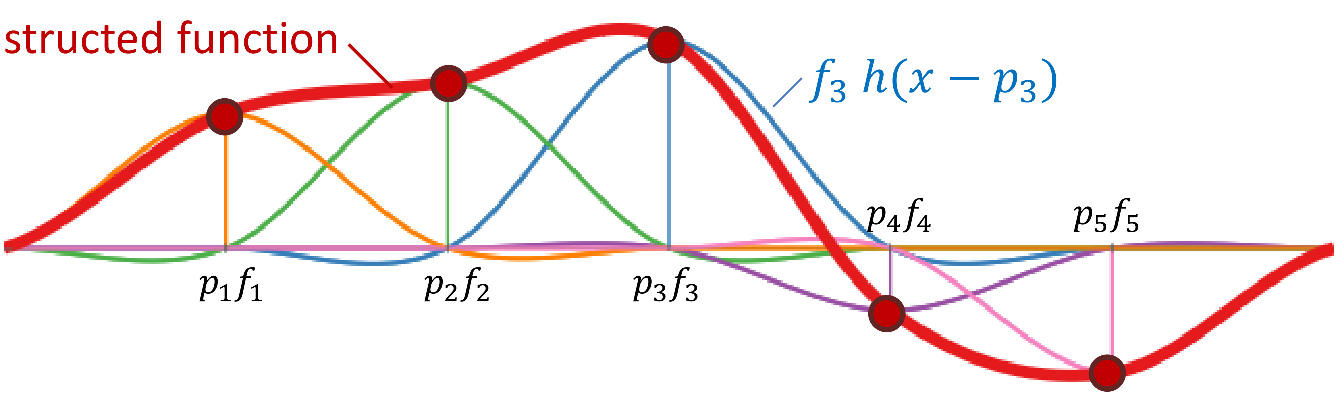
[Marschner & Lobb 94]

Reconstructed Marschner-Lobb function



Preserves depth of waves at cost of aliasing

Interpolation: 1 at center and 0 at integers

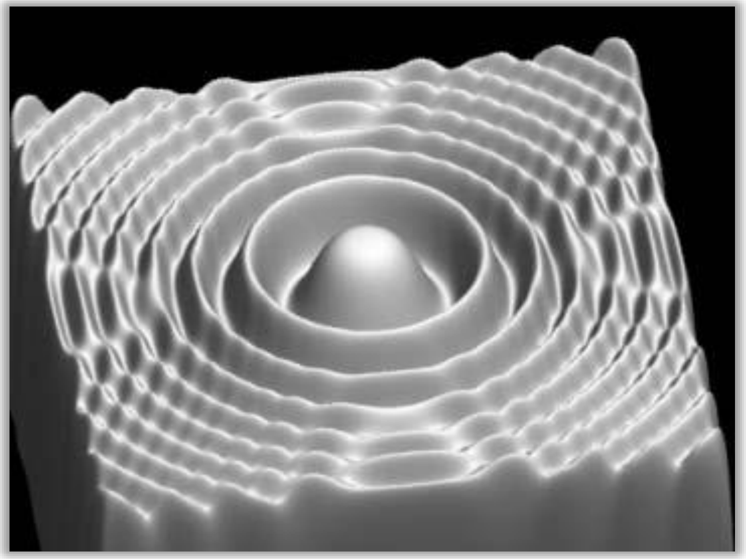


multivis.net/lecture/interpolation.html

High-quality Reconstruction

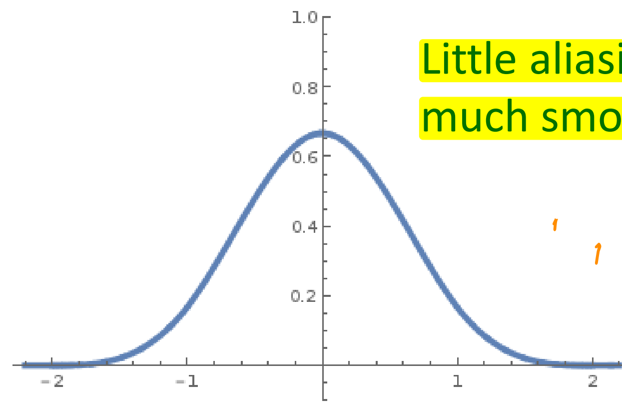
SIEMENS
Ingenuity for life

- Cubic B-spline (with smoothing)



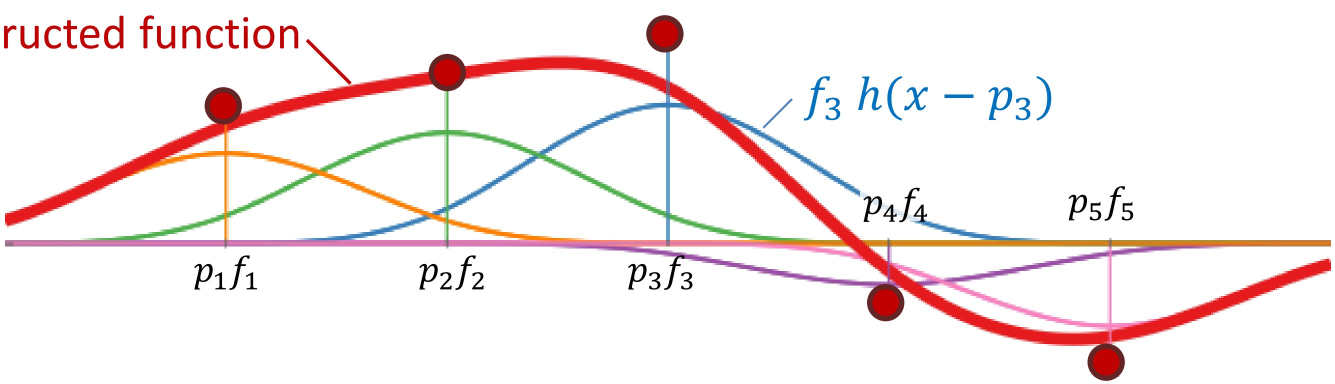
[Marschner & Lobb 94]

Reconstructed Marschner-Lobb function



Little aliasing, but much smoothing

Smoothing: $^2/_3$ at center and $^1/_6$ at ± 1

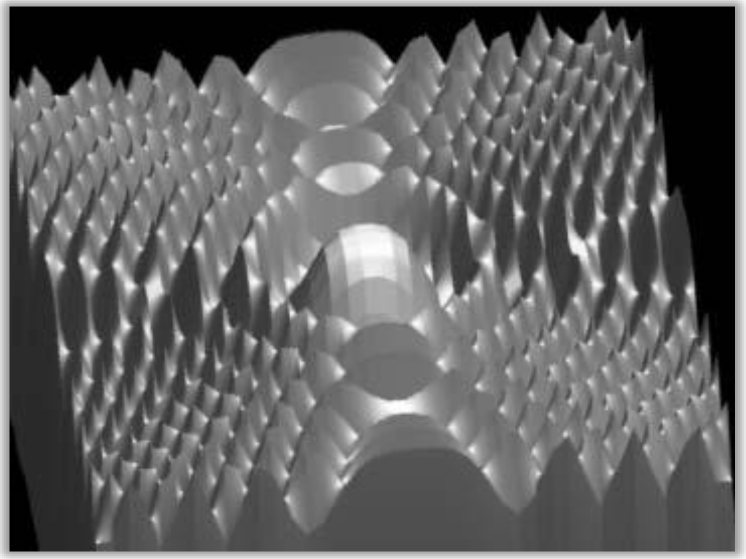


multivis.net/lecture/interpolation.html

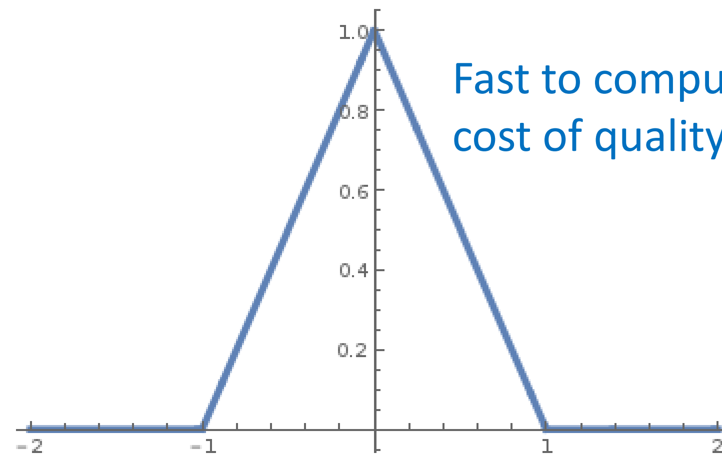
High-quality Reconstruction

SIEMENS
Ingenuity for life

- Trilinear interpolation (Tent)

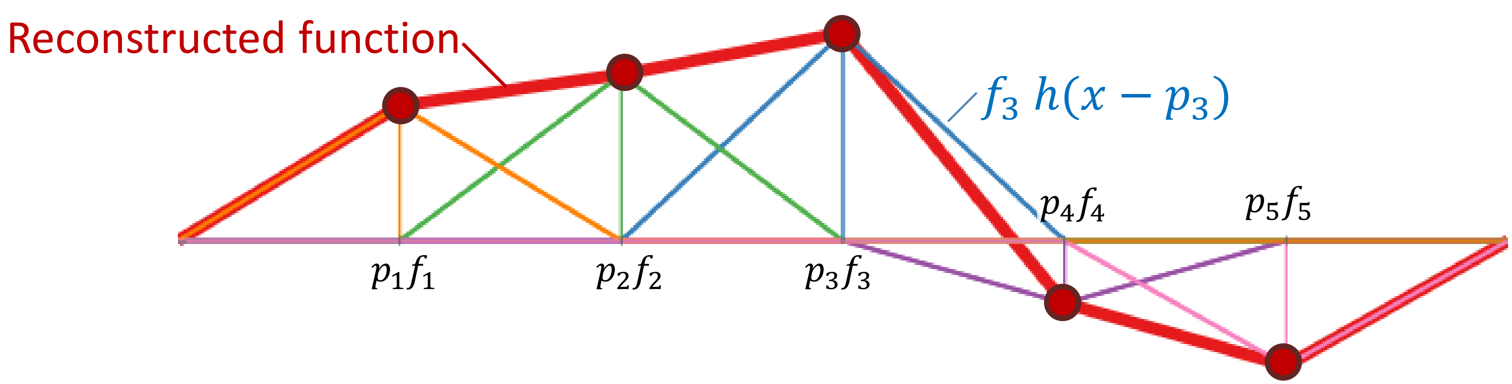


Reconstructed Marschner-Lobb function



Fast to compute at cost of quality

Interpolation: 1 at center and 0 at integers

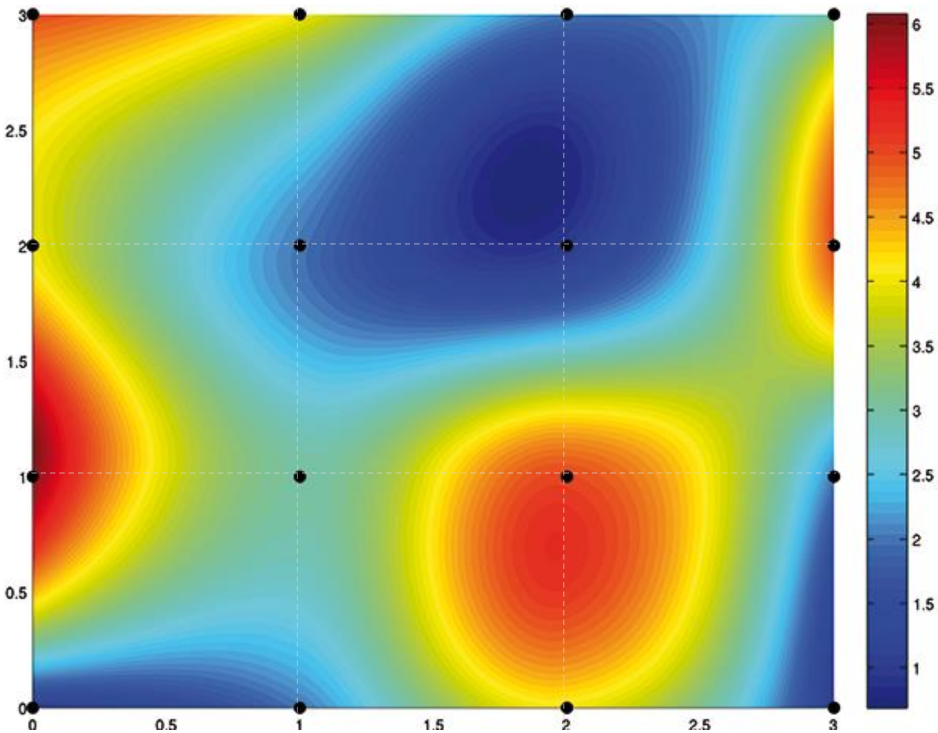


multivis.net/lecture/interpolation.html

Interpolation on grids

SIEMENS
Ingenuity for life

- 2D interpolation with different reconstruction filters



Bicubic interpolation (default in Photoshop)

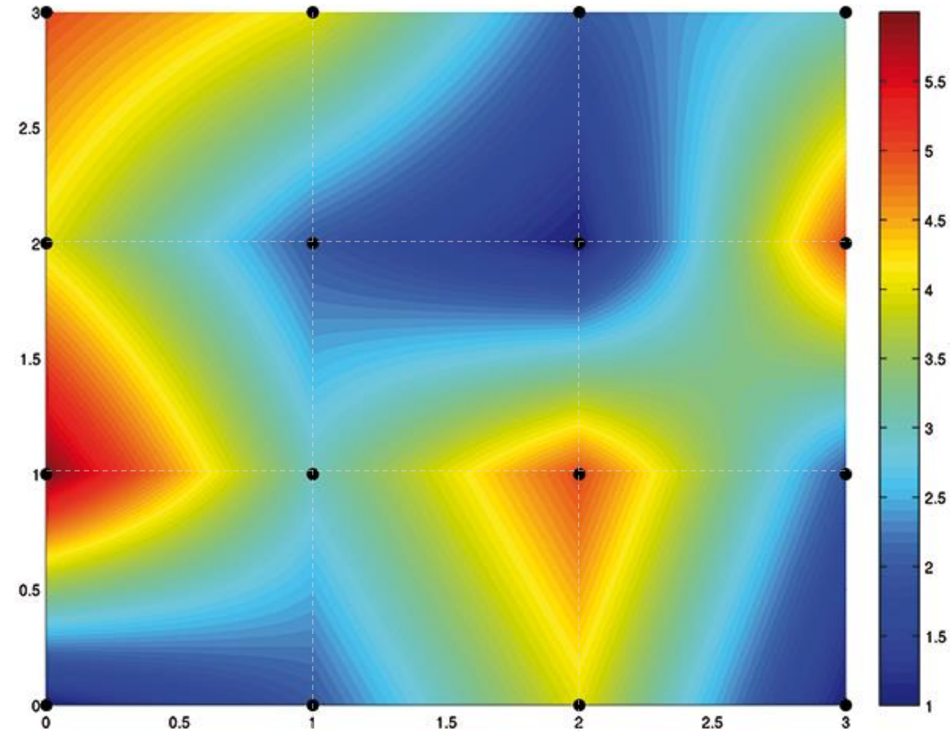
C^1 continuous → tangents match at segment transition

First derivative

Second derivative

Cubic B-spline (C^2 continuous → also curvature matches)

Eğrilik



Bilinear interpolation

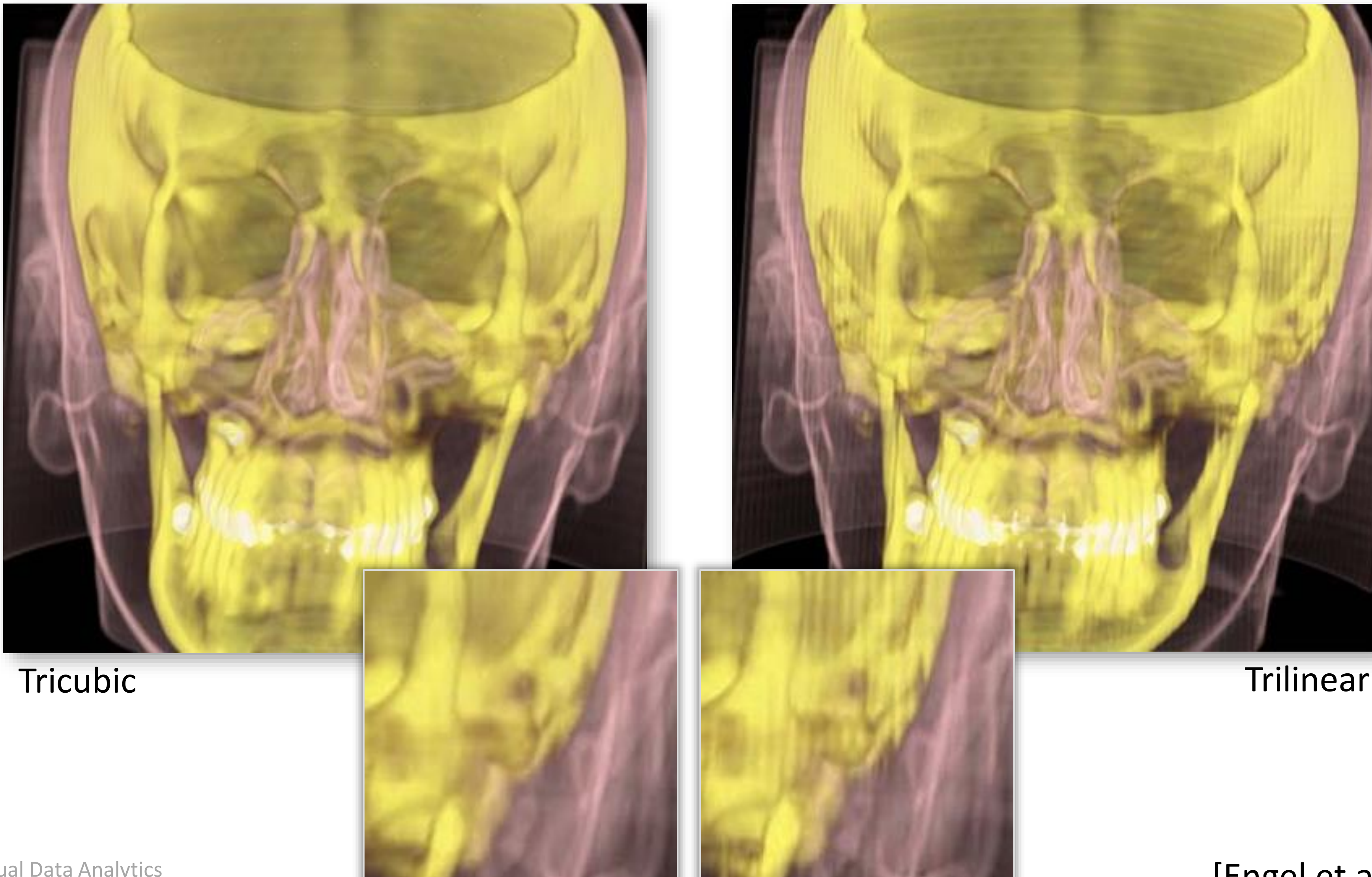
(C^0 continuous → values match at segment transition)

No derivative

[Wikipedia]

Interpolation on grids

- Volume rendering with different reconstruction filters



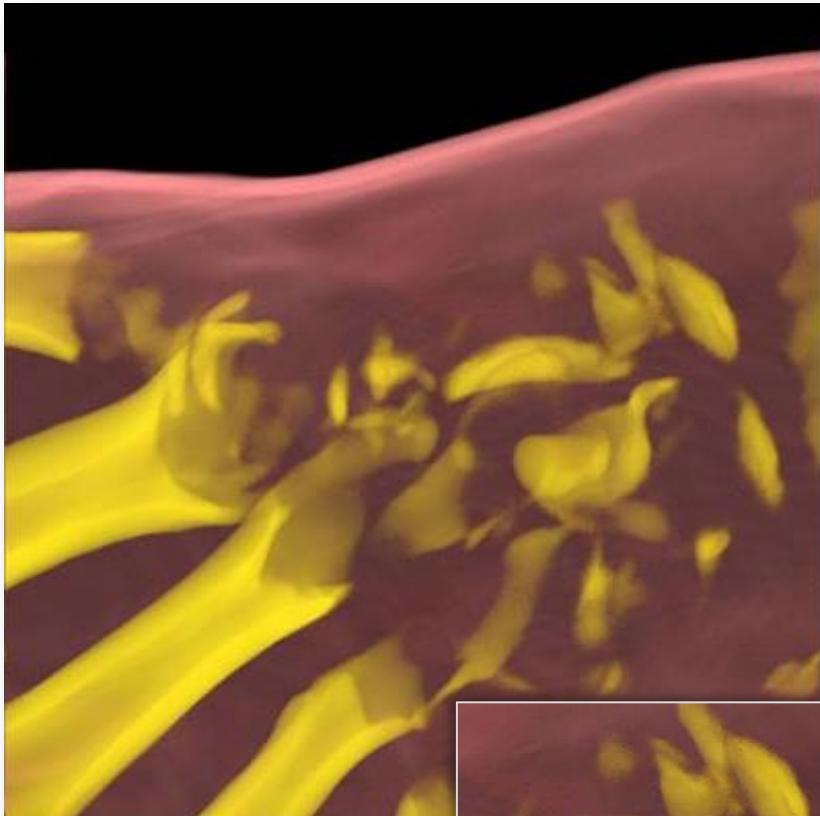
Tricubic

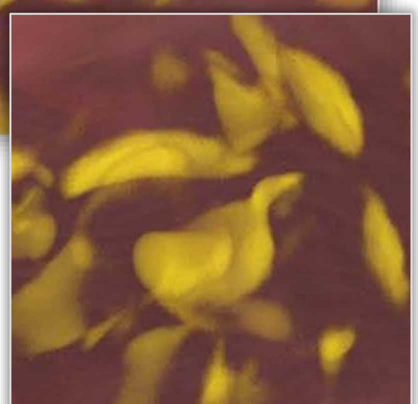
Trilinear

[Engel et al. 06]

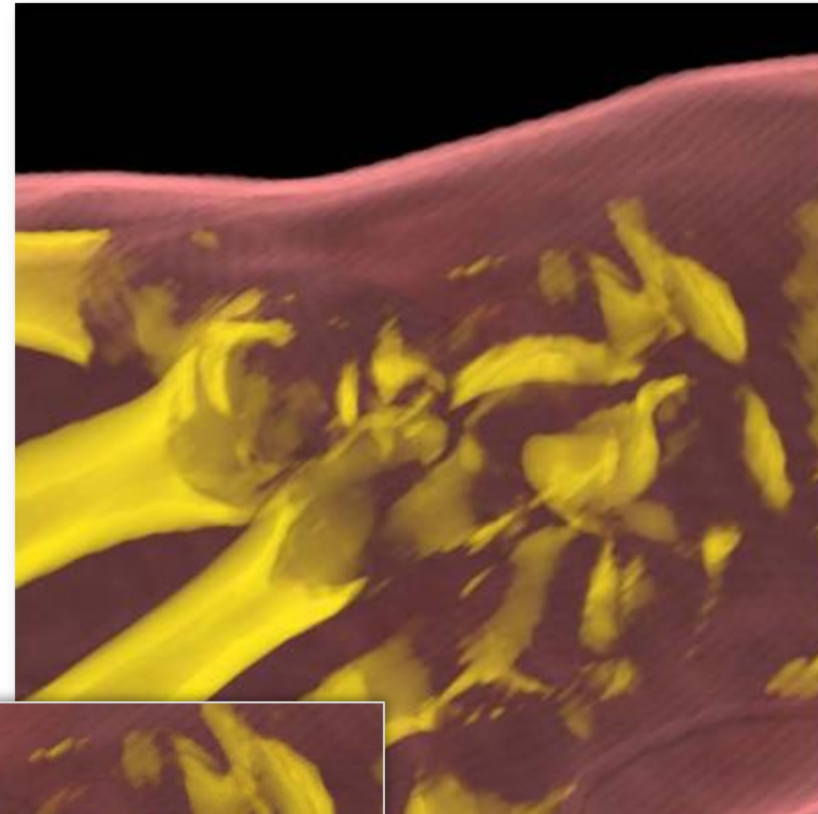
Interpolation on grids

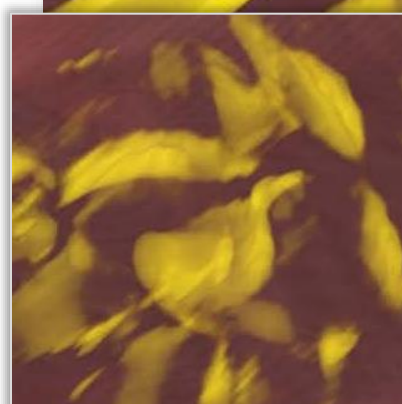
- Volume rendering with different reconstruction filters





Tricubic



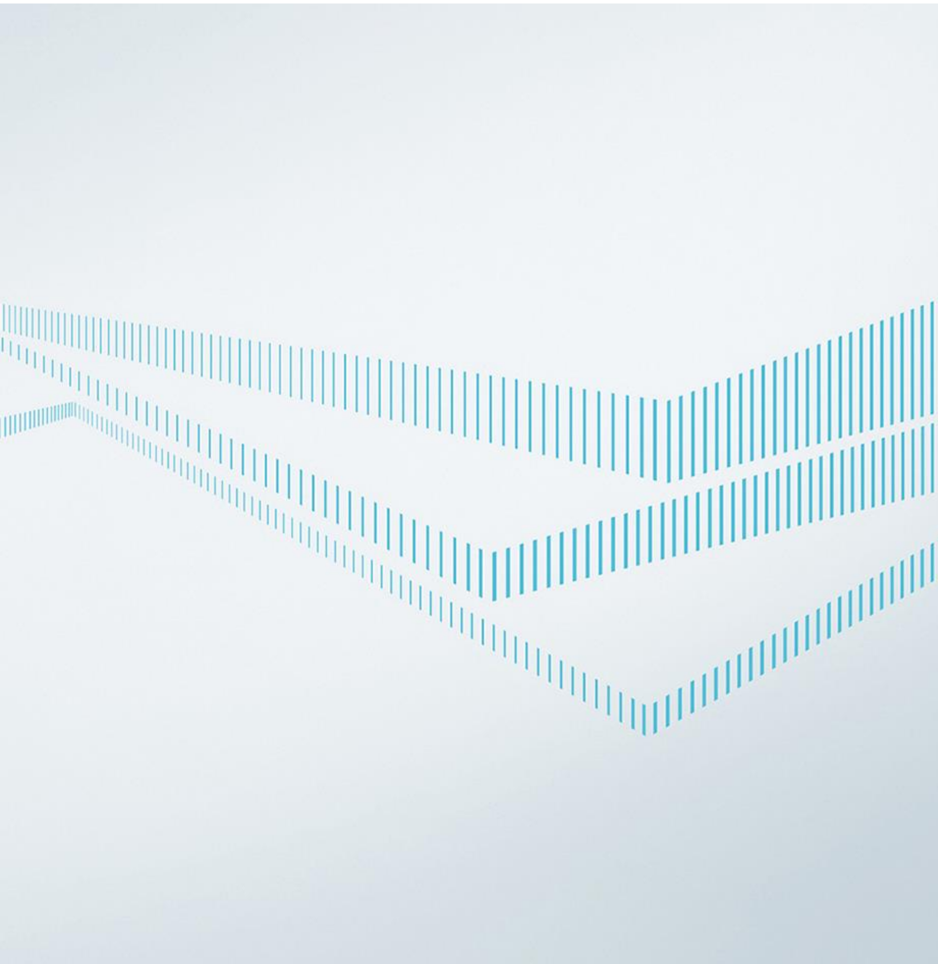


Trilinear

[Engel et al. 06]

Contact information

SIEMENS
Ingenuity for life



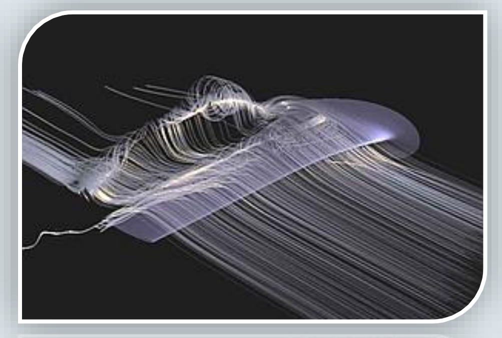
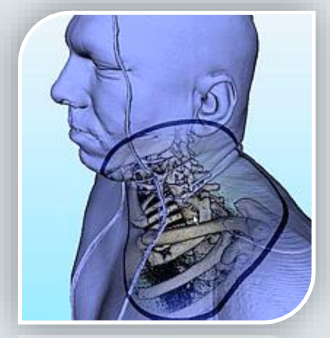
Dr. Johannes Kehrer

Siemens Technology
T DAI HCA-DE
Otto-Hahn-Ring 6
81739 München, Deutschland

E-mail:
kehrer.johannes@siemens.com

Internet
siemens.com/innovation

Intranet
intranet.ct.siemens.com


Visual Data Analytics

Isolines & Isosurfaces

Dr. Johannes Kehrer

Disclaimer

These lecture slides and materials are for personal use by students and are intended for educational purposes only. Distribution or reproduction of the material is prohibited.

The views and opinions expressed by any individual during the course of these lecturers are their own and do not necessarily represent the views, opinions, or positions of Siemens or the Technical University of Munich.

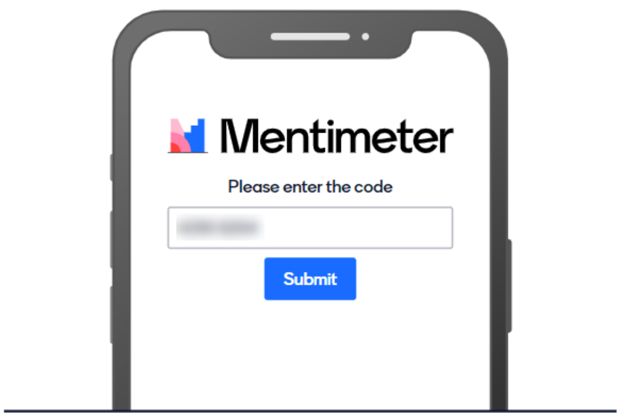
Although reasonable efforts have been made to ensure the information in these lecture slides is current and accurate, we do not assume any responsibility for the content of the posted material.

These slides also contain links to websites of third parties. As the content of these websites is not under our control, we cannot assume any liability for such external content.

Any comments/questions?

Go to

www.menti.com



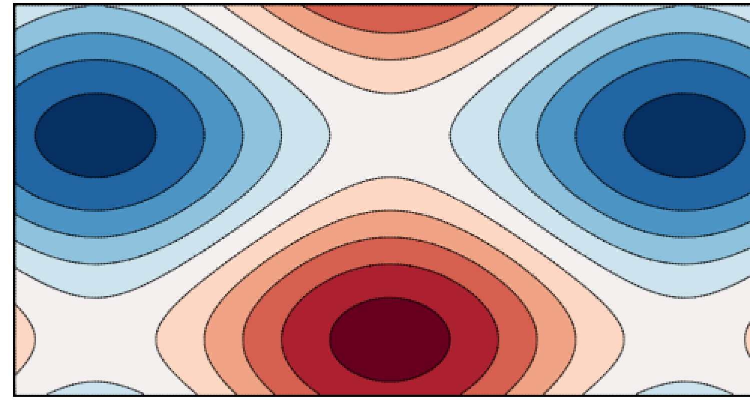
Enter the code

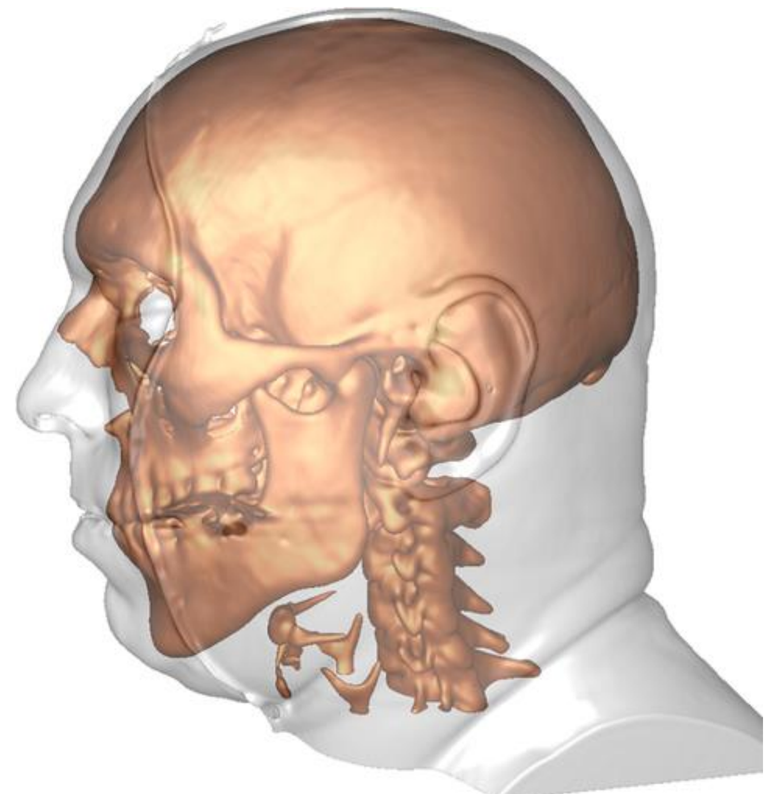
59 72 03 7

Or use QR code

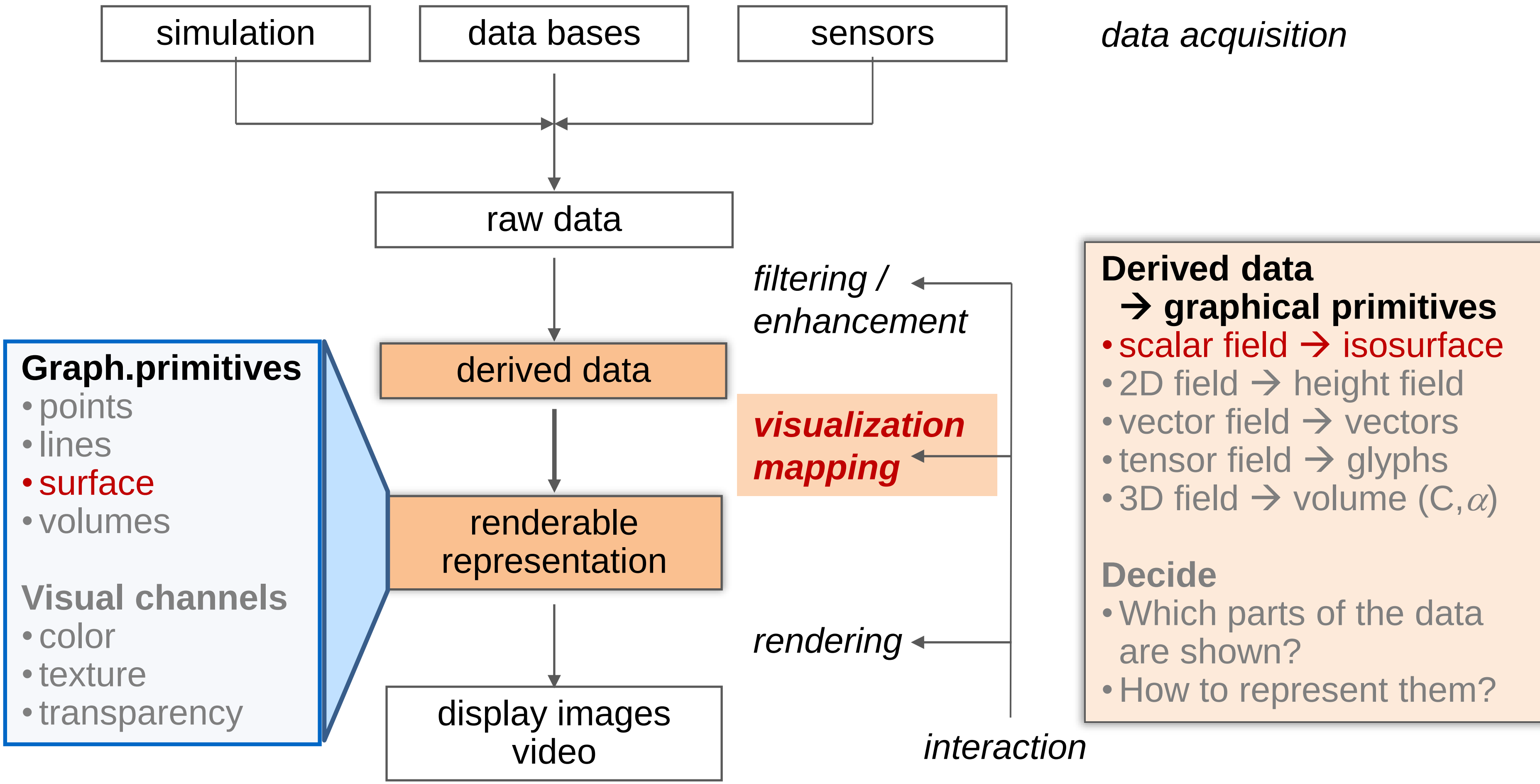
Overview

- Isolines & Isosurfaces
  - Marching squares
  - Marching cubes
- Lighting
  - Phong illumination model
- Gradient approximation



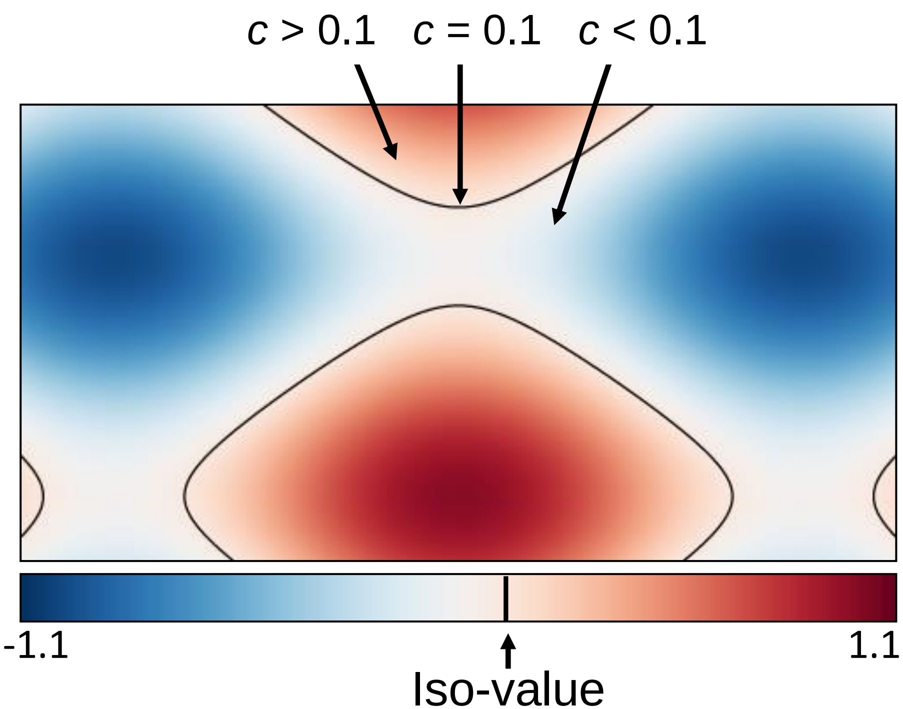


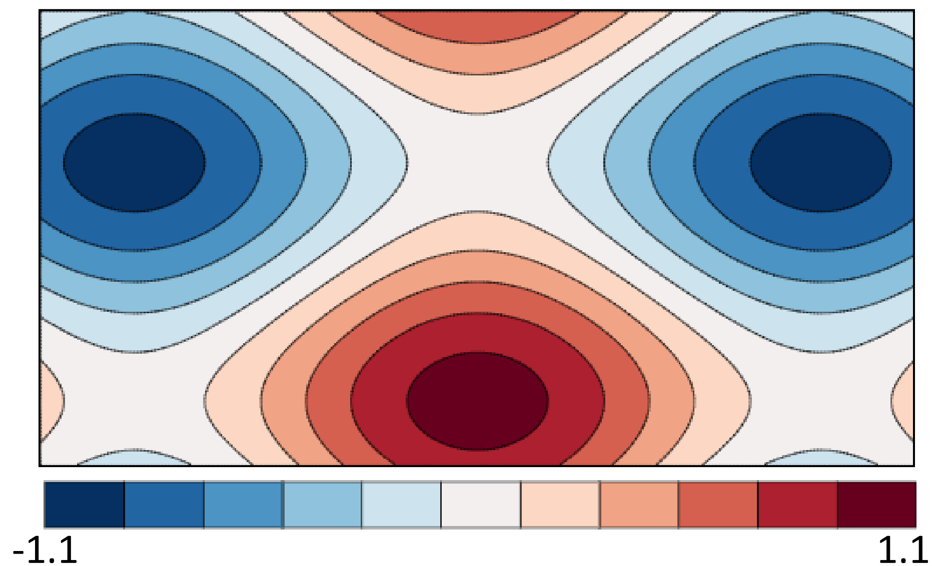
Isolines & Isosurfaces



Isolines & Isosurfaces

- How to see where different values appear?



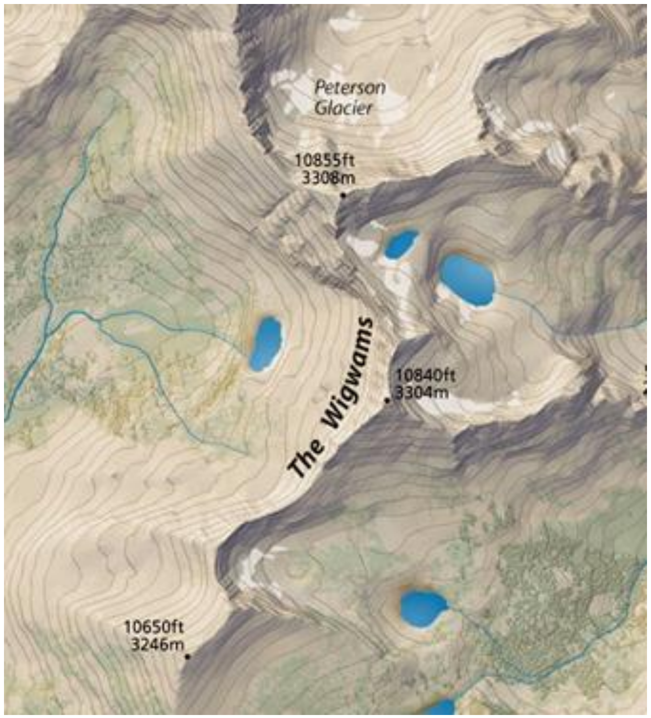


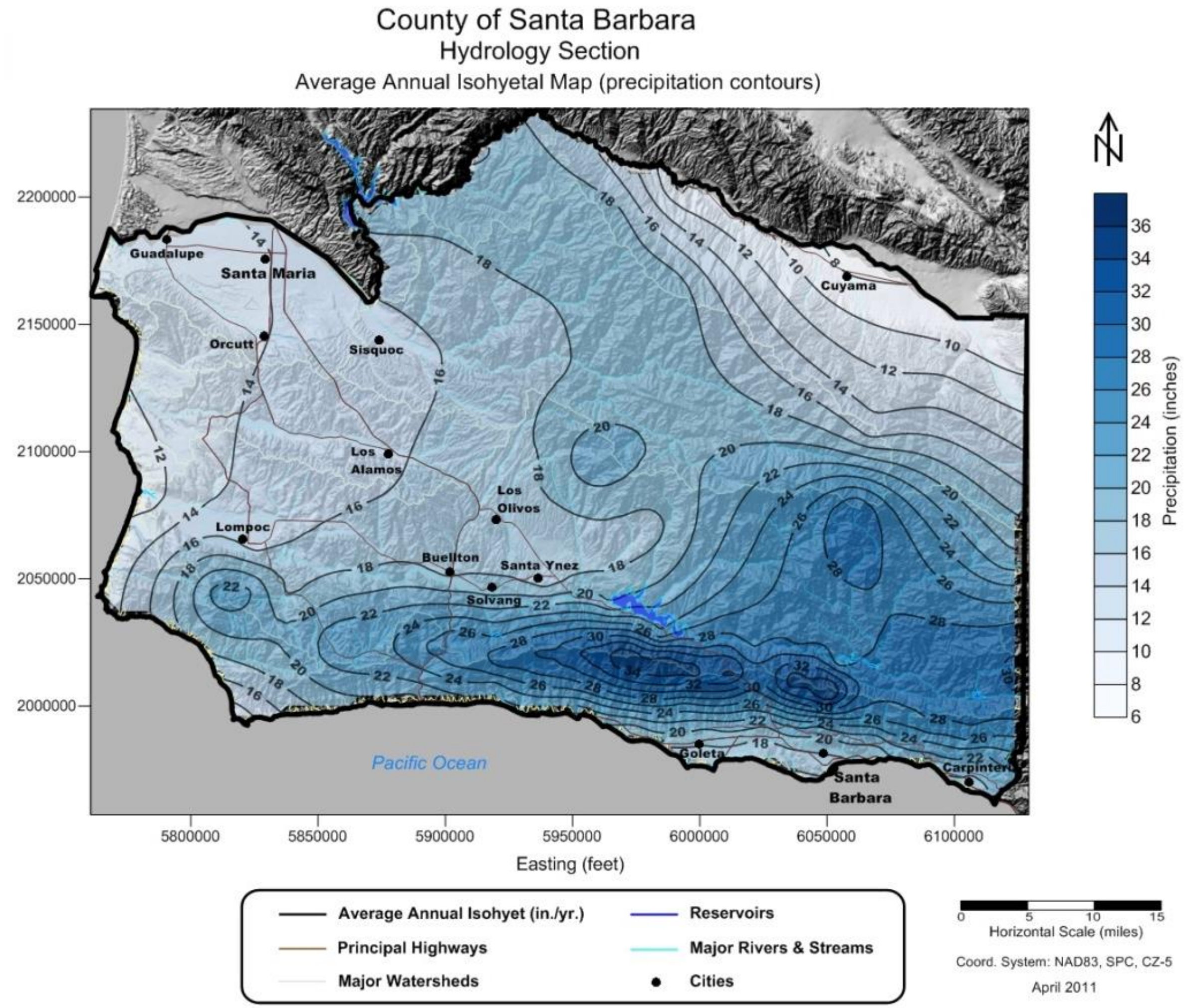
Isoline = all points having the scalar value $c = 0.1$

Ten different isolines, equidistant in value space

Isolines & Isosurfaces

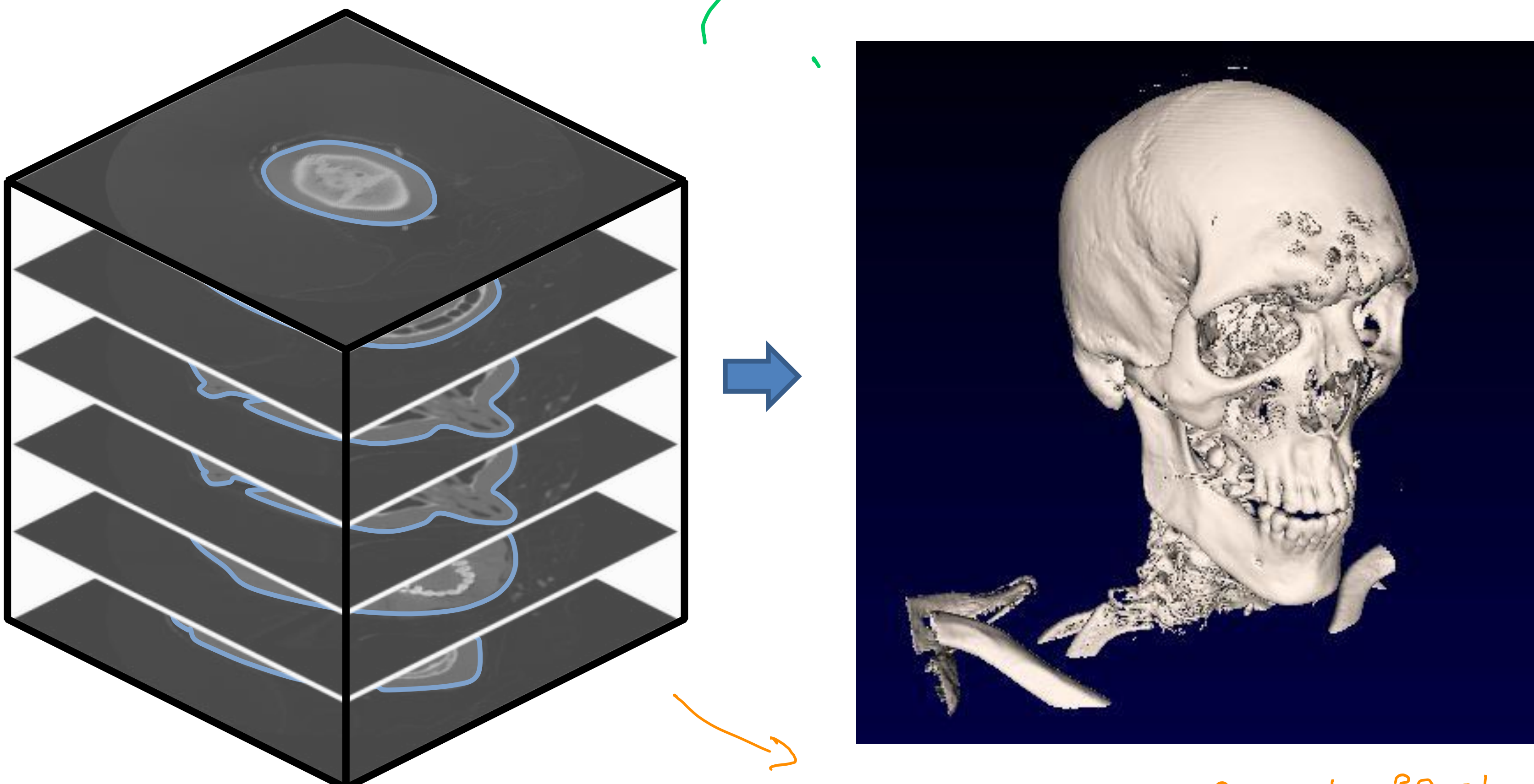
- Known for hundreds of years in cartography





Isolines & Isosurfaces

- Isosurfaces in continuous data fields

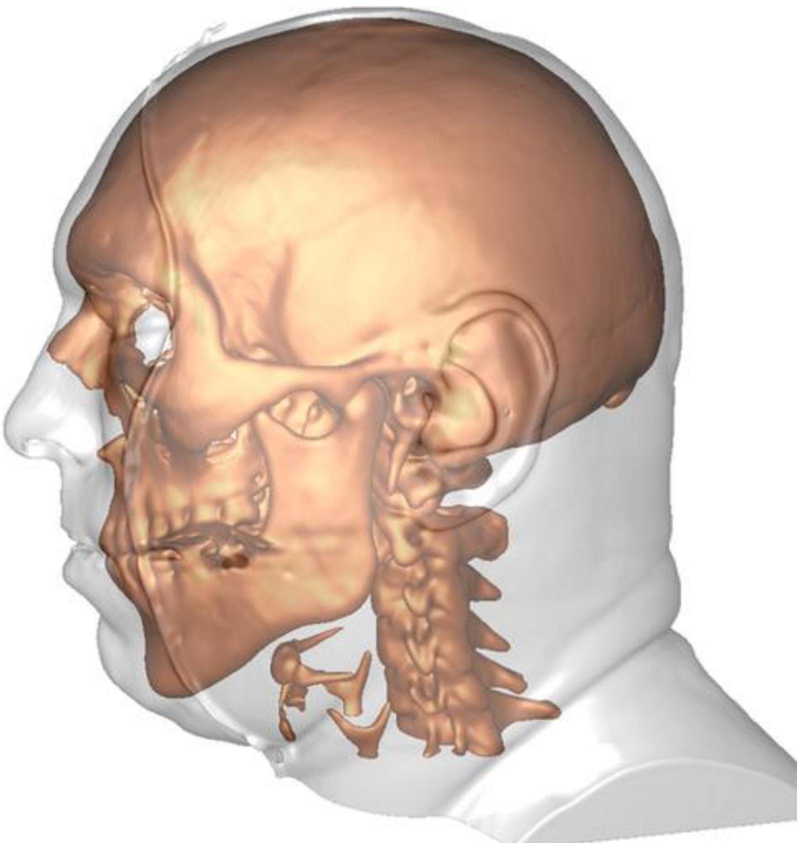


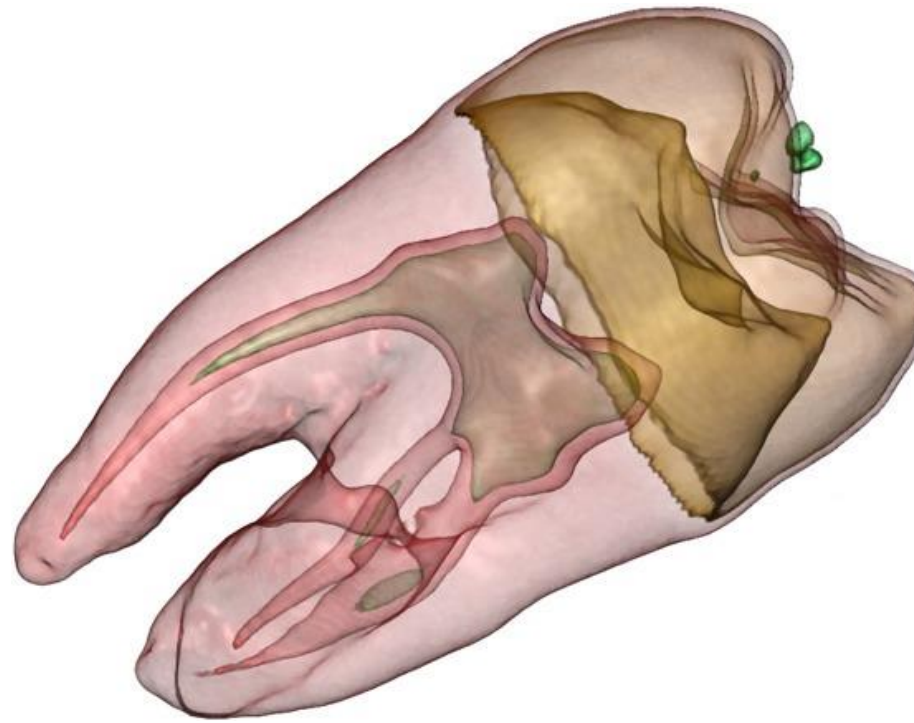
You can construct from isosurfaces to 3D object

Isolines & Isosurfaces

SIEMENS
Ingenuity for life

- Isosurfaces in continuous data fields





Semitransparent isosurfaces
for different iso-values [Kniss 02]

Isolines (2D)

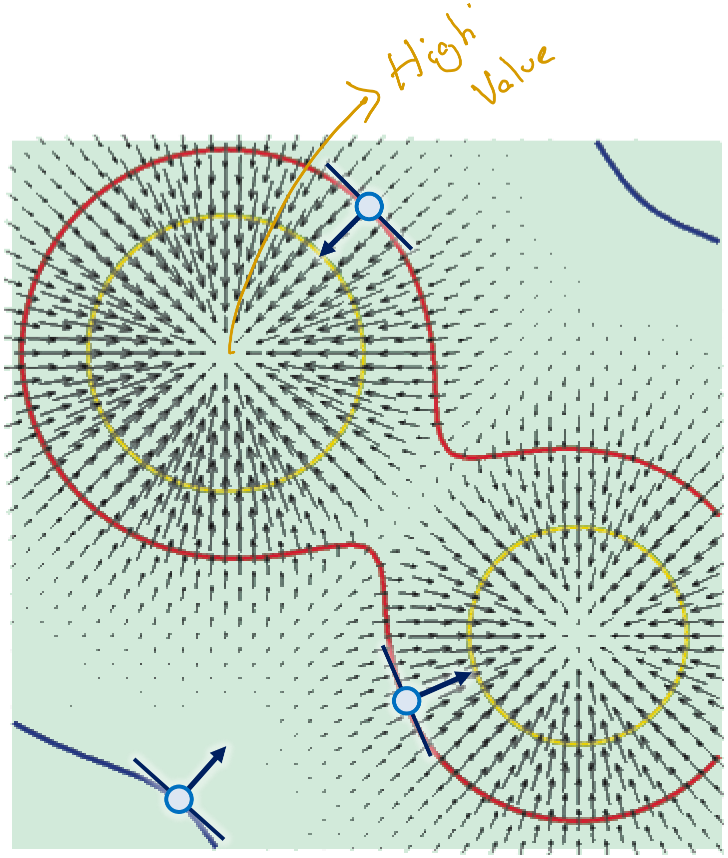
- Given a 2D scalar function $f: \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}$, and a scalar iso-value $c \in \mathbb{R}$
- Map the data to specific isolines
  - An isoline (iso-contour) consists of all points at which the data has a specific value *c*

$$\{(x, y) \mid f(x, y) = c\}$$

  - Can be seen as a special kind of data condensation

Isolines

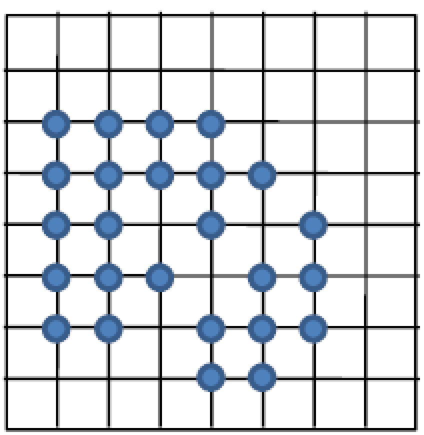
- Properties
  - Isolines are always closed curves (except when they exit the domain)
  - Isolines never (self-)intersect, thus they are nested
    - What would happen if a point belonged to 2 different isolines?
  - Isolines are always orthogonal to the scalar field's gradient

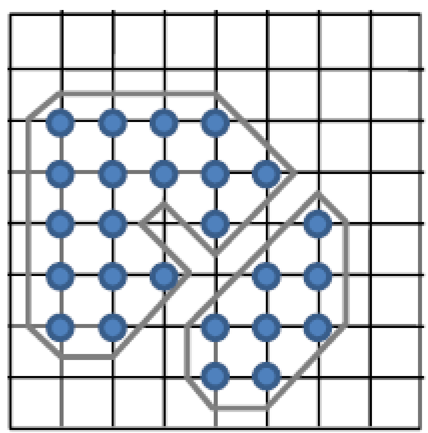


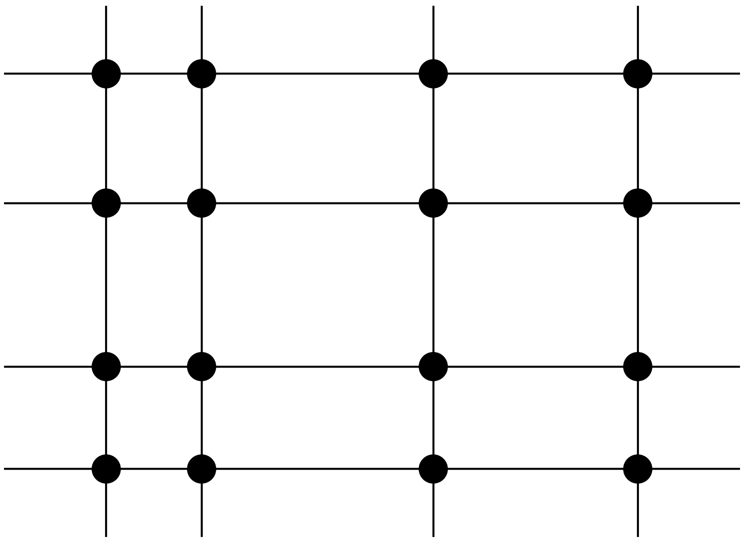
Isolines

SIEMENS
Ingenuity for life

- Computing isolines using marching squares
  - Assumes scalar values given on a rectangular grid
    - Scalar values are given at each vertex $f \leftrightarrow f_{ij}$
  - Divide and conquer: considers each cell independently
  - Takes into account the interpolation along edges
    - Isolines cannot be missed

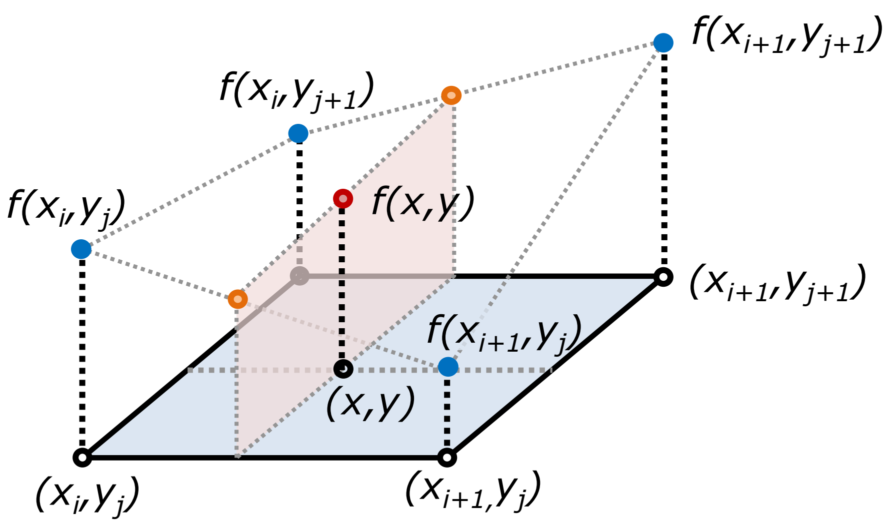


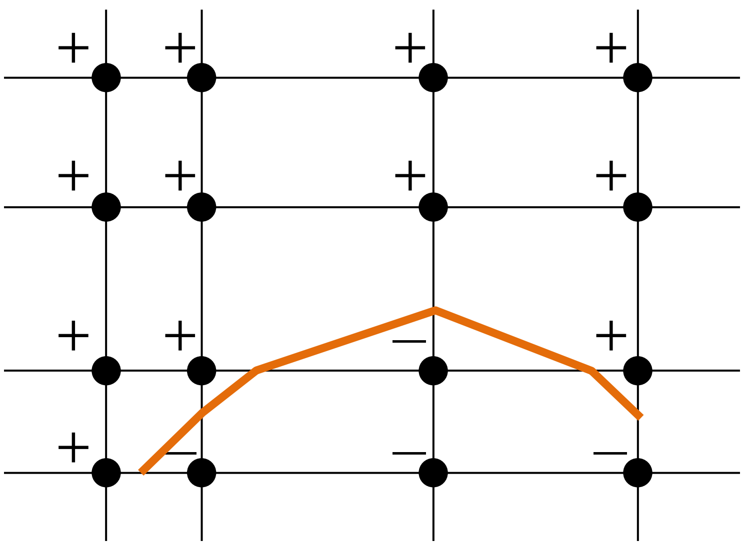




Isolines

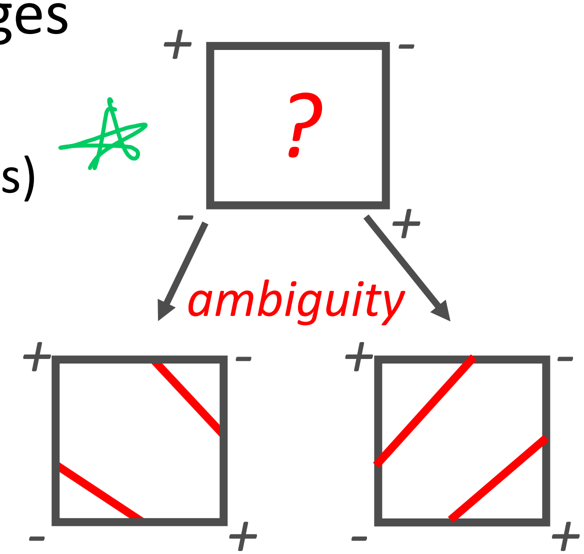
- Which cells have an intersection with the isoline?
  - Initially mark all vertices by – or +, depending on the conditions $f_{i,j} < c$ or $c \leq f_{i,j}$
  - We assume bilinear interpolation inside a cell, thus isoline can not pass through cells with same sign at all 4 vertices
  - Only consider cells having edges with different signs

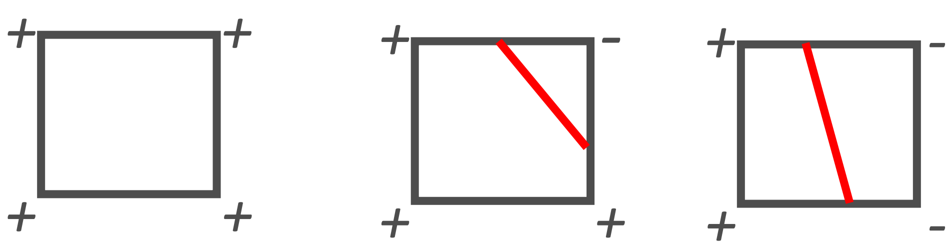




Isolines

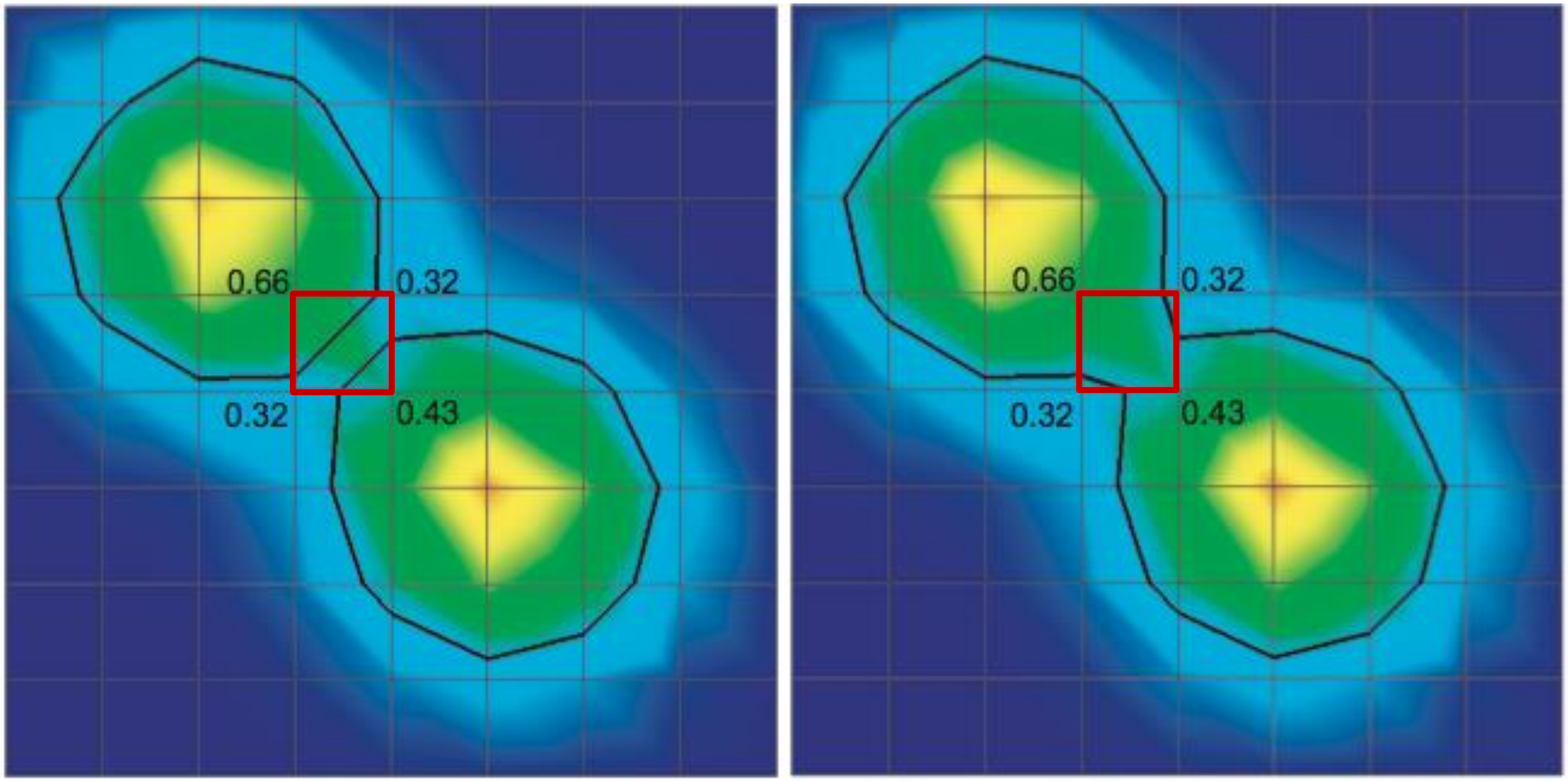
- 16 different sign configurations in 2D
  - Only 4 different base cases
  - Symmetries: rotation, reflection, change + to – and – to +
- Compute intersections between isoline and cell edge
  - Use linear interpolation along the cell edges
  - Connect intersection points via lines
    (even though we know isocontours are hyperbolas)





Isolines

- Ambiguities in surface topology



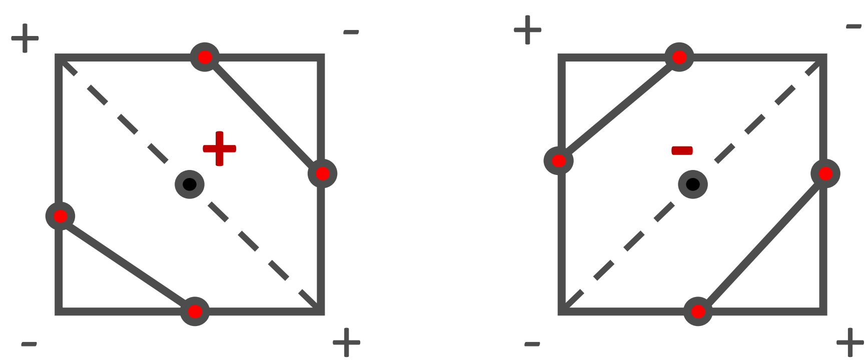
Which is the right isoline result?

Isolines

- We can distinguish (not in all cases) the ambiguous cases by a mid point decider
  – Interpolate the function value in the center

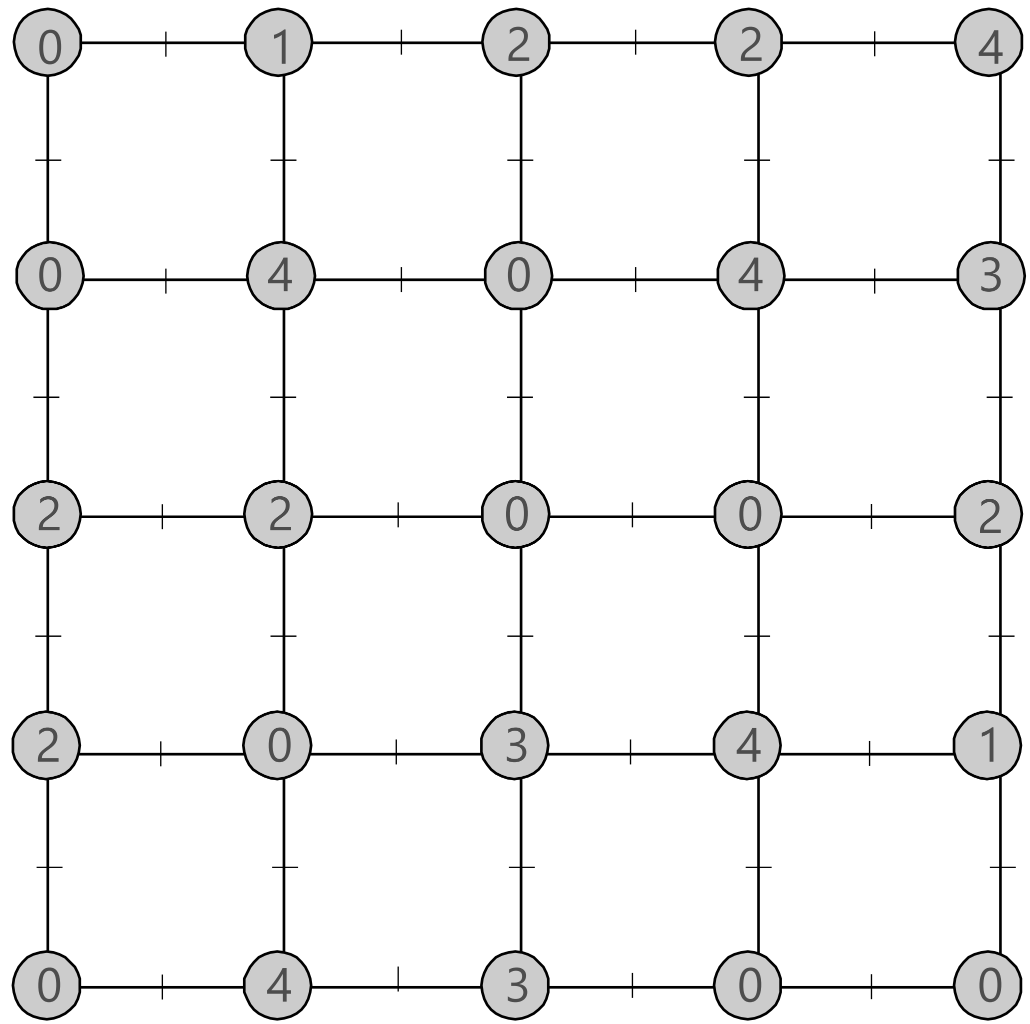
$$f_{\text{center}} = \frac{1}{4}\left(f_{i,j} + f_{i+1,j} + f_{i,\,j+1} + f_{\mathrm{i+1,\,j+1}}\right)$$

  – If $f_{\text{center}} < c$ we choose the right case, otherwise we choose the left case



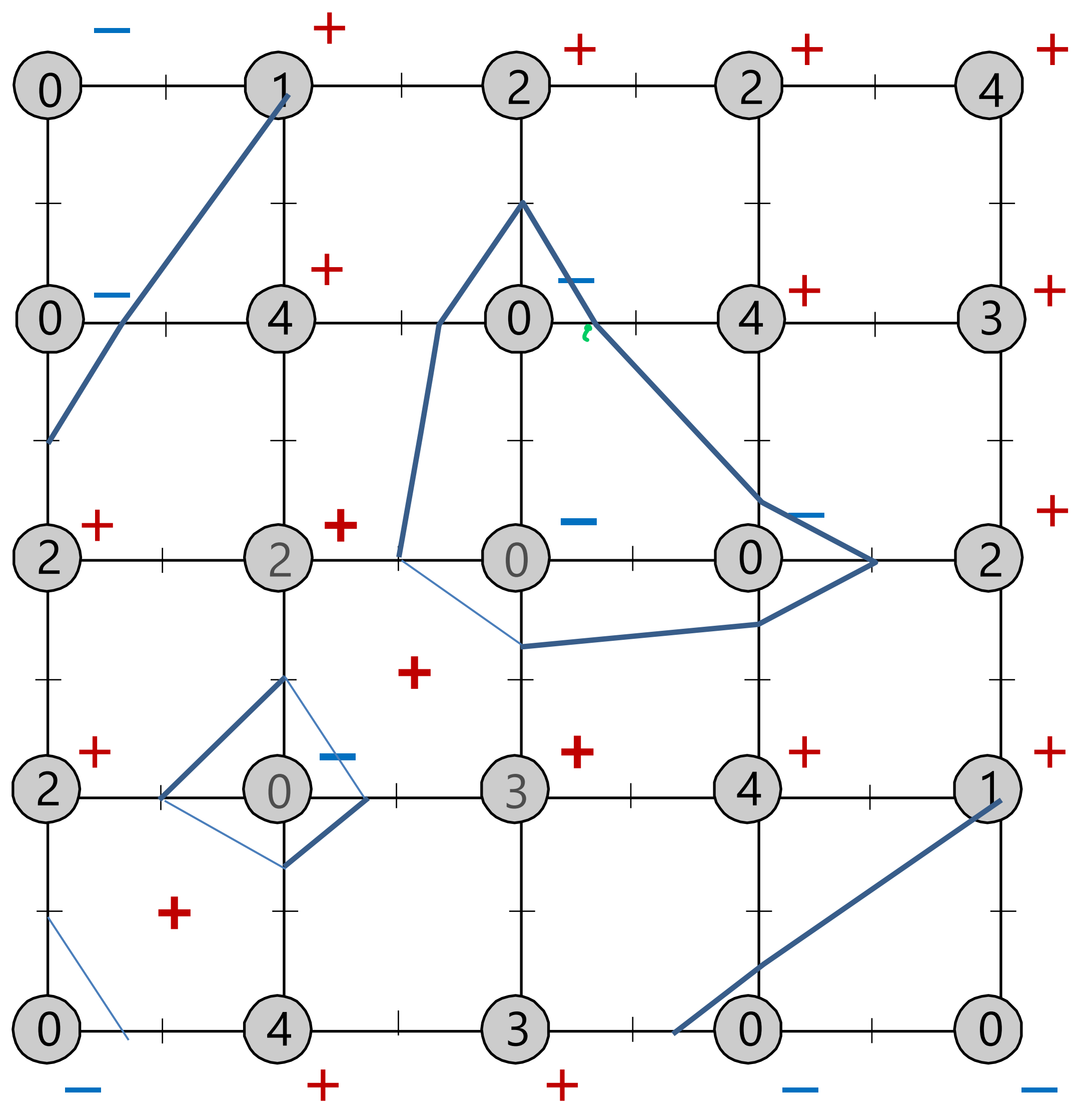
Example

iso-value 1



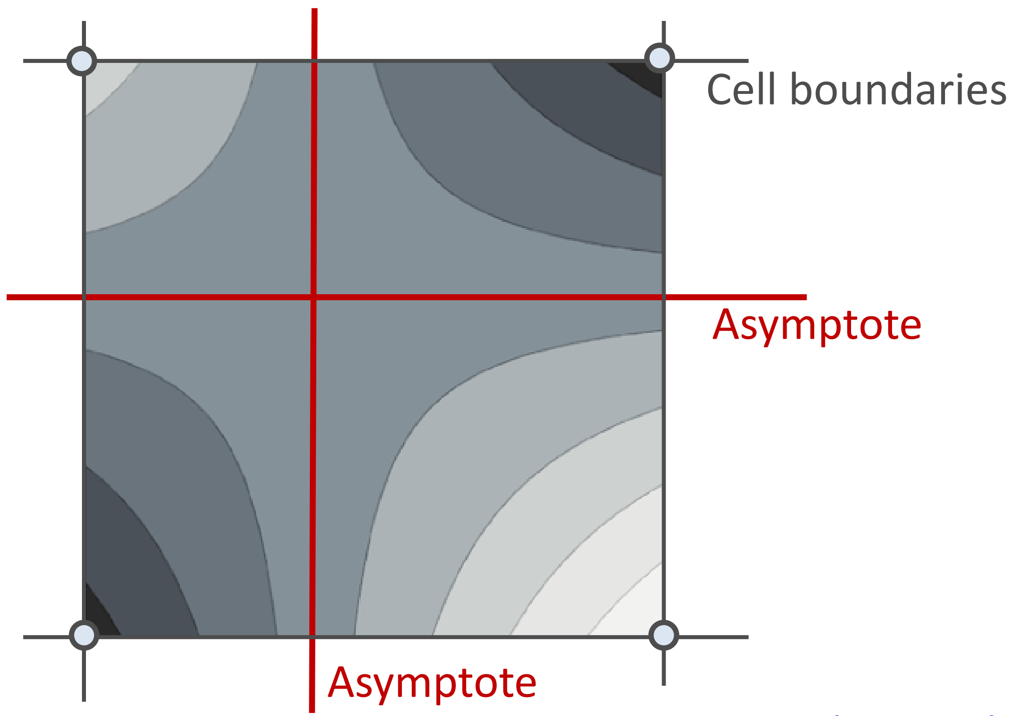
Example

iso-value 1



Isolines

- An exact decider is the asymptotic decider
  - Considers bilinear interpolation within a cell
  - The true isolines within a cell are hyperbolas
  - Value at intersection of asymptotes decides connectivity



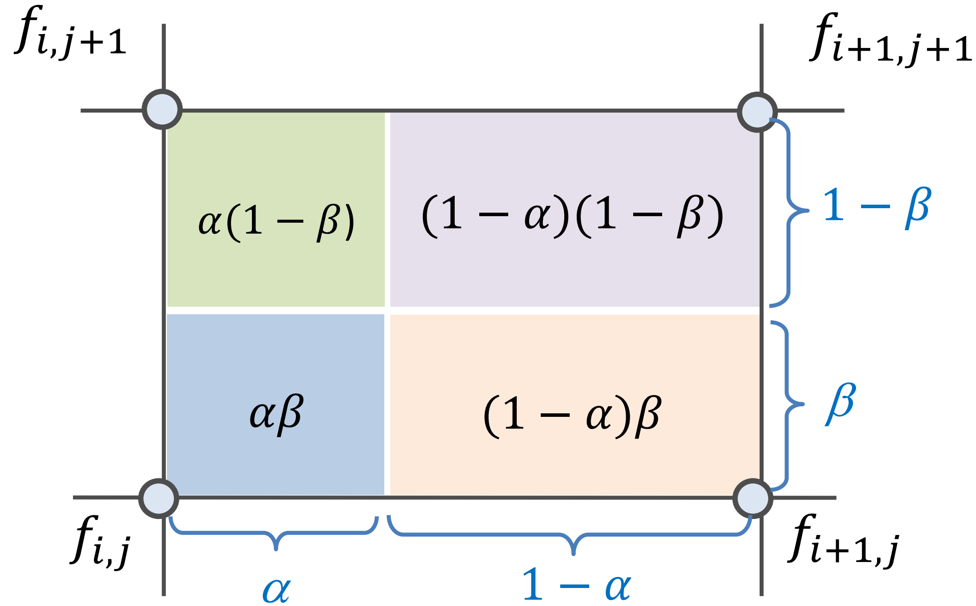
multivis.net/lecture/bilinear.html

Isolines

- How to evaluate the asymptotic decider?
  - Consider bilinear interpolation within a grid cell

$$f(\alpha,\beta) = (1-\alpha)(1-\beta)f_{i,j} + \alpha(1-\beta)f_{i+1,j} + (1-\alpha)\beta f_{i,\,j+1} + \alpha\beta f_{\mathrm{i+1,\,j+1}}$$

$\alpha, \beta \in [0, 1]$



multivis.net/lecture/bilinear.html

Isolines

SIEMENS
Ingenuity for life

- How to evaluate the asymptotic decider?
  - Consider bilinear interpolation within a grid cell

$$f(\alpha,\beta) = (1-\alpha)(1-\beta)f_{i,j} + \alpha(1-\beta)f_{i+1,j} + (1-\alpha)\beta f_{i,\,j+1} + \alpha\beta f_{\mathrm{i+1,\,j+1}}$$

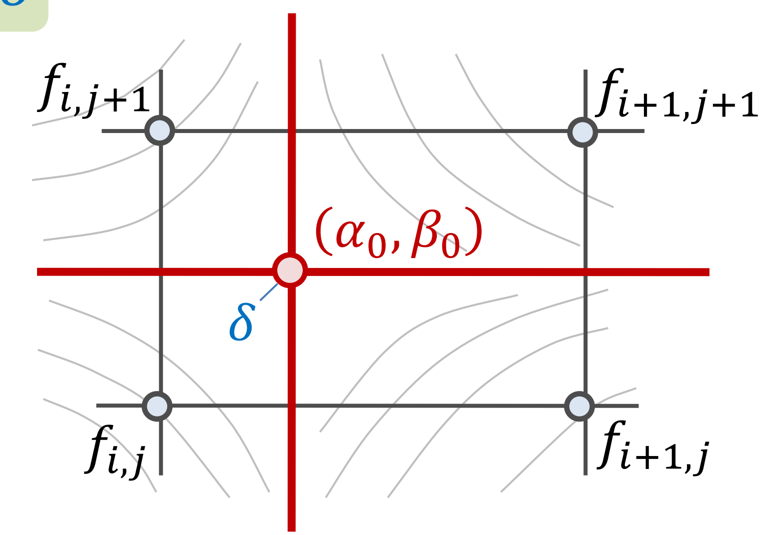
$\alpha, \beta \in [0, 1]$

  - Given the values at cell corners, transform f to

$$f(\alpha,\beta) = \gamma(\alpha - \alpha_0)(\beta - \beta_0) + \delta$$

Function of a hyperbola

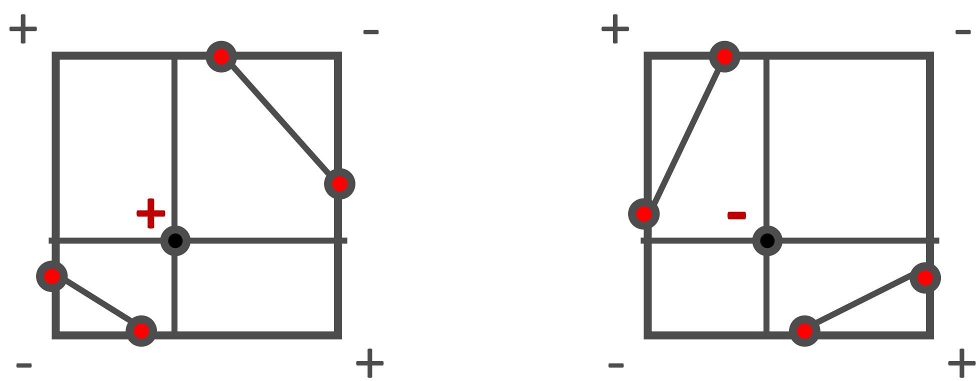
  - δ is the function value at the intersection point (α_0, β_0) of the asymptotes



multivis.net/lecture/bilinear.html

Isolines

- How to evaluate the asymptotic decider?
  - If $\delta < c$ we chose the right case, otherwise we chose the left one



Isolines

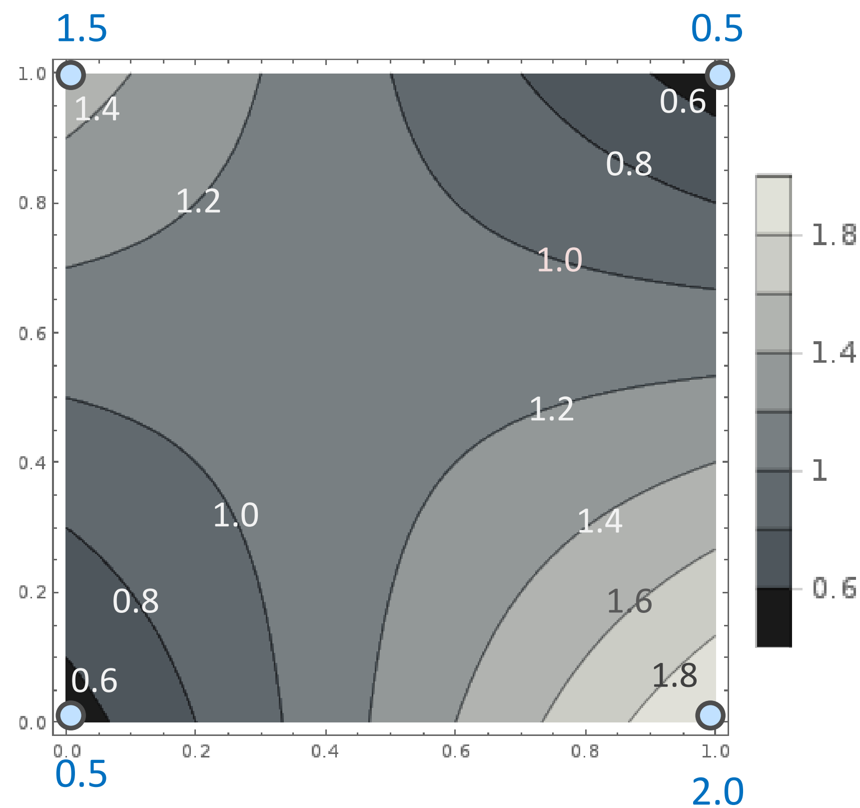
- How to evaluate the asymptotes?

  Compute asymptotes from

  $$f(\alpha,\beta) = 0.5 + 1.5\alpha + \beta - 2.5\alpha\beta$$

  Get into form of hyperbola

  $$f(\alpha,\beta) = \gamma(\alpha - \alpha_0)(\beta - \beta_0) + \delta$$



  – Where do the asymptotes intersect?

  – What is the value at the intersection?

multivis.net/lecture/bilinear.html

Isolines

- How to evaluate the asymptotes?

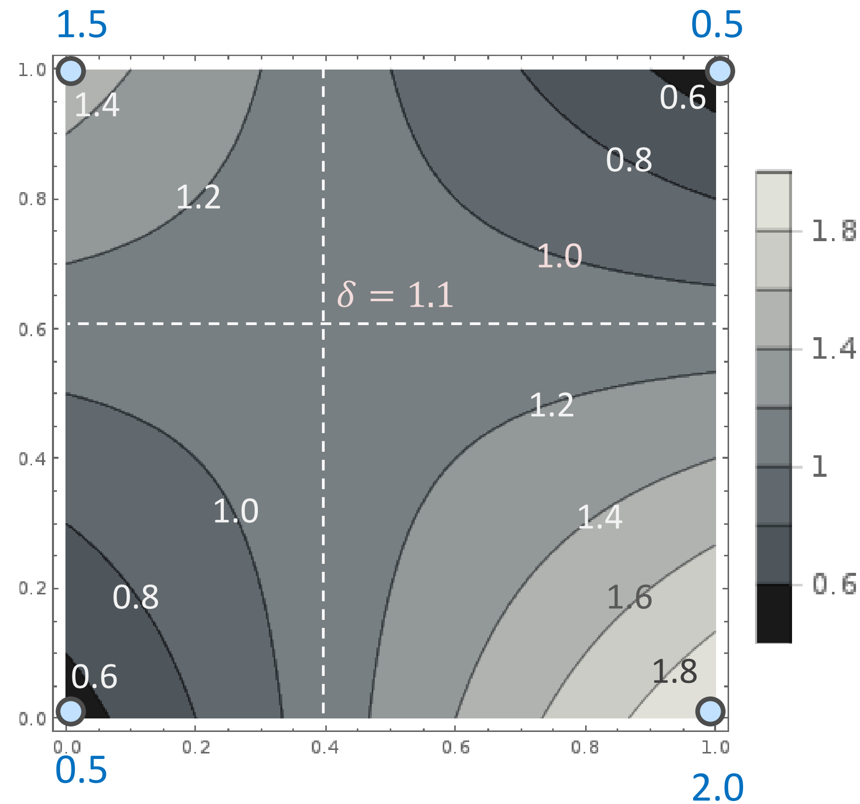
  Compute asymptotes from

  $$f(\alpha,\beta) = 0.5 + 1.5\alpha + \beta - 2.5\alpha\beta$$

  Get into form of hyperbola

  $$f(\alpha,\beta) = \gamma(\alpha-\alpha_0)(\beta-\beta_0)+\delta$$
  $$= \gamma(\alpha\beta - \beta_0\alpha - \alpha_0\beta + \alpha_0\beta_0) + \delta$$

  $$\rightarrow f(\alpha,\beta) =$$
  $$-2.5\left(\alpha - \frac{1}{2.5}\right)\left(\beta - \frac{1.5}{2.5}\right) + 0.5 + 0.6$$
  $$= -2.5(\alpha - 0.4)(\beta - 0.6) + 1.1$$



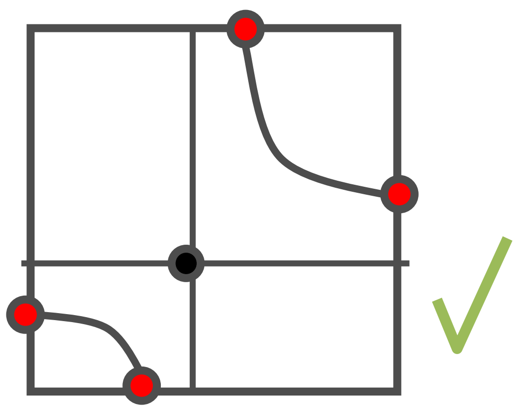
  – Asymptotes intersect at $(0.4, 0.6)$
  – Value at intersection: $\delta = 1.1$

multivis.net/lecture/bilinear.html

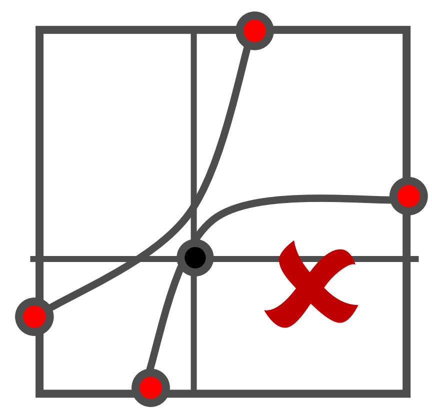
Isolines

SIEMENS
Ingenuity for life

- How to evaluate the asymptotic decider?
  - We can avoid the evaluation of δ by investigating the order of intersection points along either axis
  - Build pairs of first two and last two intersections and connect these pairs



valid

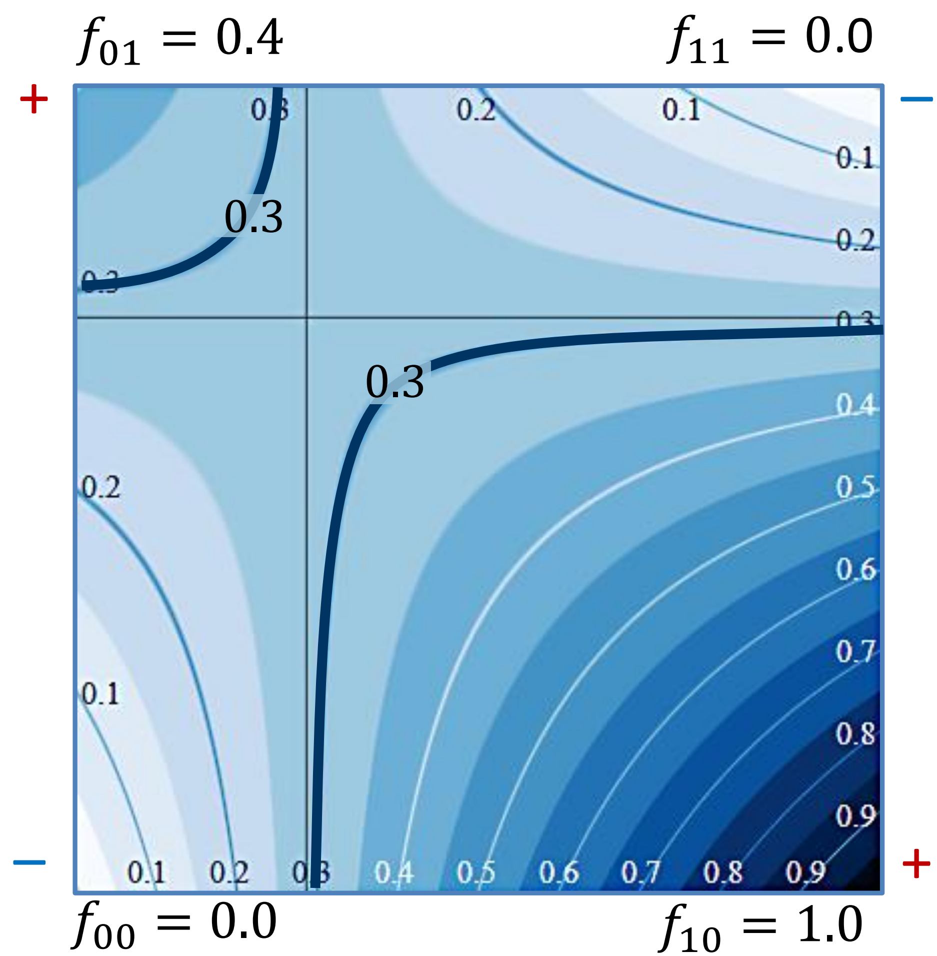


not possible

Isolines

SIEMENS
Ingenuity for life

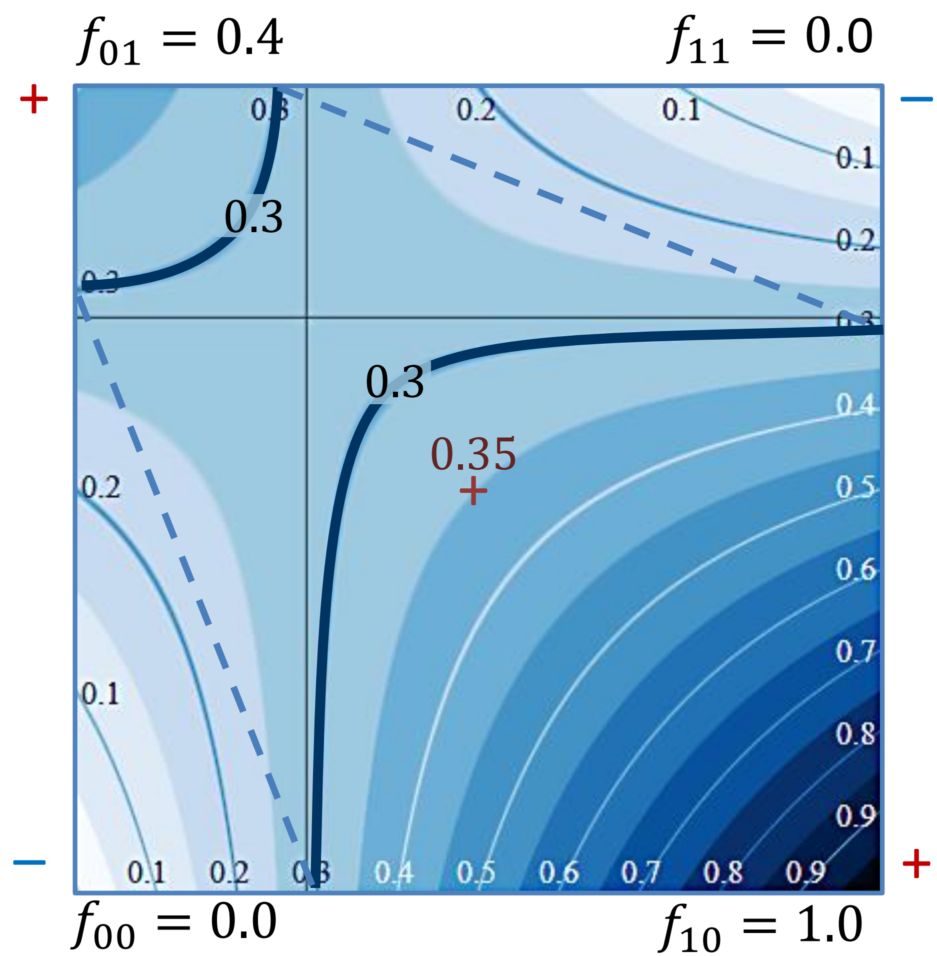
- When does the midpoint decider **not** work
  - **Example:** Iso-line for $c = 0.3$



multivis.net/lecture/bilinear.html

Isolines

- When does the midpoint decider **not** work
  - **Example:** Iso-line for $c = 0.3$
  - Value at midpoint: 0.35

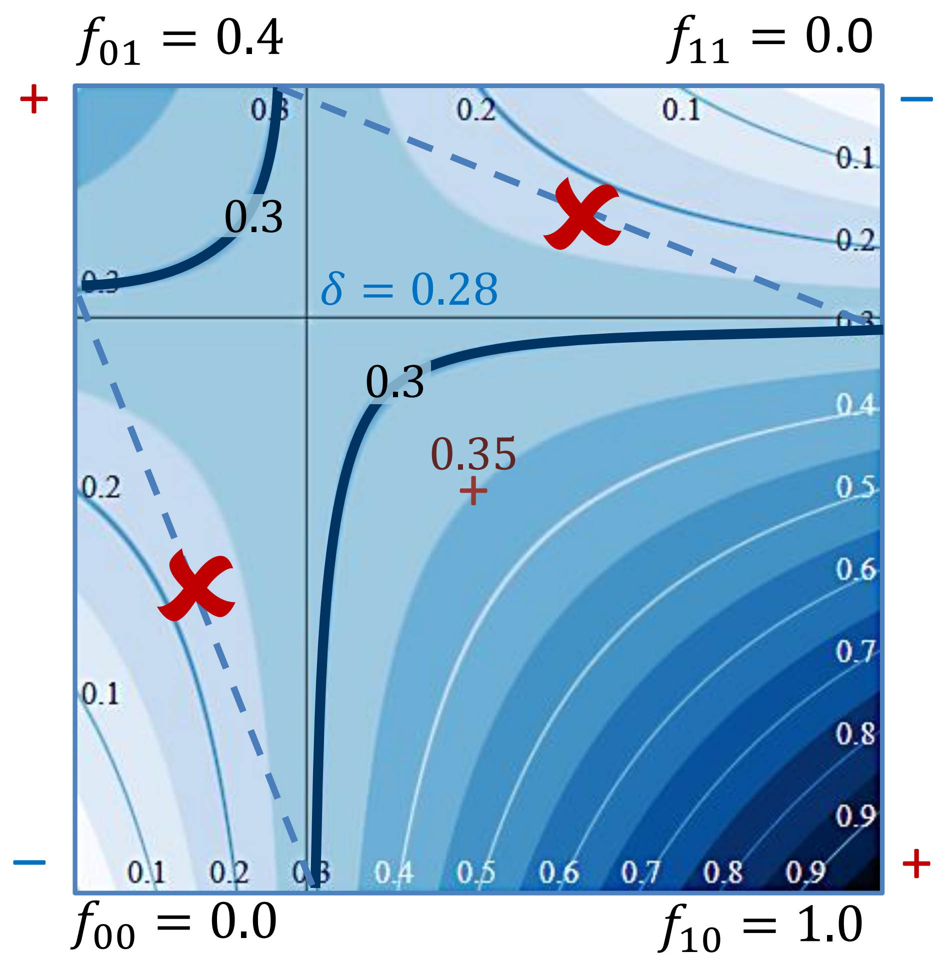


multivis.net/lecture/bilinear.html

Isolines

SIEMENS
Ingenuity for life

- When does the midpoint decider **not** work
  - **Example:** Iso-line for $c = 0.3$
  - Value at midpoint: 0.35
  - Evaluating the asymptotic decider, we get $\delta = 0.28$ at intersection of asymptotes
  - The midpoint decider would give us wrong isolines



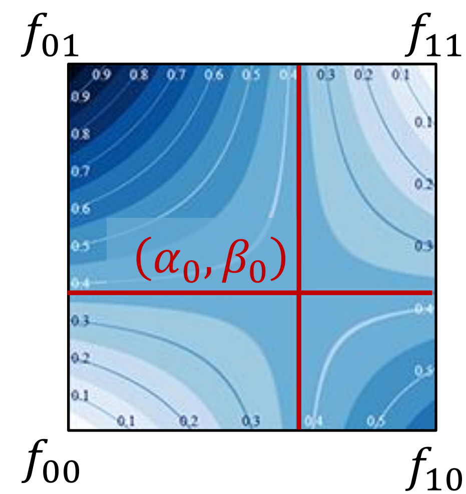
multivis.net/lecture/bilinear.html

Isolines

- Summary: Bilinear Interpolation
  - The value at each point (α, β) within the cell can be obtained by

$$f(\alpha, \beta) = (1-\alpha)(1-\beta)f_{00} + \alpha(1-\beta)f_{10} + (1-\alpha)\beta f_{01} + \alpha\beta f_{11}$$
$$= A\alpha + B\beta + C\alpha\beta + D$$

where $A = f_{10} - f_{00}$, $B = f_{01} - f_{00}$,
$C = f_{00} - f_{01} - f_{10} + f_{11}$, $D = f_{00}$



  - We can evaluate the isoline for an iso-value c by setting $f(\alpha, \beta) = c$
  - The asymptotes of the hyperbolas can be computed by

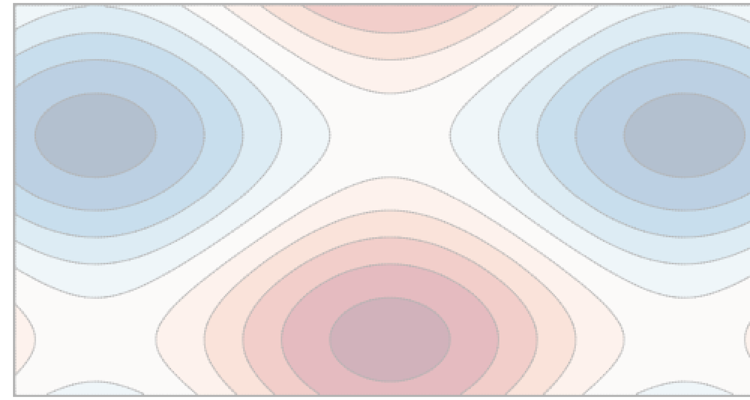
$$f(\alpha, \beta) = \gamma(\alpha - \alpha_0)(\beta - \beta_0) + \delta$$

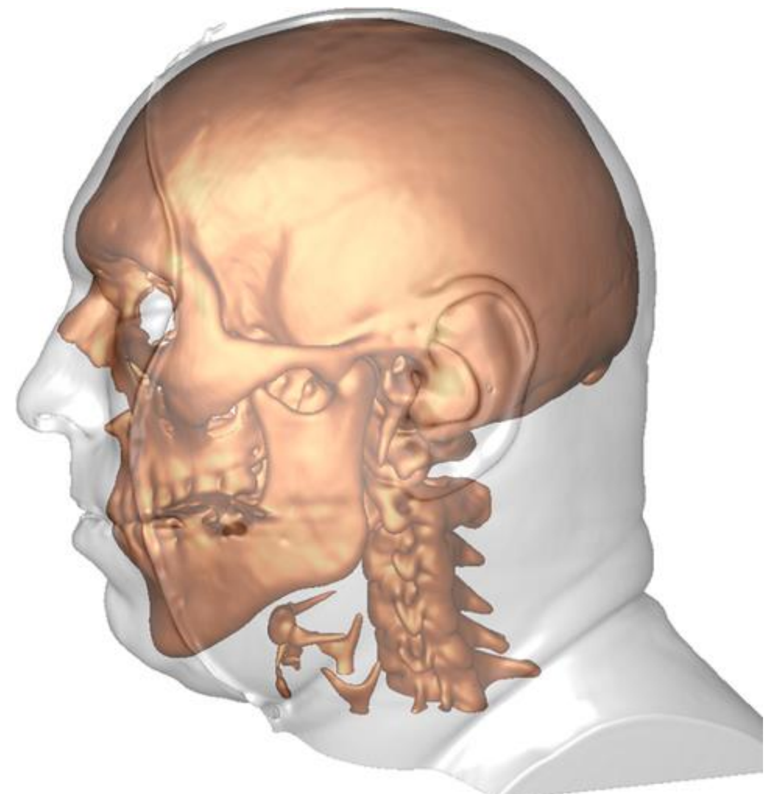
where $\gamma = C$ and $\delta = (f_{00}\, f_{11} - f_{01}\, f_{10})/C$ is the value at the intersection point (α_0, β_0) of the asymptotes with $\alpha_0 = -B/C$ and $\beta_0 = -A/C$

multivis.net/lecture/bilinear.html

Overview

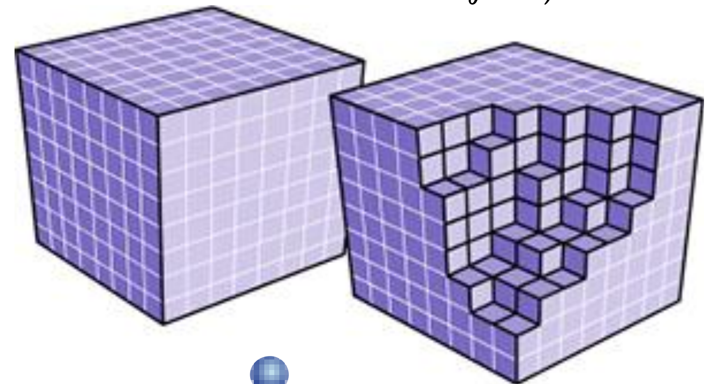
- Isolines & Isosurfaces
  - Marching squares
  - Marching cubes
- Lighting
  - Phong illumination model
- Gradient approximation



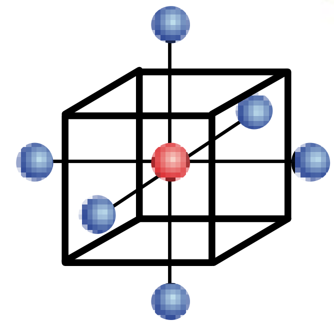


Volume visualization

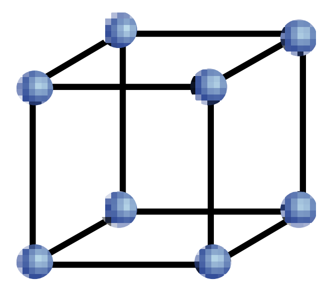
SIEMENS
Ingenuity for life



- Data values are initially given at vertices of a 3D grid
  – These are called voxels (volume elements)
  – Voxel = point sample in 3D



- “Adjacent” grid vertices make up a cell
  – Use interpolation for data reconstruction in a cell
  – Corners: 8 voxel



Surface reconstruction in 3D

SIEMENS
Ingenuity for life

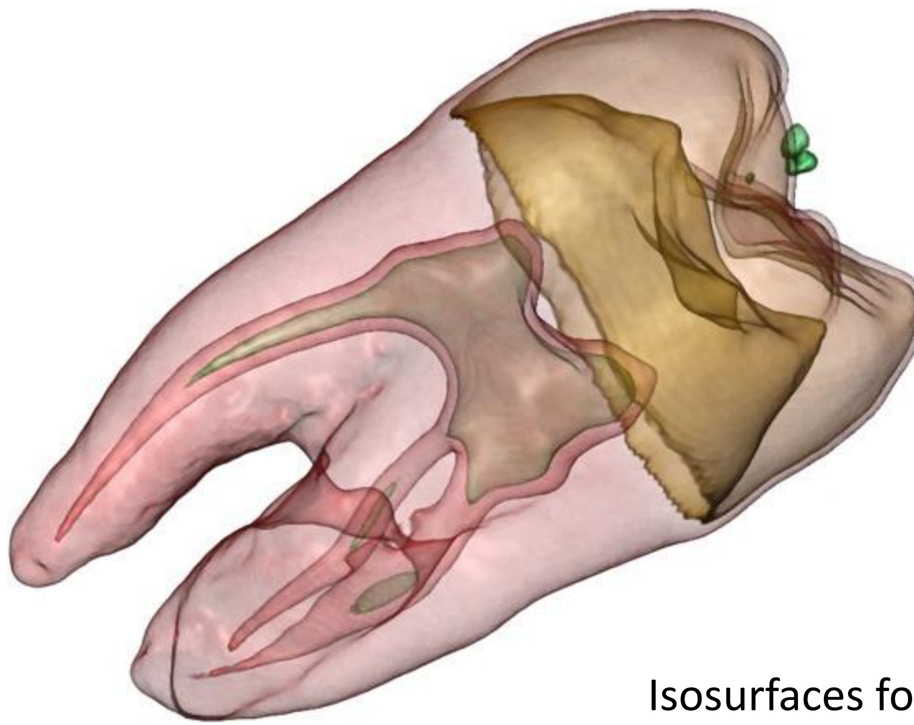
- Marching-Cubes (MC) algorithm
  - Invented by Lorensen and Cline 1987
  - Approximates the surface by a triangle mesh
  - Surface vertices are found by linear interpolation along cell edges
  - Efficient triangulation by means of lookup tables

- The standard geometry-based surface extraction algorithm for 3D scalar field

Marching Cubes

- Computes isosurface for specific iso-value

- Tasks
  - Shape understanding
  - Spatial relationships



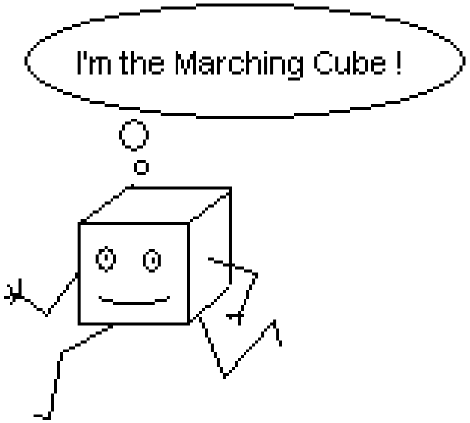
Isosurfaces for different iso-values [Kniss 02]

Marching Cubes

SIEMENS
Ingenuity for life

- The MC algorithm
  - Cell consists of 8 vertices
  - indices: (i+[0,1], j+[0,1], k+[0,1])

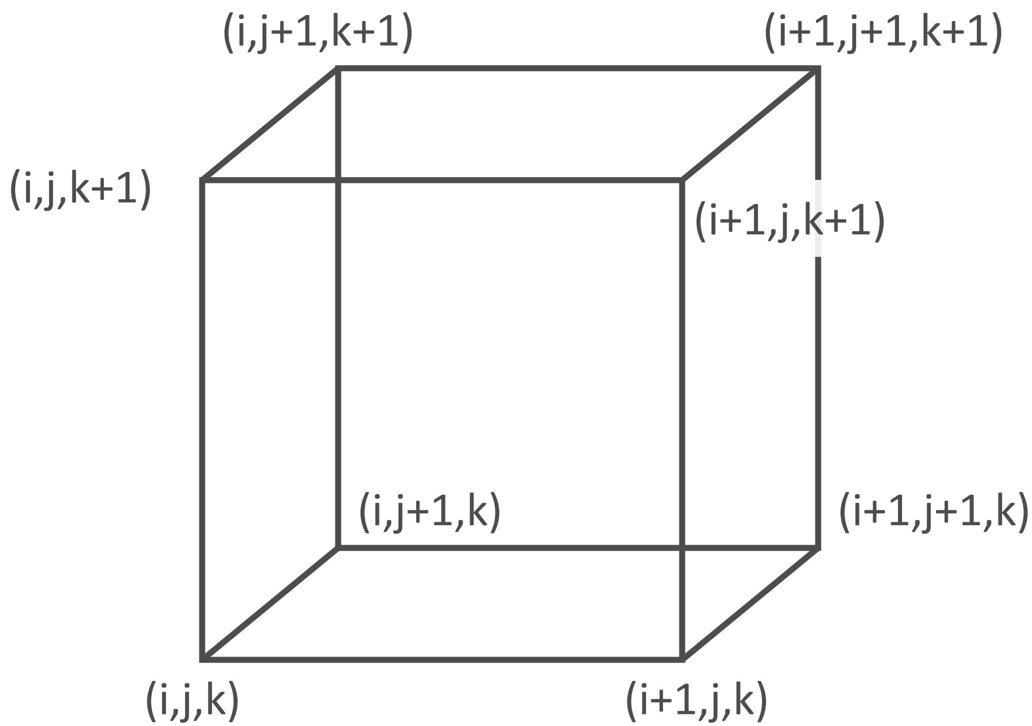
1. Consider a cell independently
2. Classify each vertex as inside or outside
3. Build an index
4. Get per-cell triangulation from index
5. Interpolate the edge location
6. Compute gradients
7. Consider ambiguous cases
8. Go to next cell



Marching Cubes

SIEMENS
Ingenuity for life

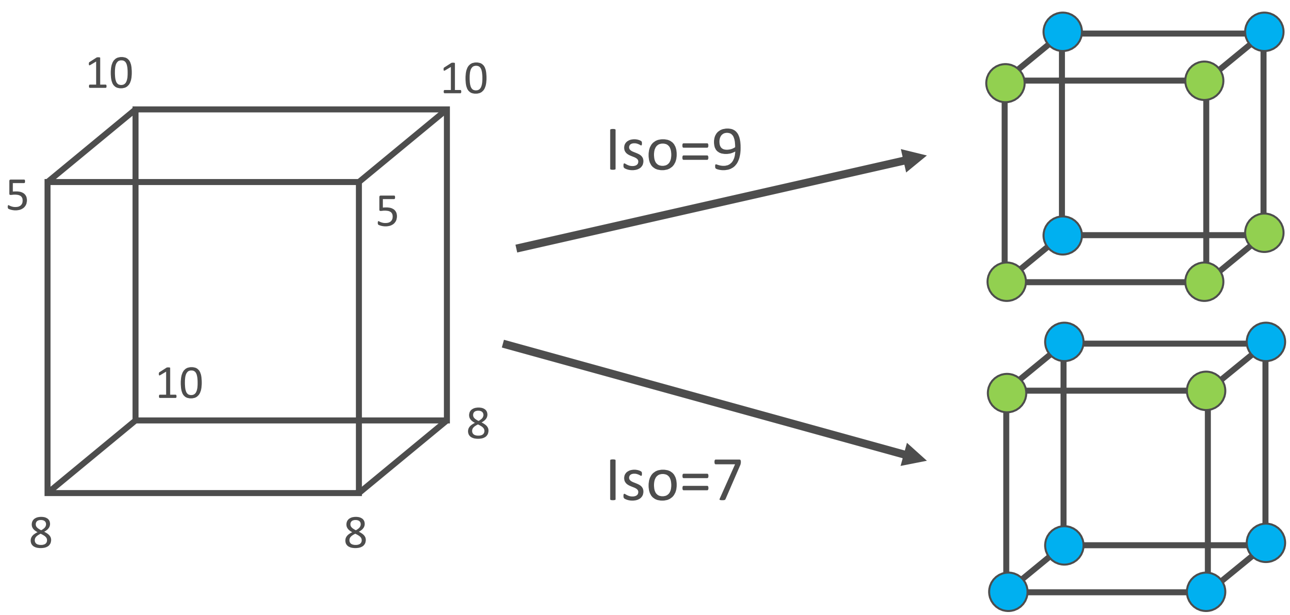
- Step 1: Consider a cell defined by eight vertices with associated data values



Marching Cubes

SIEMENS
Ingenuity for life

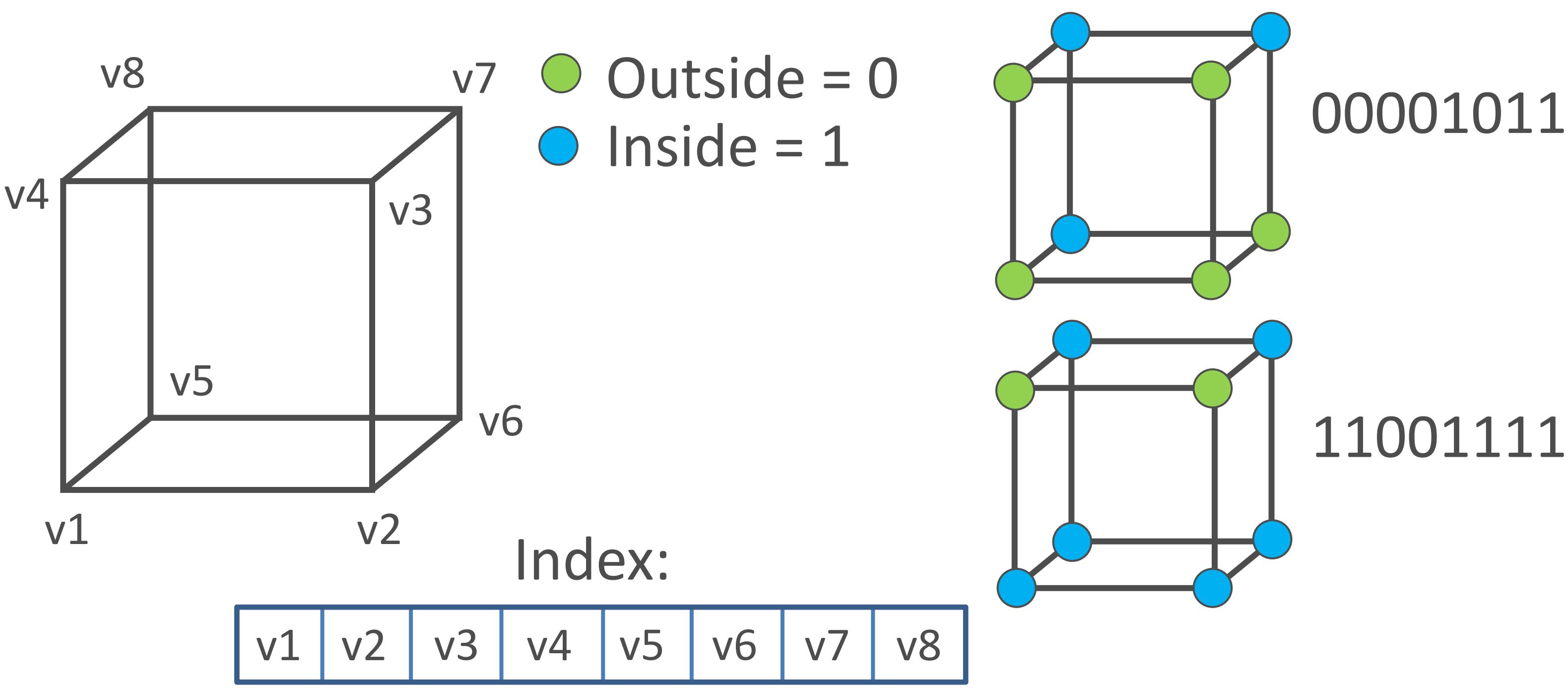
- Step 2: Classify each cell according to whether it lies
  – outside the surface (value < iso-value)
  – inside the surface (value ≥ iso-value)



Marching Cubes

SIEMENS
Ingenuity for life

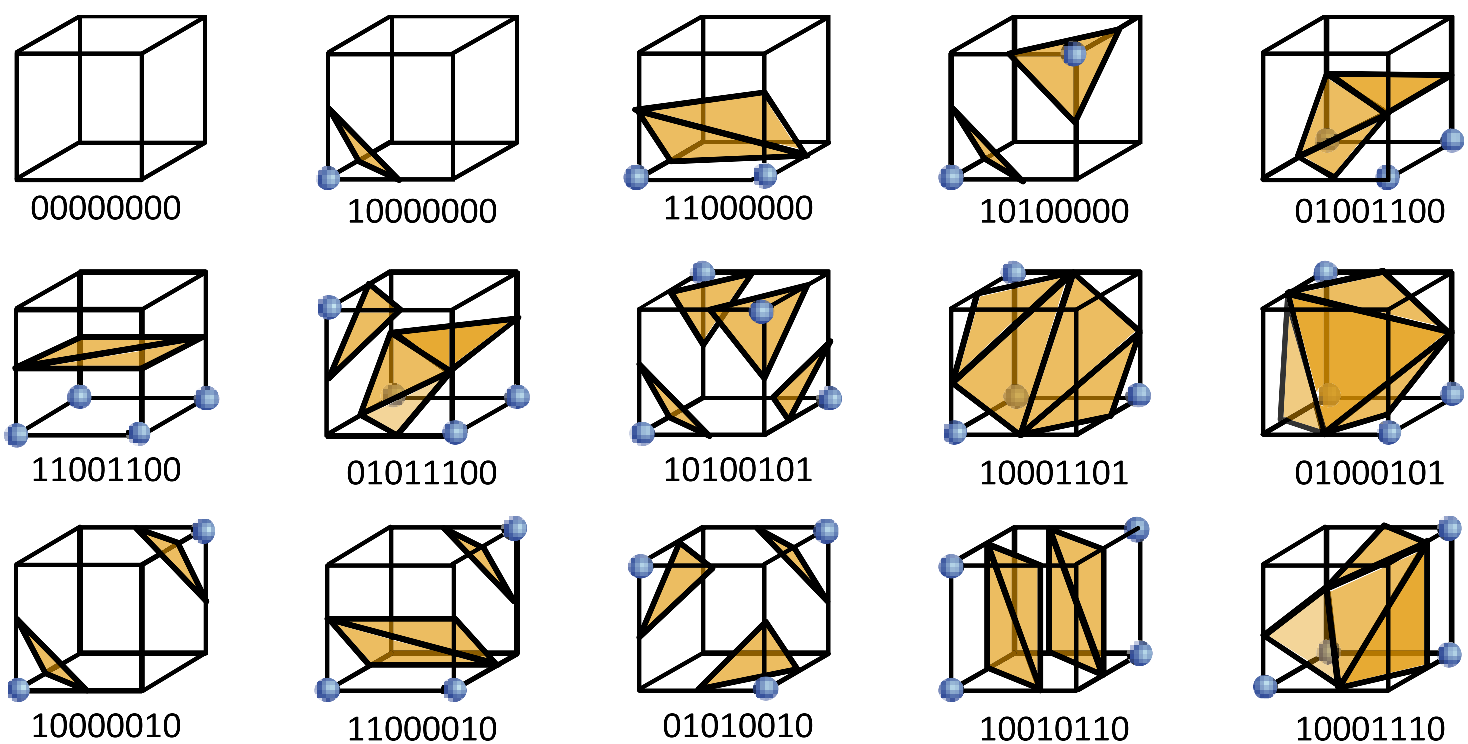
- Step 3: Use the binary labeling of each cell to compute an index



Marching Cubes

SIEMENS
Ingenuity for life

- Step 4: Per index: look up the triangulation for this index from a pre-computed table
  - All 256 cases can be derived from 15 base cases due to symmetries

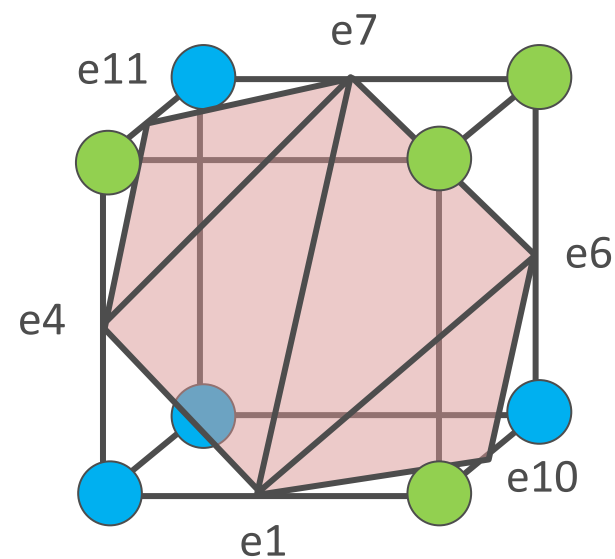


Marching Cubes

SIEMENS
Ingenuity for life

- Step 4 cont.: Get triangulation from table
  - Example for index = 10001101
  - Table at entry 10001101 stores:

    #triangles = 4
    triangle 1 = e4, e7, e11
    triangle 2 = e1, e7, e4
    triangle 3 = e1, e6, e7
    triangle 4 = e1, e10, e6

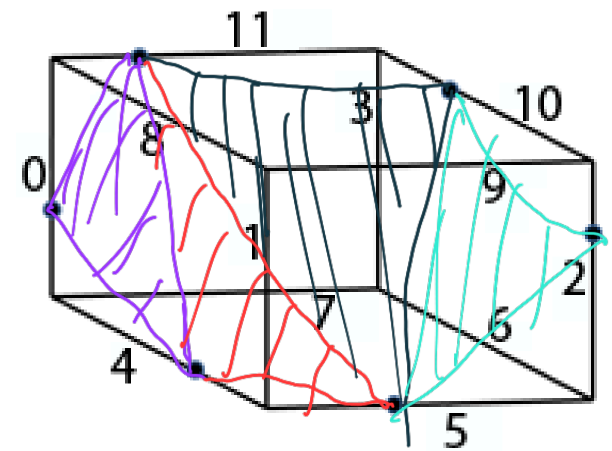


Marching Cubes

SIEMENS
Ingenuity for life

- Exercise

Cubic cell with intersection points with iso-surface. Intersection points are marked by dots. Edges are numbered as shown. Write down all the information that would be stored in the pre-computed Marching Cubes table.



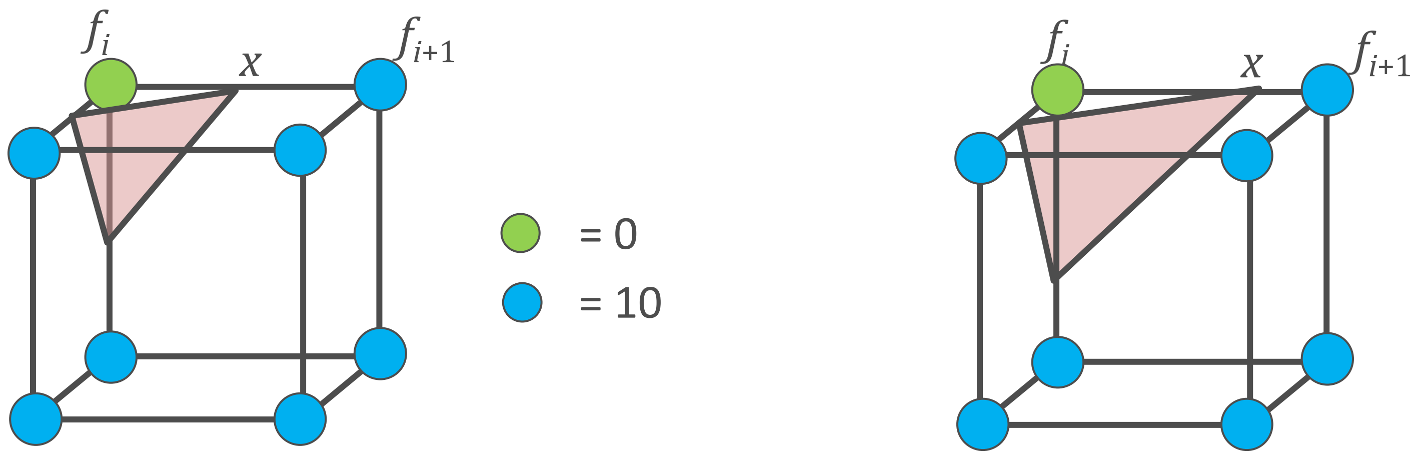
(Note: multiple solutions possible, based on triangulation)

Marching Cubes

SIEMENS
Ingenuity for life

- Step 5: For each triangle edge, find the vertex location along the edge using linear interpolation of the vertex values

kenar lerdan biri bir köşeye geliyor



Iso=5

Iso=8

$$x = i + \left(\frac{Iso - f_i}{f_{i+1} - f_i} \right)$$

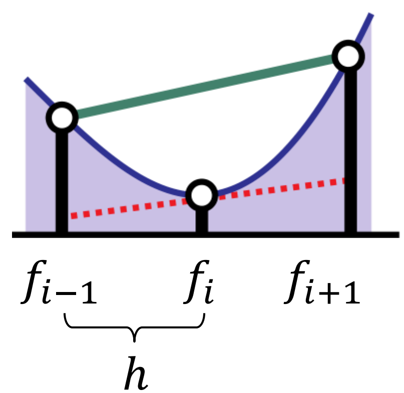
Marching Cubes

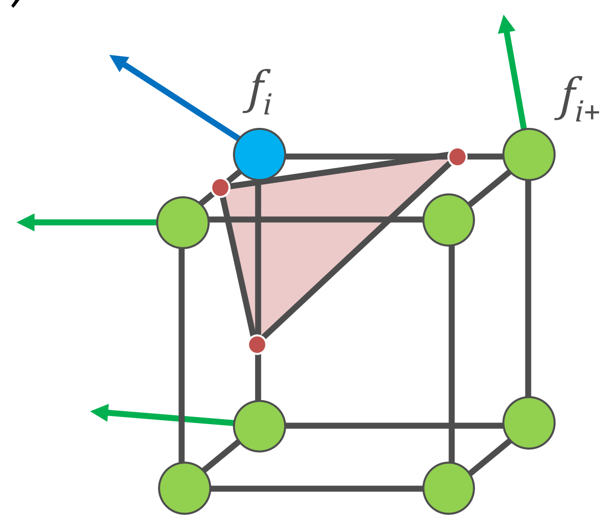
- Step 6: Calculate normals at each cube vertex (via finite differences), and interpolate along the edges
  - Normal vector at vertex (i, j, k):

$$\nabla\rho(\boldsymbol{x}) \approx \begin{pmatrix} (f_{i+1,j,k} - f_{i-1,j,k})/2h \\ (f_{i,j+1,k} - f_{i,j-1,k})/2h \\ (f_{i,j,k+1} - f_{i,j,k-1})/2h \end{pmatrix}$$

inside

outside





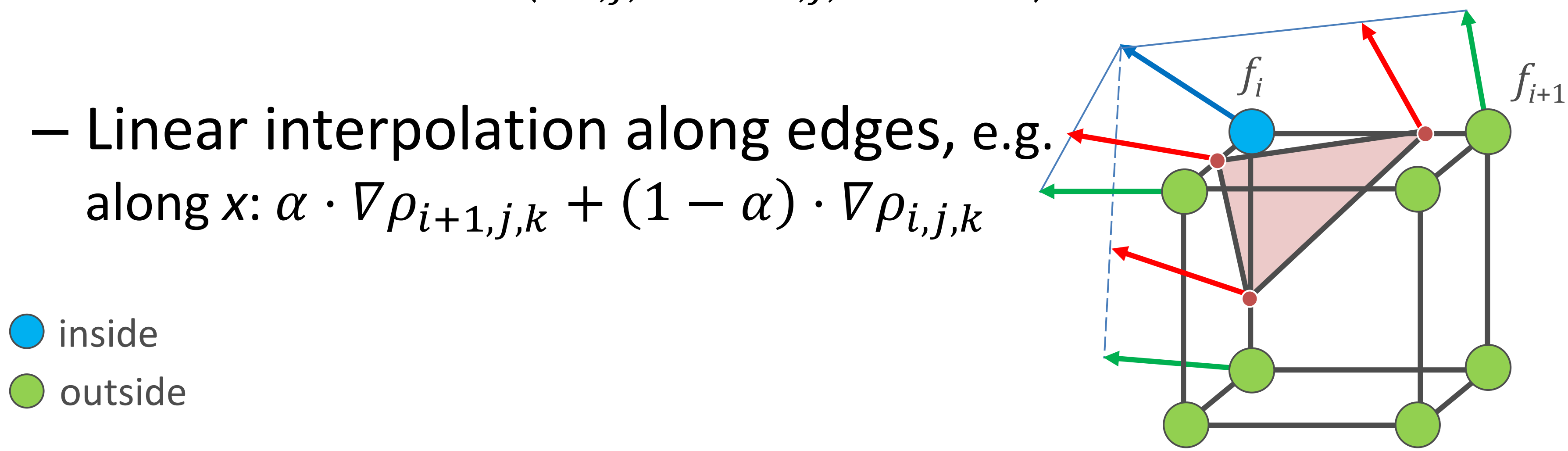
Marching Cubes

SIEMENS
Ingenuity for life

- Step 6: Calculate normals at each cube vertex (via finite differences), and interpolate along the edges
  - Normal vector at vertex (i, j, k):

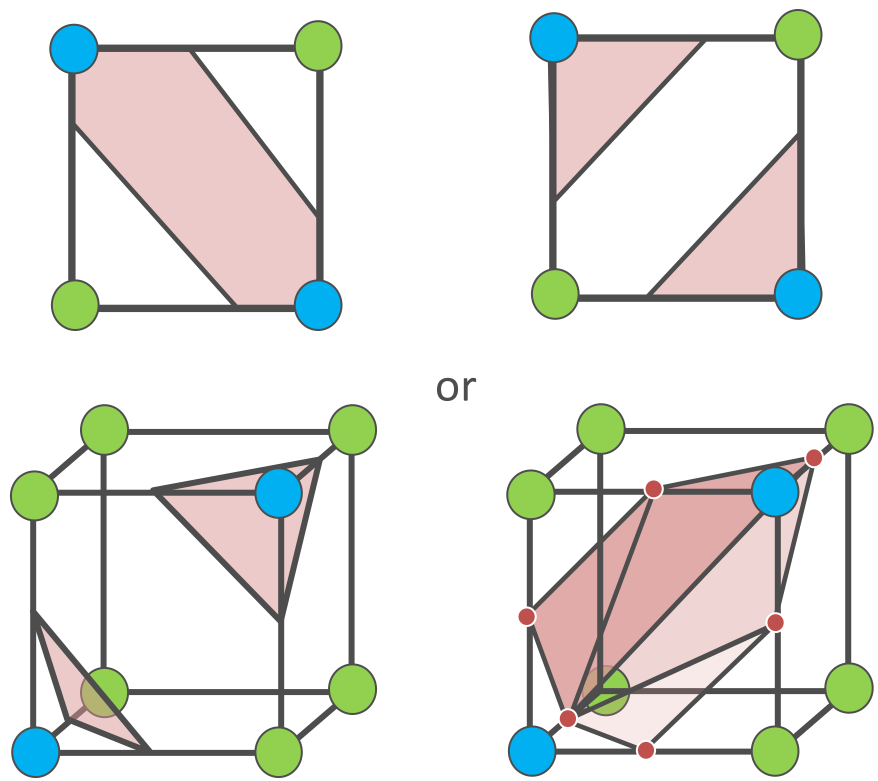
$$\nabla\rho(\boldsymbol{x}) \approx \begin{pmatrix} (f_{i+1,j,k} - f_{i-1,j,k})/2h \\ (f_{i,j+1,k} - f_{i,j-1,k})/2h \\ (f_{i,j,k+1} - f_{i,j,k-1})/2h \end{pmatrix}$$

  - Linear interpolation along edges, e.g. along x: $\alpha \cdot \nabla\rho_{i+1,j,k} + (1-\alpha) \cdot \nabla\rho_{i,j,k}$



Marching Cubes

- Step 7: Consider ambiguous cases
  - Ambiguous cases: 3, 6, 7, 10, 12, 13
  - Use decider as in 2D



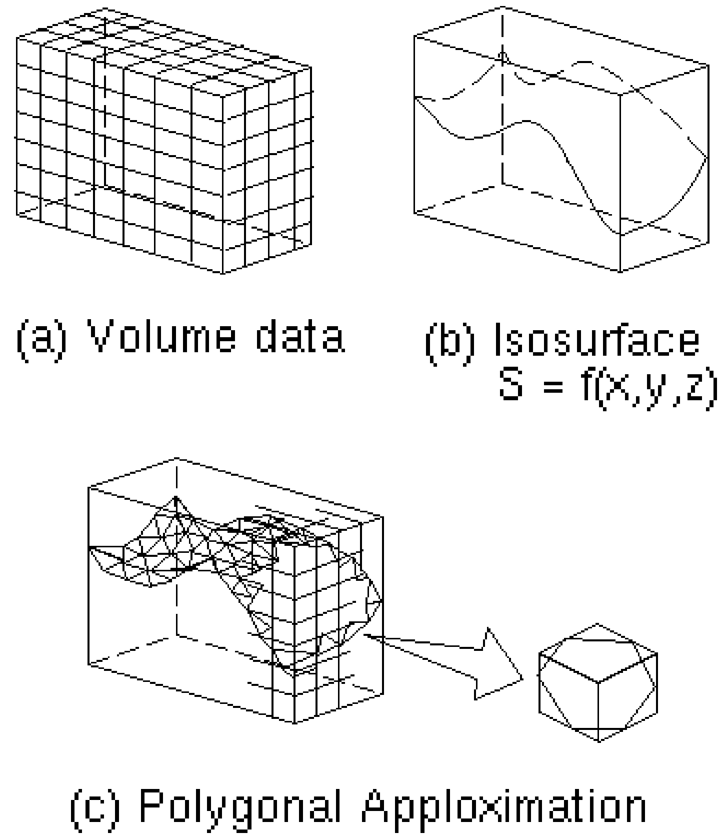
Case 3　　Case 3c

inside

outside

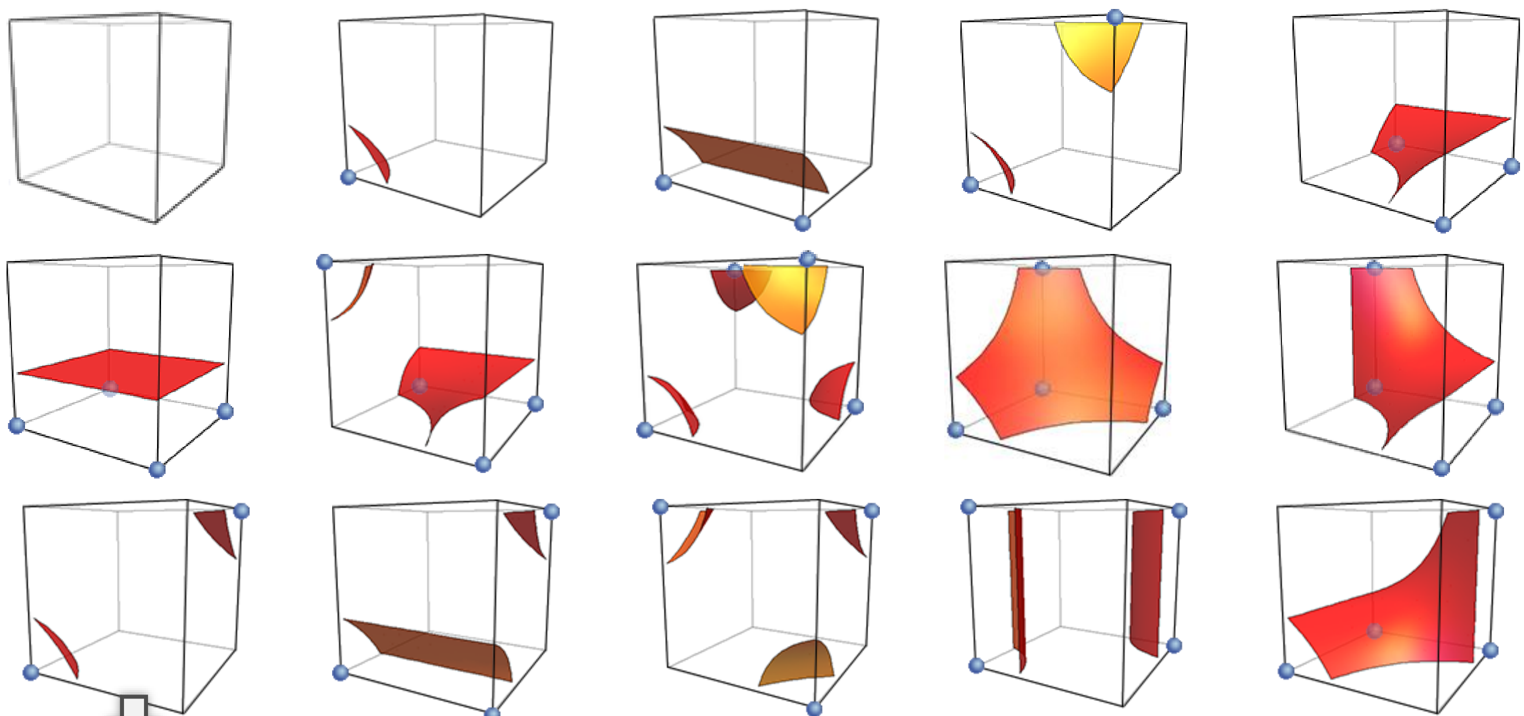
Marching Cubes

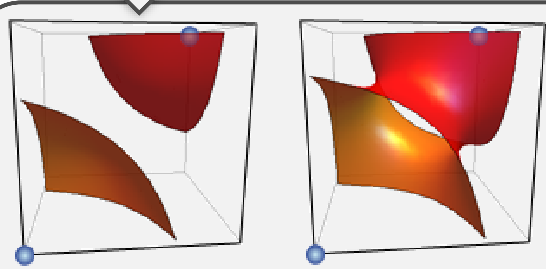
- Summary
  - 256 Cases
    - Reduce to 15 cases by symmetry
  - Ambiguity resides in cases 3, 6, 7, 10, 12, 13
  - Causes holes if arbitrary choices are made
- Up to 5 triangles per cube
- Large datasets can result in several millions of triangles



Marching Cubes

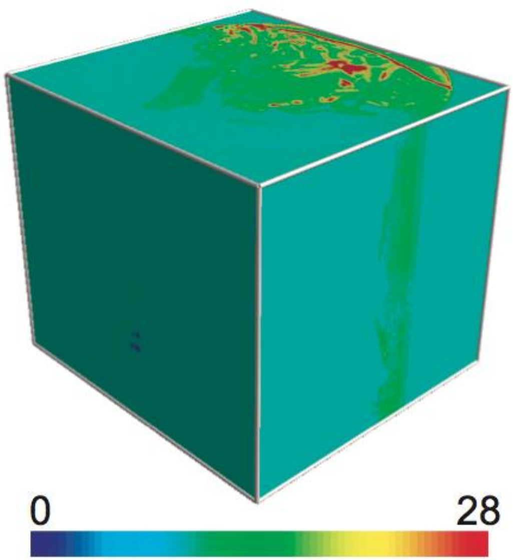
- Summary: Note that triangulation is only approximation of true isosurfaces produced by trilinear interpolation





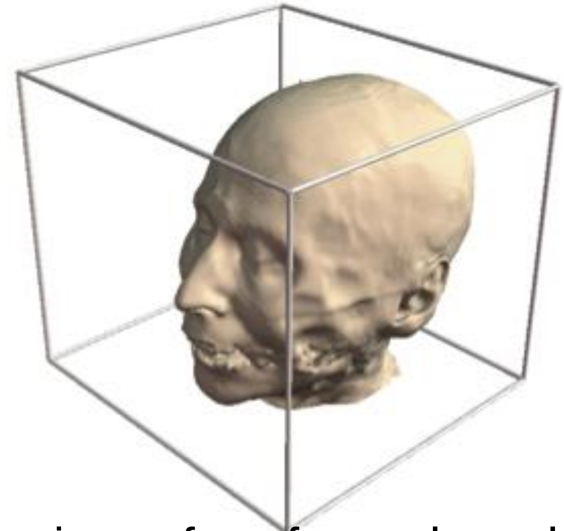
Dependent on iso-value
“tunnels” can be generated

Marching Cubes

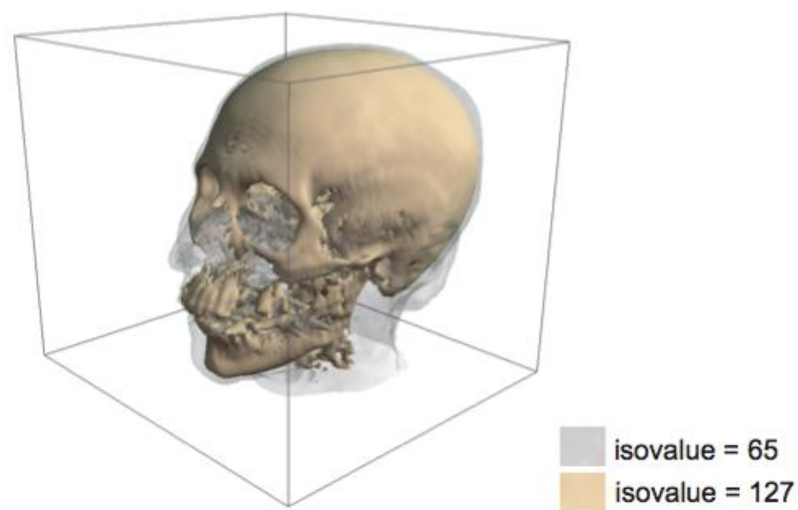


scalar CT volume
(tissue density)

isosurfaces



isosurface for scalar value
corresponding to skin

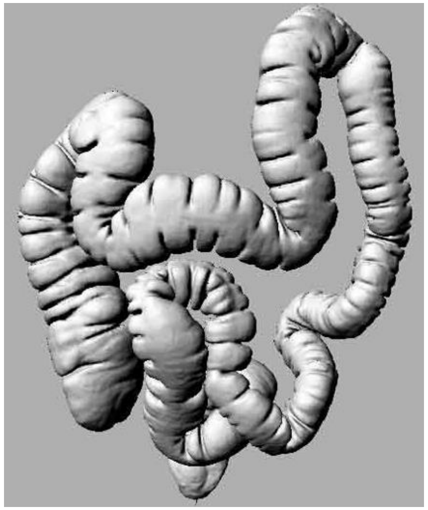


isosurfaces for skin and bone

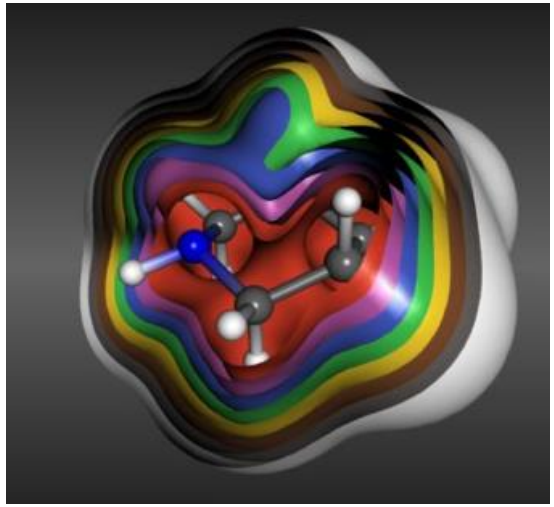
- extremely simple to use tool
- insightful results

[A. Telea]

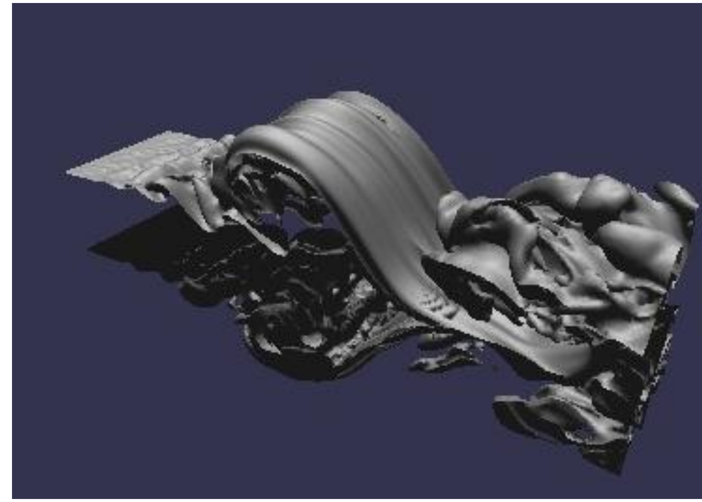
Marching Cubes



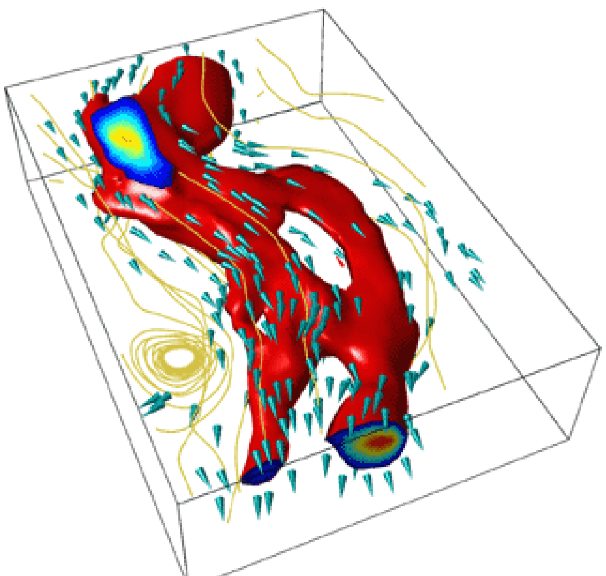
Colon (CT dataset)



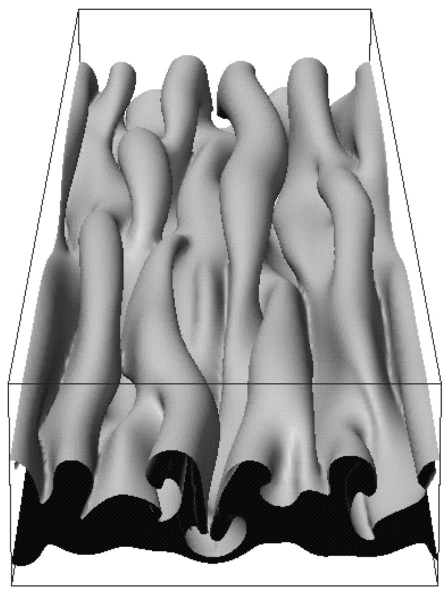
Electron density in molecule



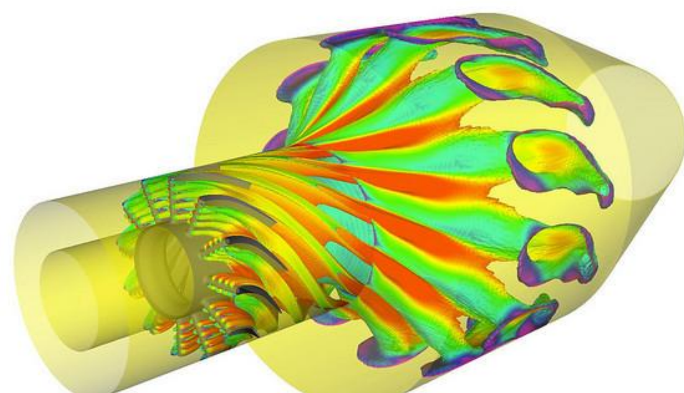
Velocity in 3D fluid flow



Velocity in 3D fluid flow



Magnetic field in sunspots

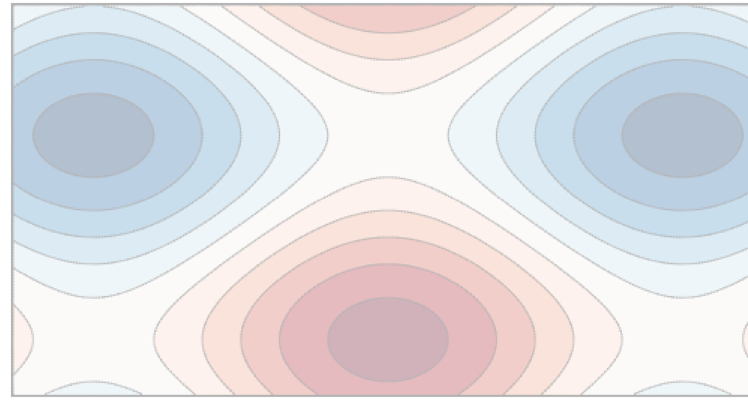


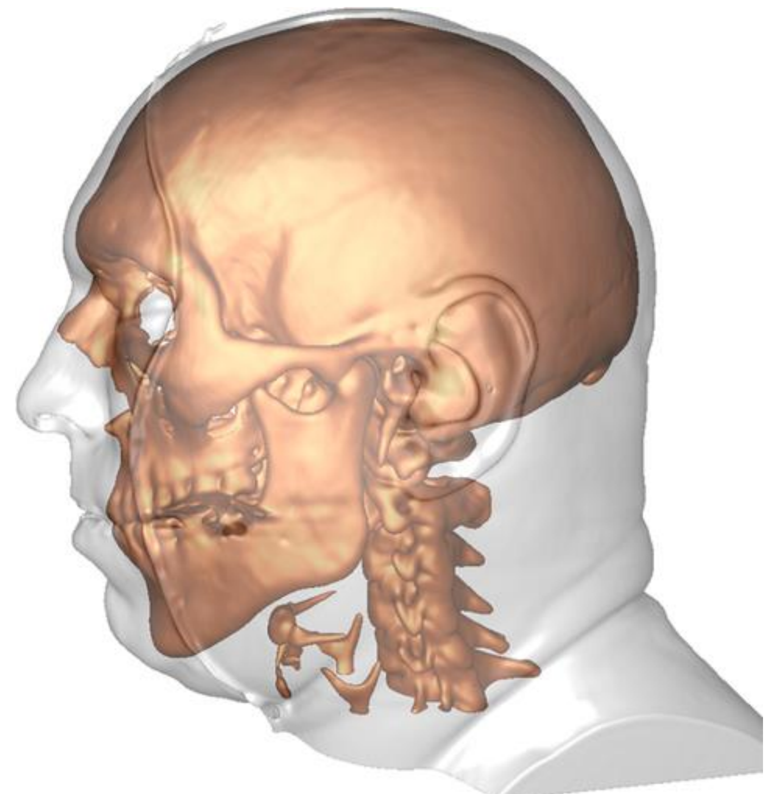
Fuel concentration, colored by temperature in jet engine

[A. Telea]

Overview

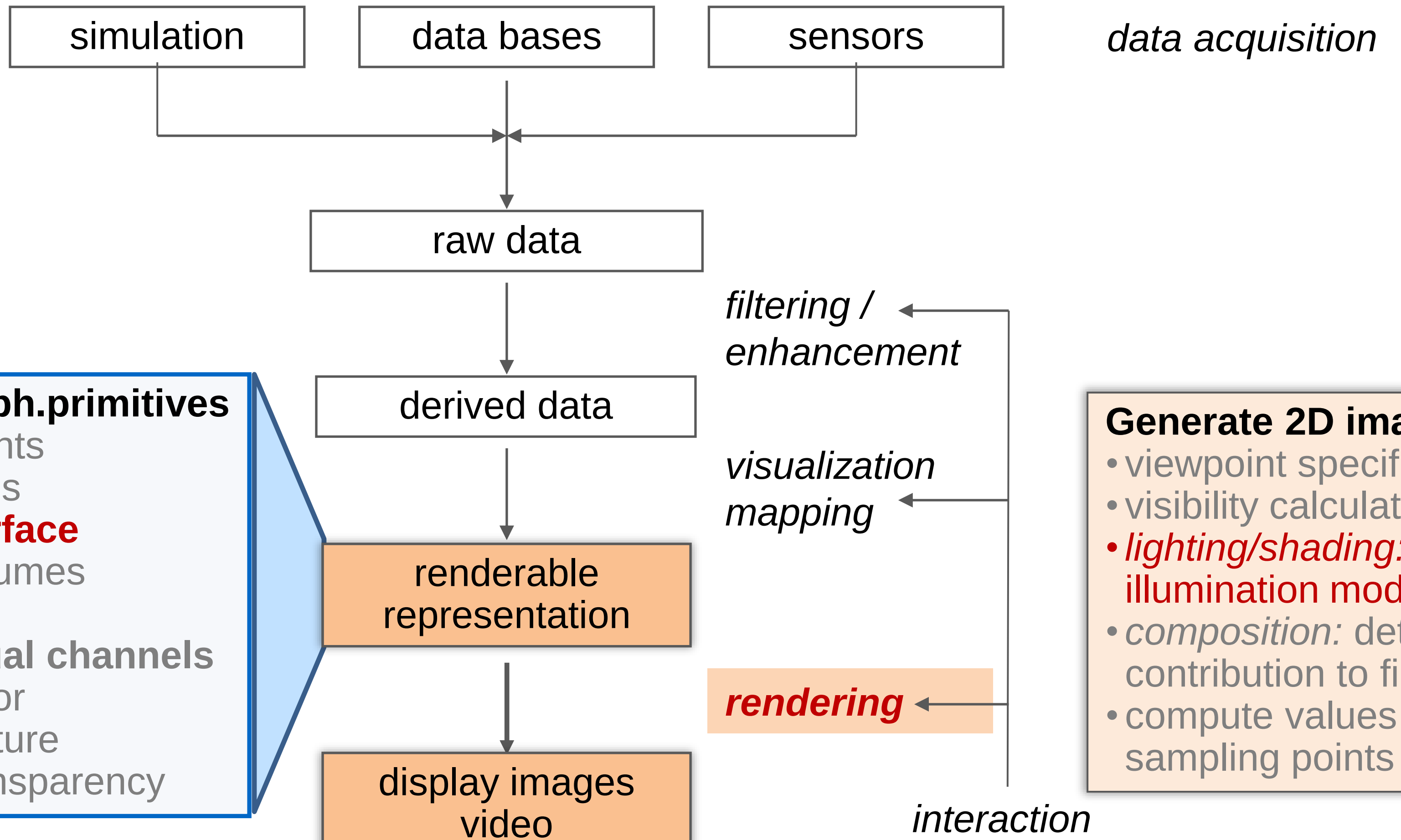
- Isolines & Isosurfaces
  - Marching squares
  - Marching cubes
- Lighting
  - Phong illumination model
- Gradient approximation





Lighting

SIEMENS
Ingenuity for life



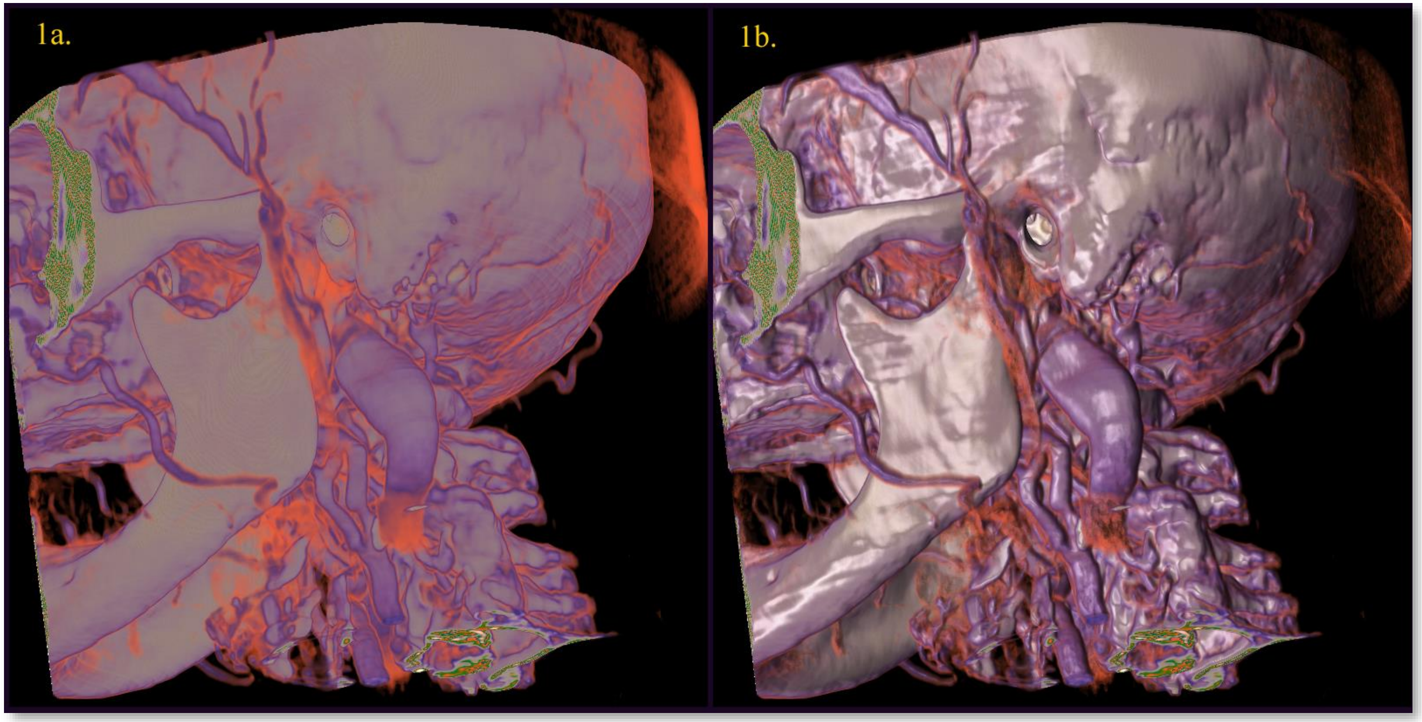
Graph.primitives
- points
- lines
- **surface**
- volumes

Visual channels
- color
- texture
- transparency

Generate 2D image/video
- viewpoint specification
- visibility calculation
- *lighting/shading:* evaluate illumination model
- *composition:* determine contribution to final image
- compute values at sampling points

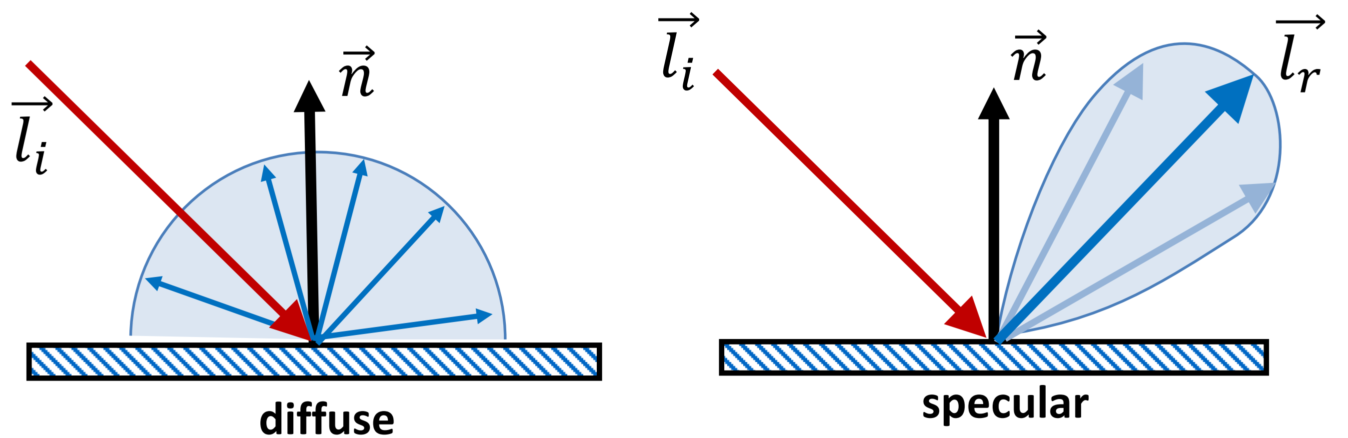
Lighting

- Necessary to emphasize iso-surface shape
  - Simulate reflection of light and effect on color

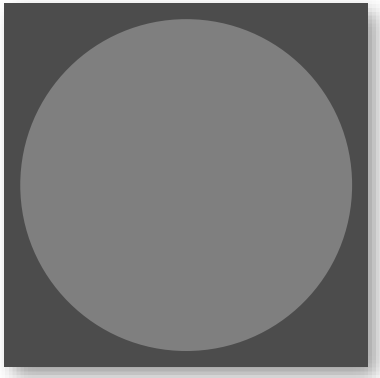


Lighting

- Phong's illumination model
  - Considers ambient light and point lights as well as the material color and reflection properties
  - Ambient light: background light, constant everywhere
  - Diffuse reflector reflects equally into all directions
  - Specular reflector reflects mostly into the mirror direction



Lighting



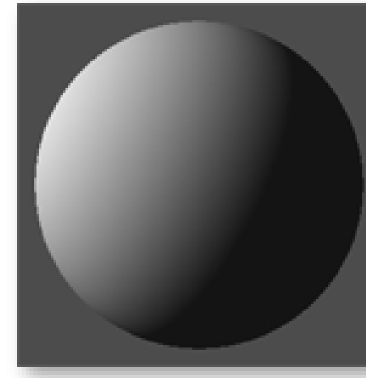
- Ambient light: $C = k_a C_a O_d$
  - Background light, constant everywhere
  - k_a … ambient reflection coefficient $\in [0, 1]$
  - C_a … color of the ambient light
  - O_d … object color

We know that light doesn't come directly always.

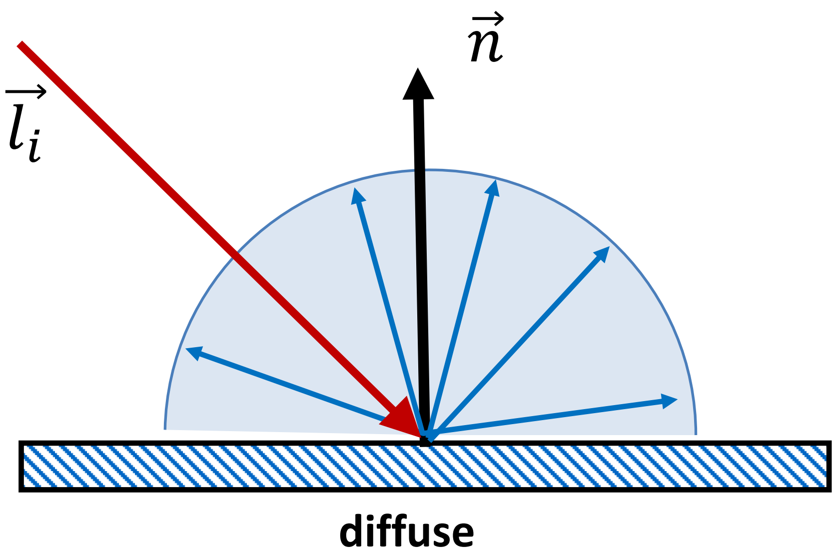
Generally, Light reflect from objects and comes our object. Ambient Light is the basic color of object.

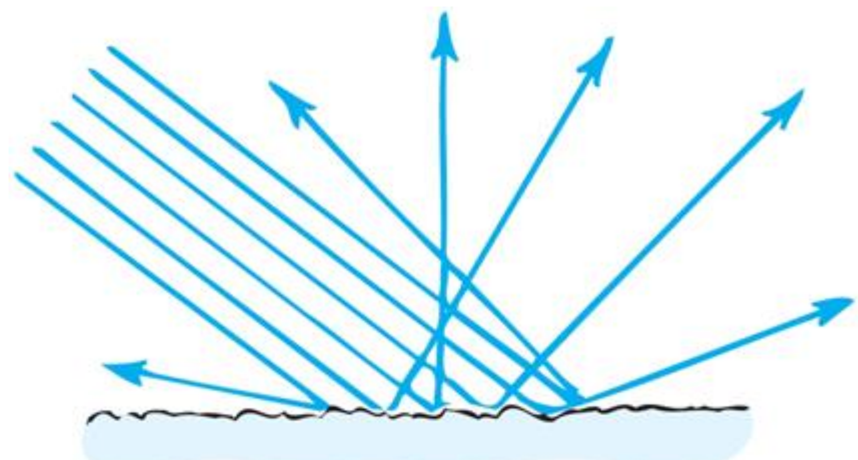
it is constant.

Lighting



- Diffuse reflection (Lambert term)
  - Connected to physical reality (Lambert's law)
  - Approximates rough surfaces
  - Scatters light equally in all directions



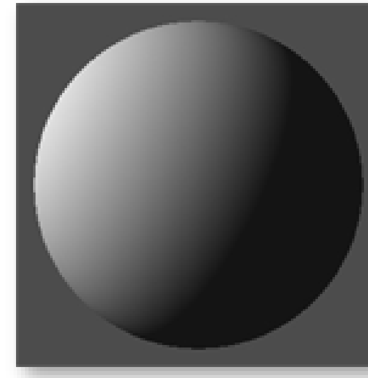


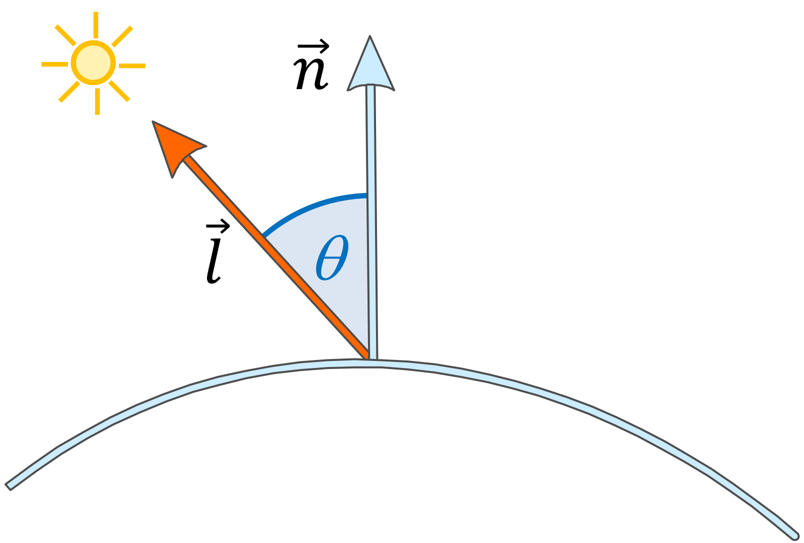
http://multivis.net/lecture/phong.html

Lighting

- Diffuse reflection: $C = k_d C_p O_d \cos\theta$
  - k_d … diffuse reflection coefficient $\in [0, 1]$
  - C_p … color of the point light
  - O_d … object color
  - $\cos\theta$ … angle between light vector $\vec{l}$ and normal $\vec{n}$

yüzeyin yansıması

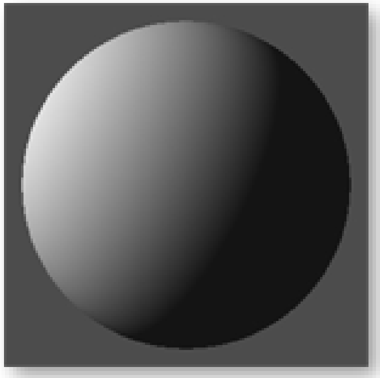




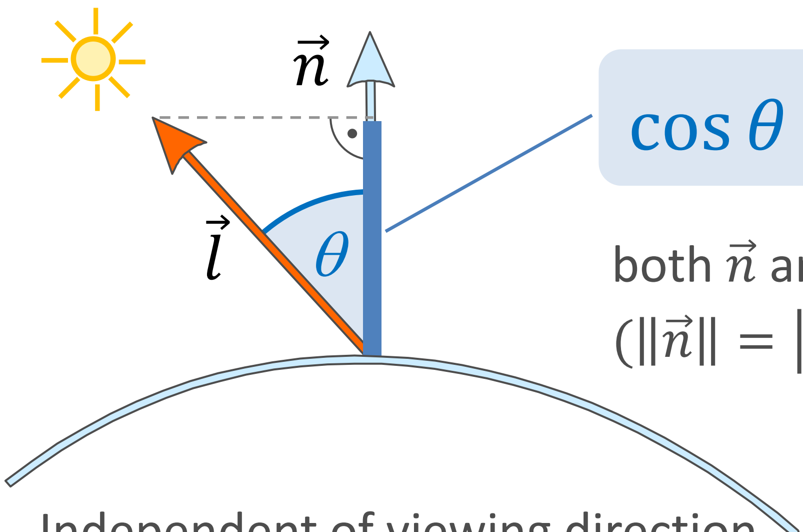
http://multivis.net/lecture/phong.html

Lighting

SIEMENS
Ingenuity for life



- Diffuse reflection: $C = k_d C_p O_d \cos\theta$
  - k_d … diffuse reflection coefficient $\in [0, 1]$
  - C_p … color of the point light
  - O_d … object color
  - $\cos\theta$ … angle between light vector $\vec{l}$ and normal $\vec{n}$



$$\cos\theta = \vec{n} \cdot \vec{l}$$

→ dot product.

both $\vec{n}$ and $\vec{l}$ have unit length
$(\|\vec{n}\| = \|\vec{l}\| = 1)$

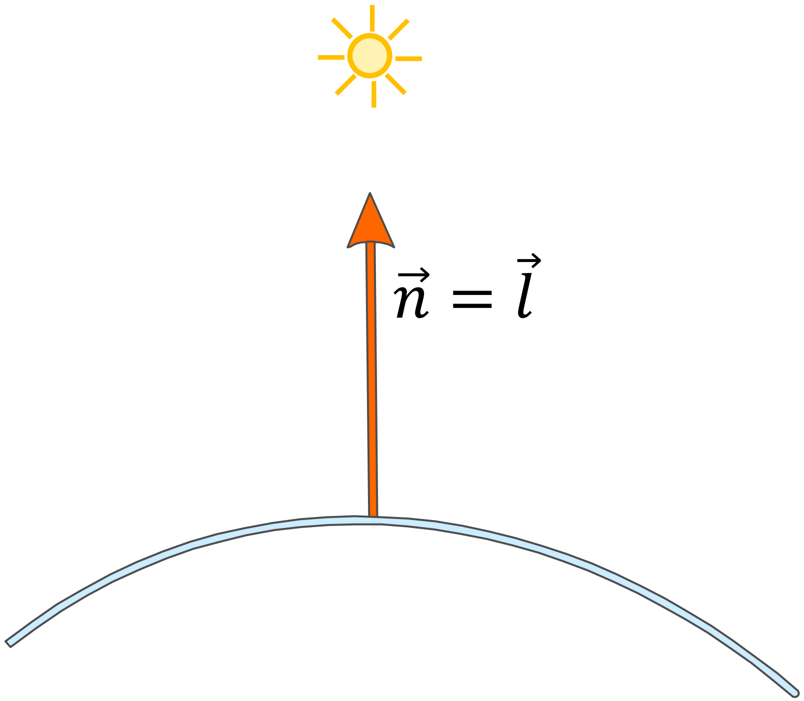
Bir nesneye yukarıdan ışık kaynağını tutarsak en parlak görüntüyü alırız. Bunun nene gelen ışıkile normunun çarpımlarının ⊥ olması-

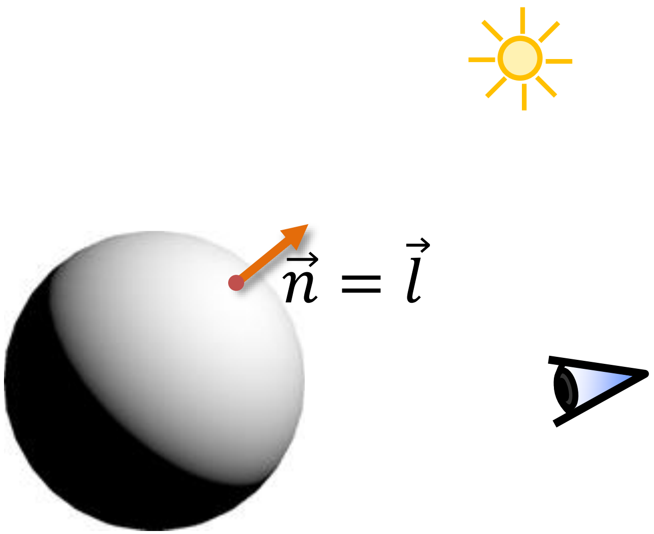
Independent of viewing direction

http://multivis.net/lecture/phong.html

Lighting

- Diffuse reflection: $C = k_d C_p O_d \cos\theta$
  - The light is precisely above the point ($\theta = 0$)
  - $\cos 0° = 1$ → 100% intensity

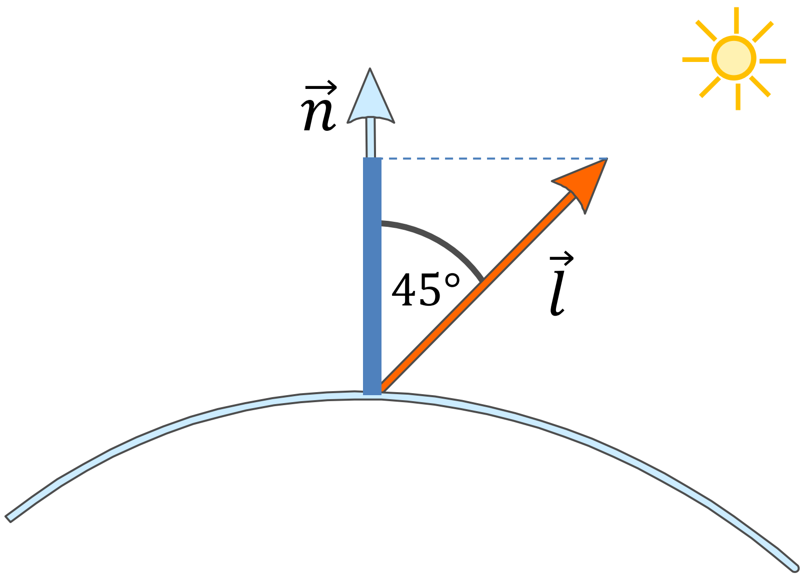


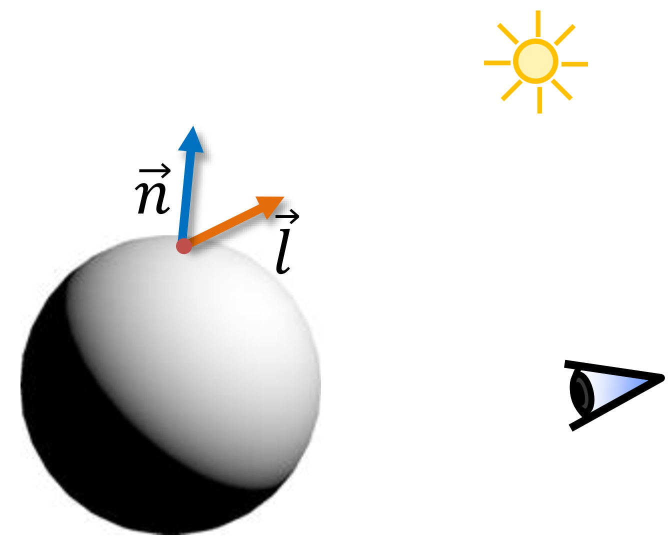


Lighting

SIEMENS
Ingenuity for life

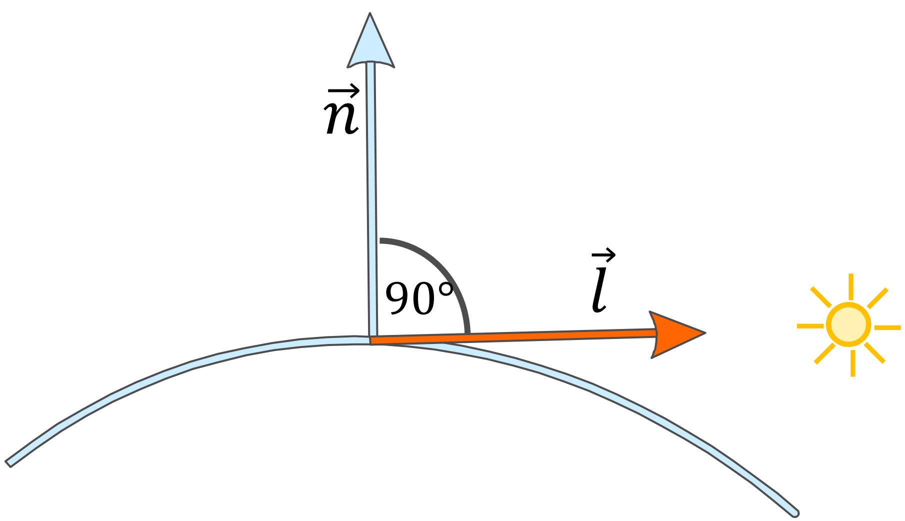
- Diffuse reflection: $C = k_d C_p O_d \cos\theta$
  - The light shines onto a point in a 45° angle
  - $\cos 45° \approx 0.7$ → 70% intensity

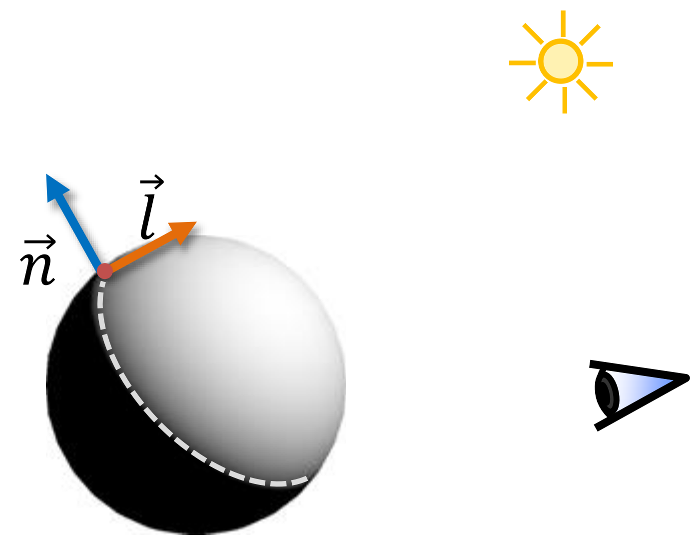




Lighting

- Diffuse reflection: $C = k_d C_p O_d \cos\theta$
  - The light shines onto a point in a 90° angle
  - $\cos 90° = 0$ → 0% intensity

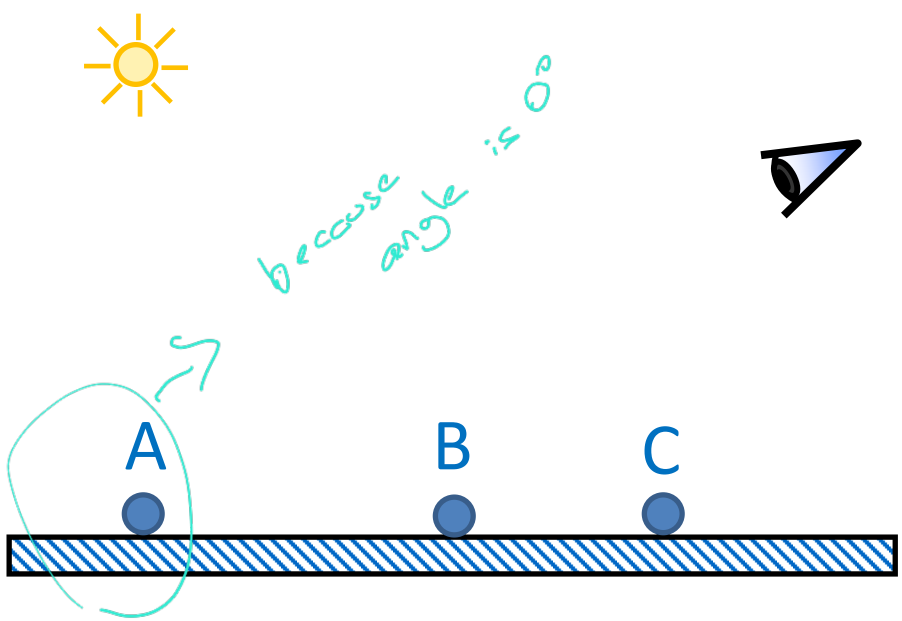




The flatter light falls on a surface,
the darker it will appear

Lighting

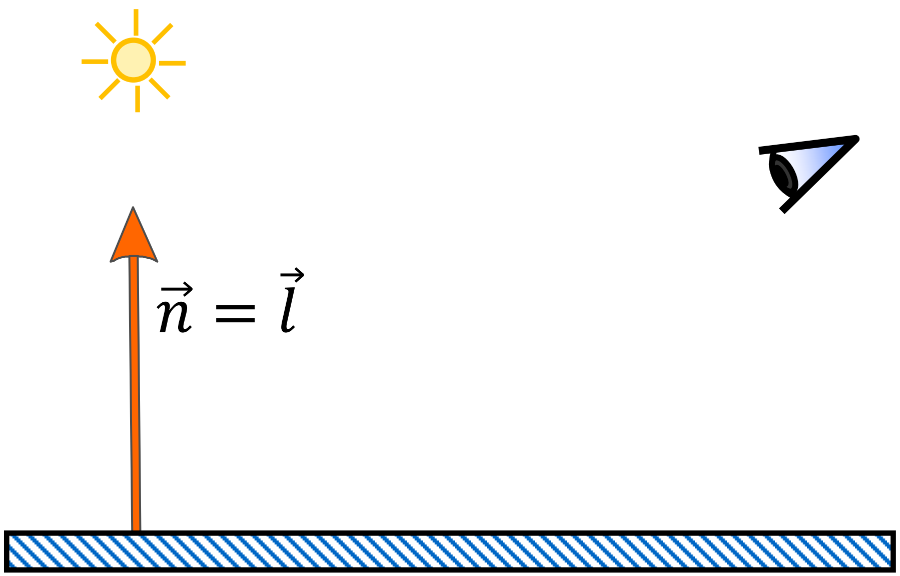
- Plane with only diffuse reflection
  - At which point does the viewer see highest diffuse reflection?

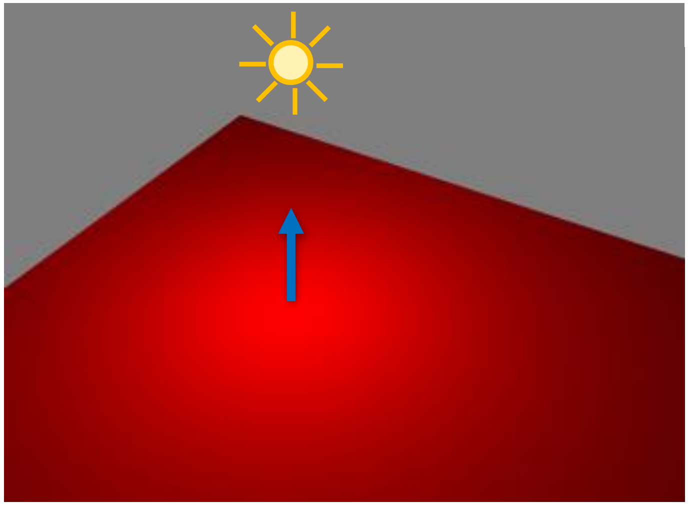


http://multivis.net/lecture/phong.html

Lighting

- Plane with only diffuse reflection
  - At which point does the viewer see highest diffuse reflection?
  - Max. intensity → $\cos\theta = 1$
  - Normal vector = light vector
  - Point precisely below light source



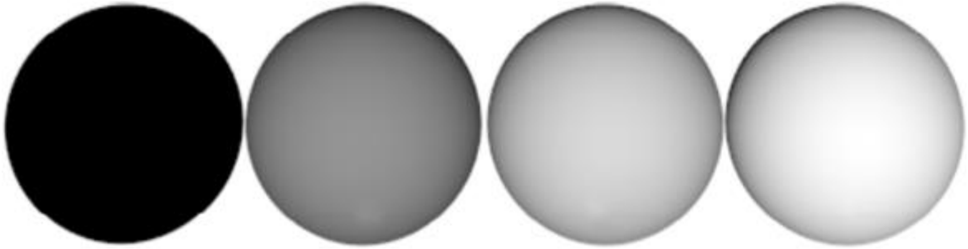


http://multivis.net/lecture/phong.html

Lighting

SIEMENS
Ingenuity for life

- Diffuse reflection: $C = k_d C_p O_d \cos\theta$
  - Varying k_d (diffuse reflection coefficient)

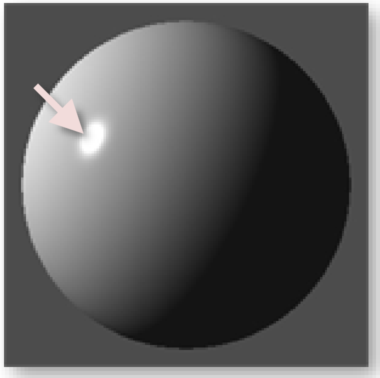


$k_d = 0$ $\quad k_d = 0.3$ $\quad k_d = 0.7$ $\quad k_d = 1$

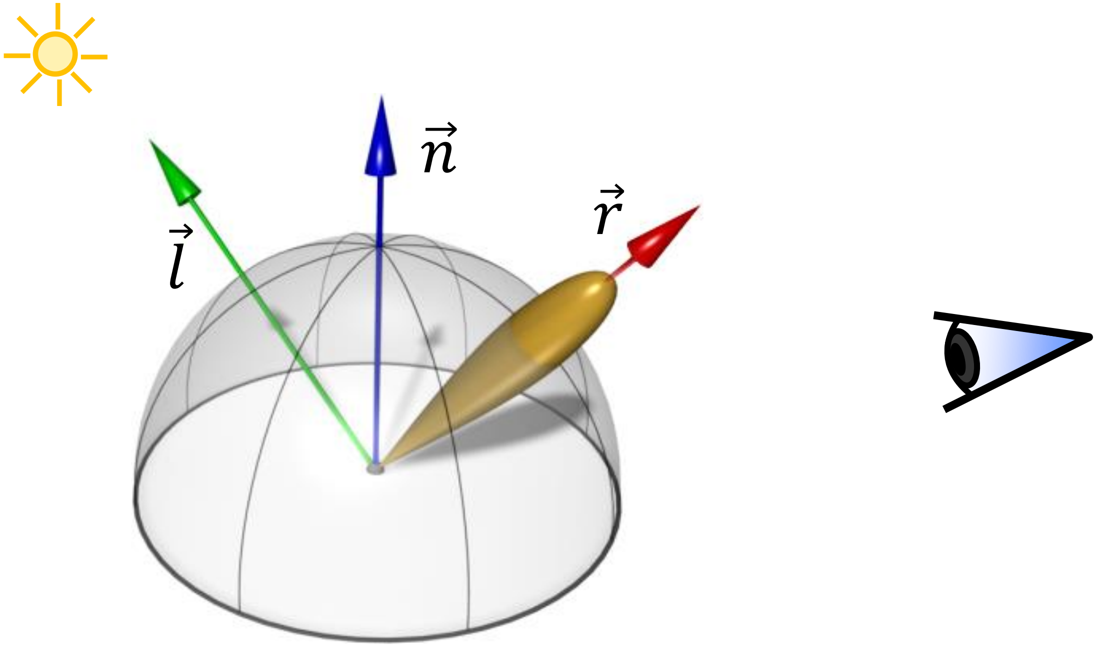
http://multivis.net/lecture/phong.html

Lighting

SIEMENS
Ingenuity for life

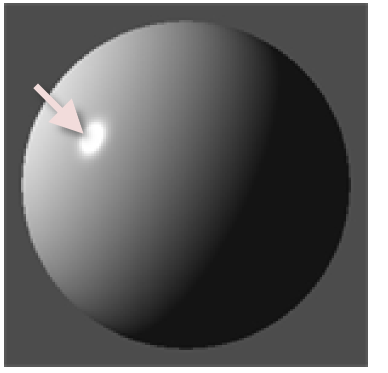


- Specular reflection
  - Highlight = reflection of light source
  - Glossy surfaces
  - Reflects mostly into the mirror direction
  - View dependent!

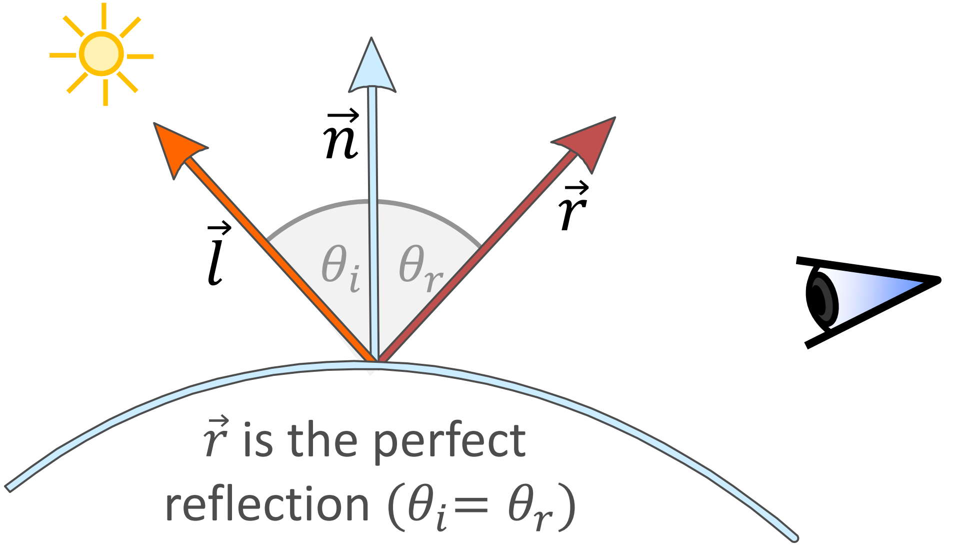


Lighting

SIEMENS
Ingenuity for life



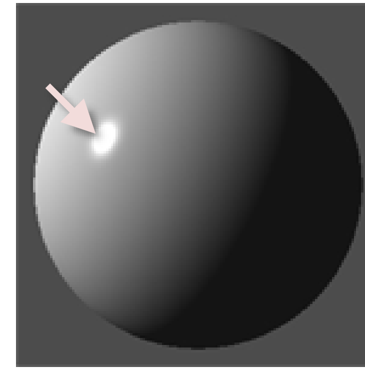
- Specular reflection: $C = k_s C_p O_d \cos^n \varphi$
  - k_s … specular reflection coefficient $\in [0, 1]$
  - $\cos \varphi$ … angle between the reflected light ray $\vec{r}$ …

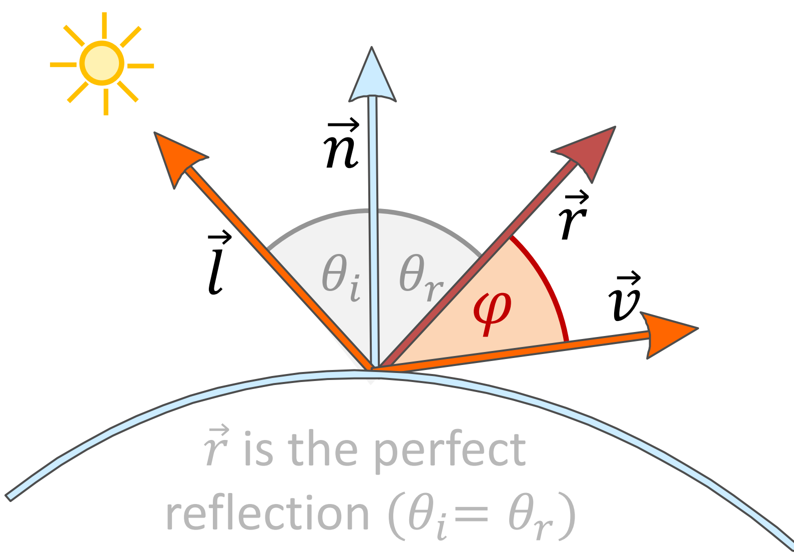


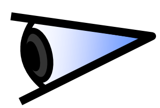
http://multivis.net/lecture/phong.html

Lighting

- Specular reflection: $C = k_s C_p O_d \cos^n \varphi$
  - k_s ... specular reflection coefficient $\in [0, 1]$
  - $\cos \varphi$ ... angle between the reflected light ray $\vec{r}$ and the vector to the viewer $\vec{v}$
  - $(\quad)^n$ ... shininess factor (controls extend of highlight)







$$\cos^n \varphi = (\vec{r} \cdot \vec{v})^n$$

$\vec{r}$ and $\vec{v}$ have unit length

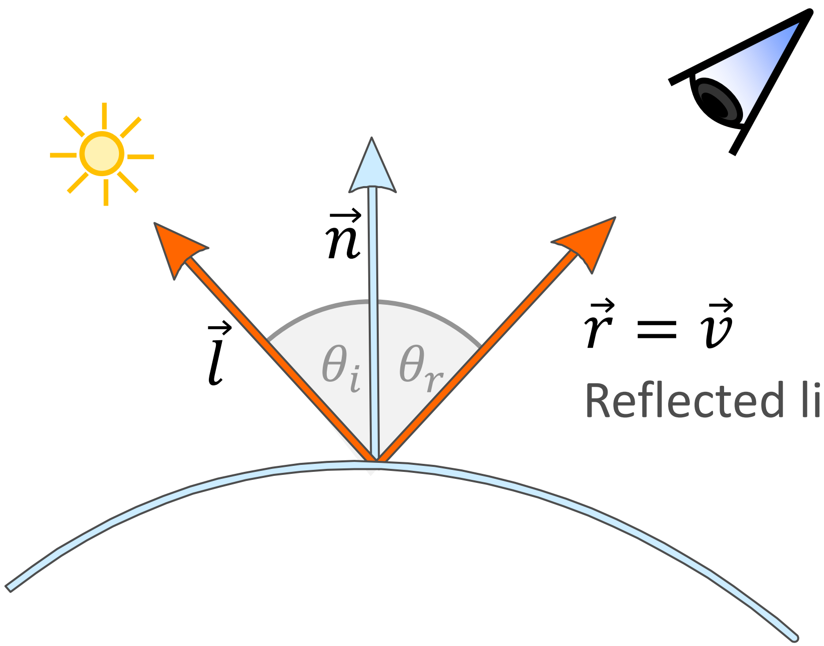
Very similar to diffusion but this time we consider receiver position. We take reflection of L and calculate φ by using n

The view vector $\vec{v}$ is in the formula, so it should look differently if we look from a different angle

Lighting

SIEMENS
Ingenuity for life

- Specular reflection: $C = k_s C_p O_d \cos^n \varphi$
  - We are looking directly into the reflection ($\varphi = 0°$)
  - $(\cos 0°)^n = 1^n$ → 100% intensity

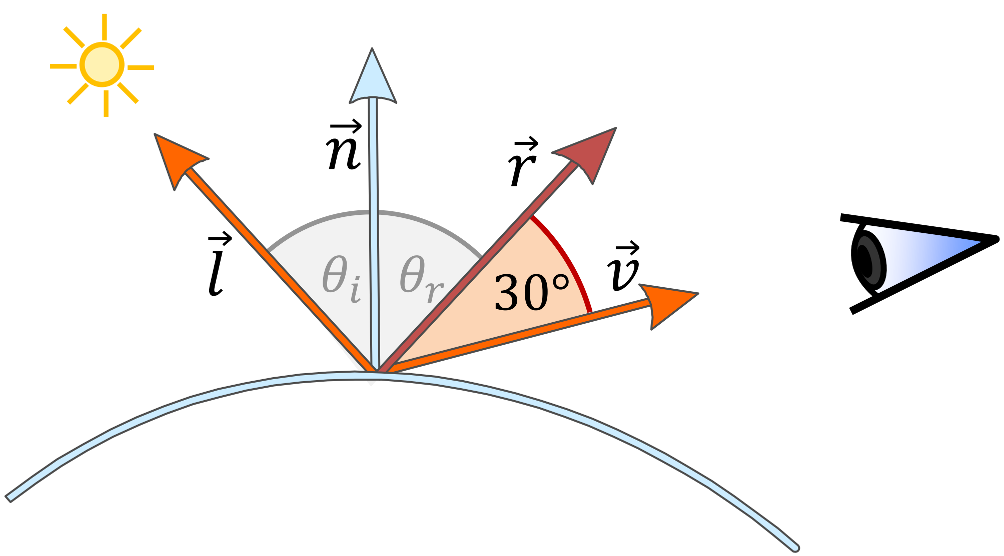


Lighting

- Specular reflection: $C = k_s C_p O_d \cos^n \varphi$

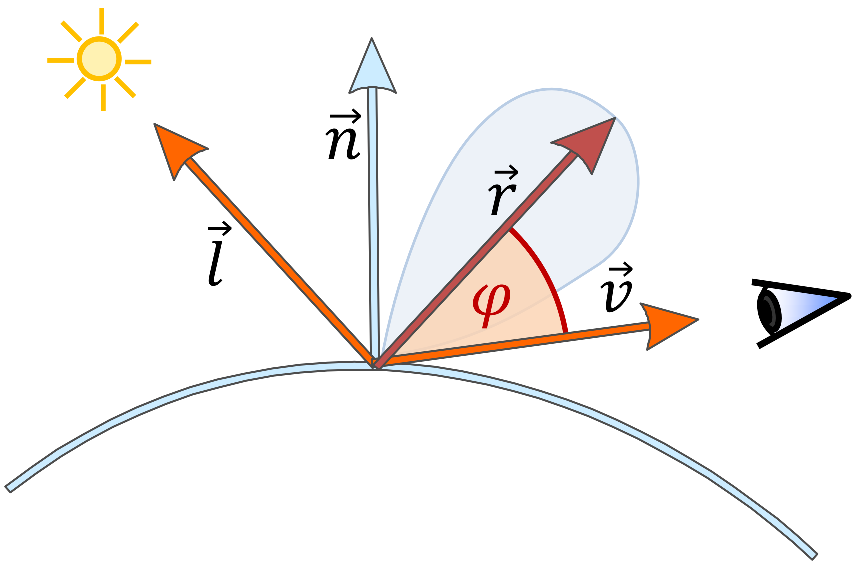
$$\varphi = 30° \rightarrow (\cos 30°)^n = \left(\frac{\sqrt{3}}{2}\right)^n \approx 0.87^n$$

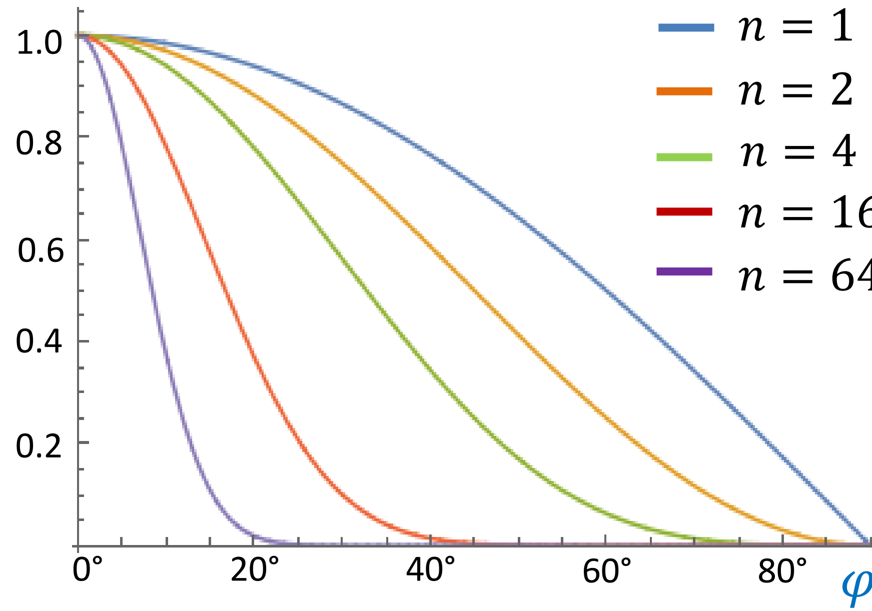
  - Case $n = 2 \rightarrow \left(\frac{\sqrt{3}}{2}\right)^2 = \frac{3}{4}$ → 75% intensity
  - Case $n = 4 \rightarrow \left(\frac{\sqrt{3}}{2}\right)^4 = \frac{9}{16} \approx 0.56$ → 56% intensity



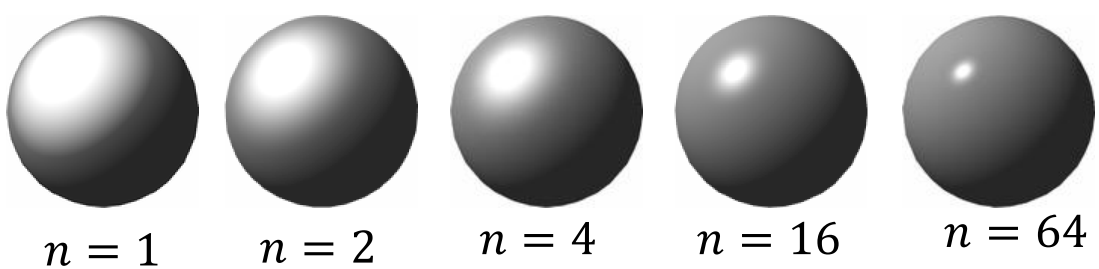
Lighting

- Effect of the exponent n of highlight





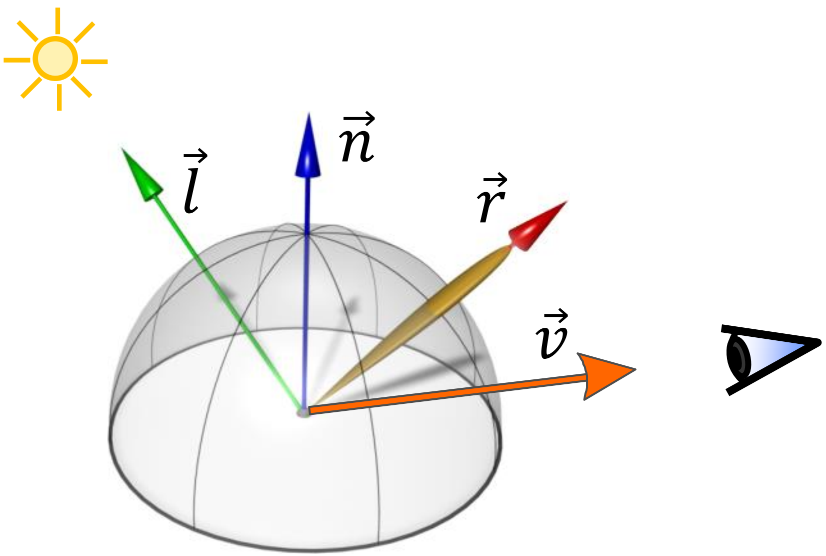
$$\cos^n \varphi = (\vec{r} \cdot \vec{v})^n$$

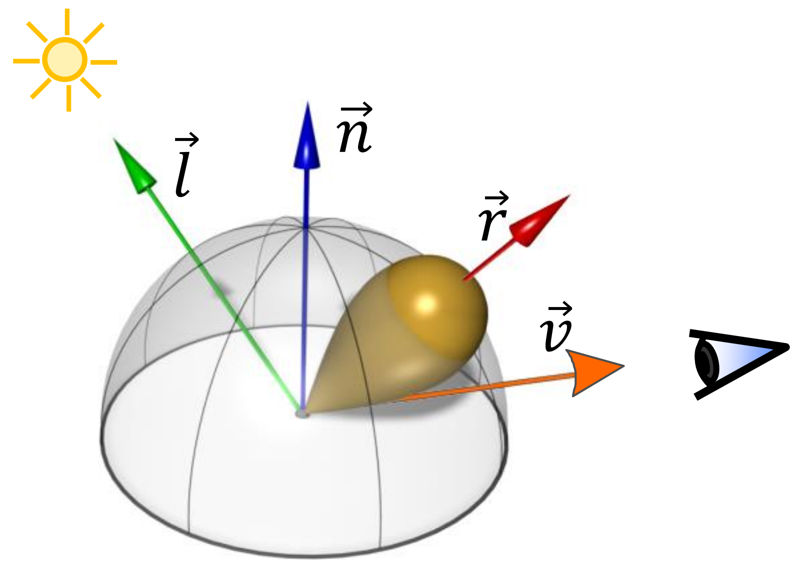


Lighting

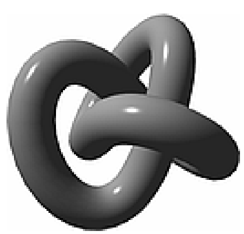
- Effect of the exponent n of highlight

$$C = k_s C_p O_d \cos^{n} \varphi$$

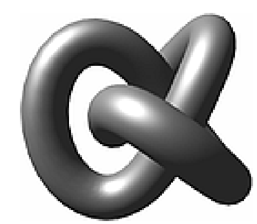




Shiny surface (large n)



Dull surface (small n)

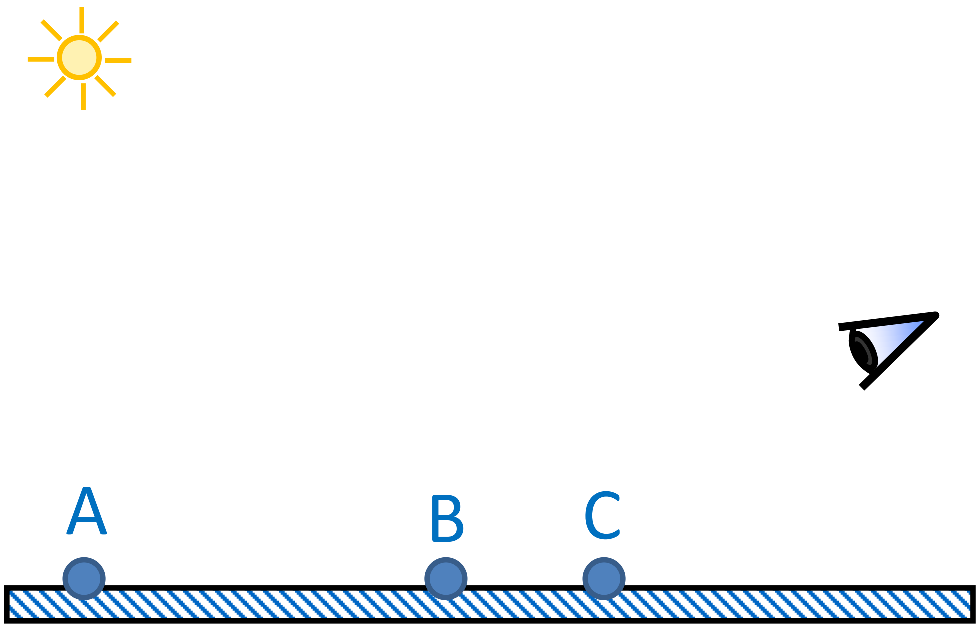


http://multivis.net/lecture/phong.html

Lighting

SIEMENS
Ingenuity for life

- Plane with diffuse and specular reflection
  - At which point is the highest specular reflection?

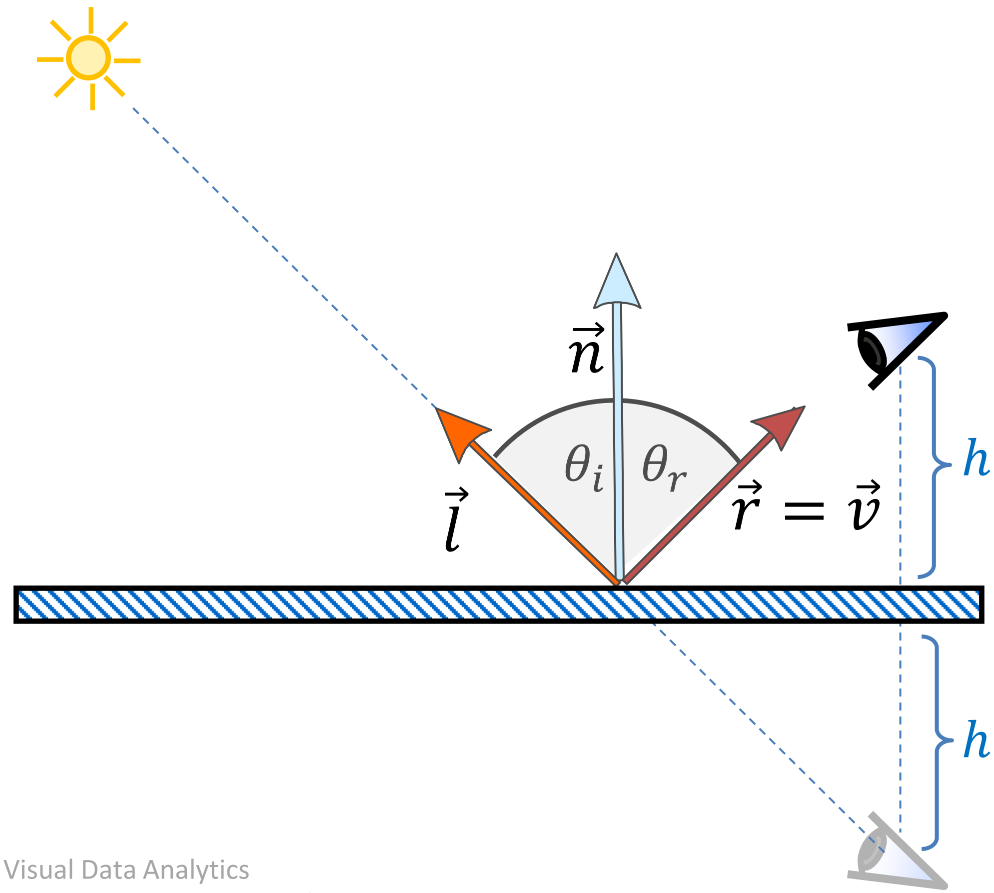


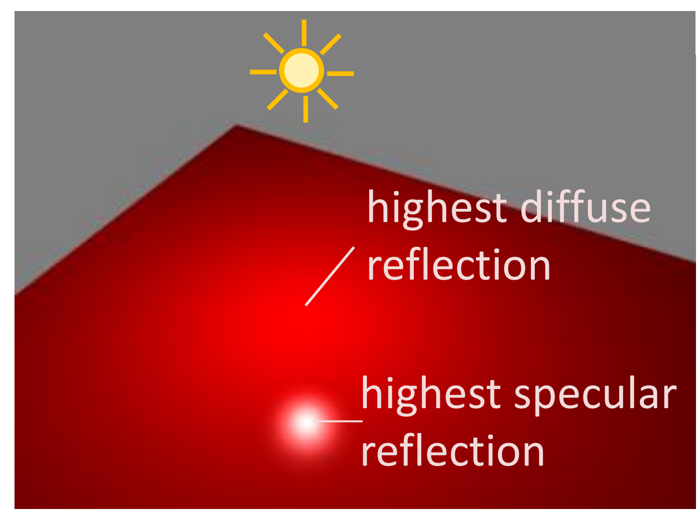
http://multivis.net/lecture/phong.html

Lighting

- Plane with diffuse and specular reflection
  - At which point is the highest specular reflection?
  - Reflected vector = view vector

Two of them are in different position

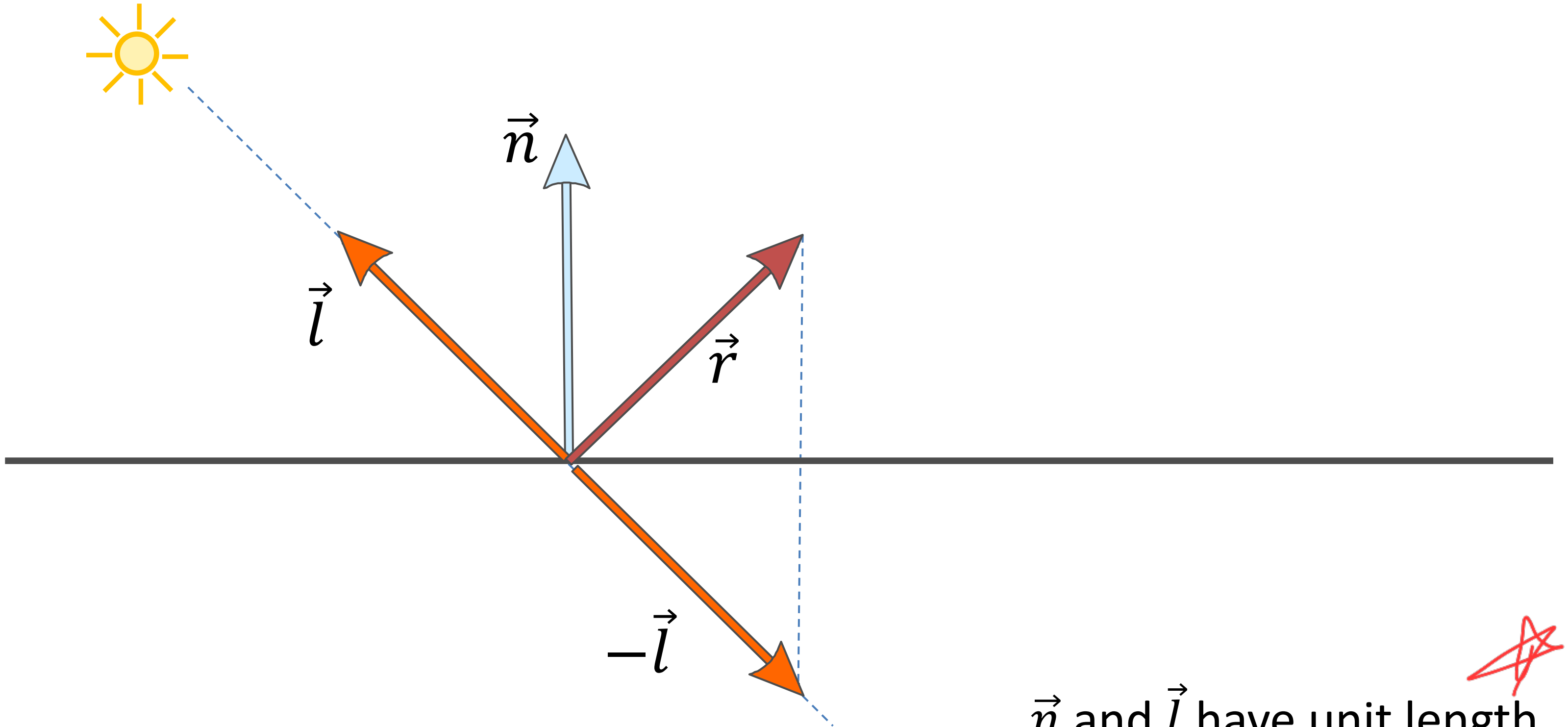




http://multivis.net/lecture/phong.html

Lighting

- Calculation of reflected light ray $\vec{r}$

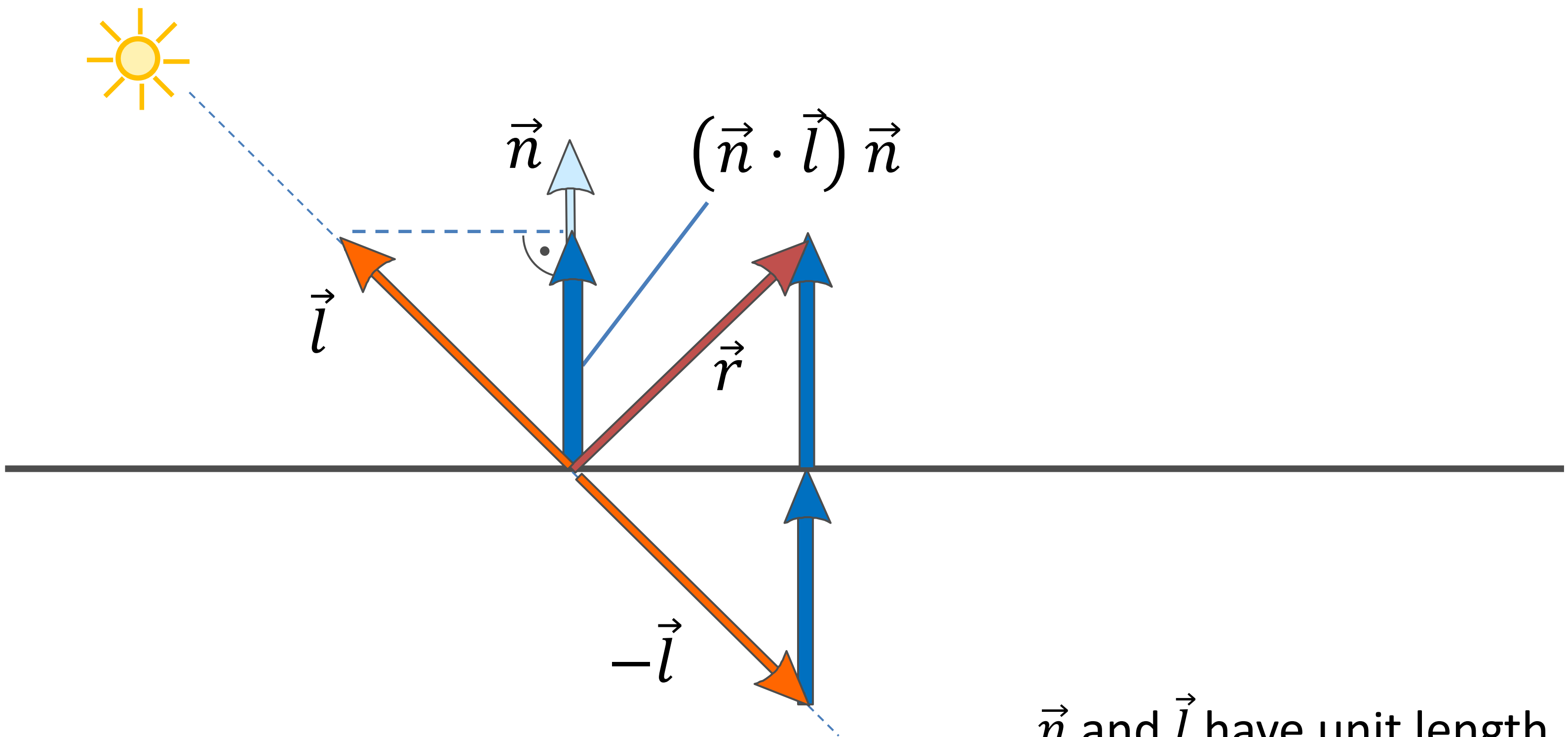


$\vec{n}$ and $\vec{l}$ have unit length

Lighting

- Calculation of reflected light ray $\vec{r}$

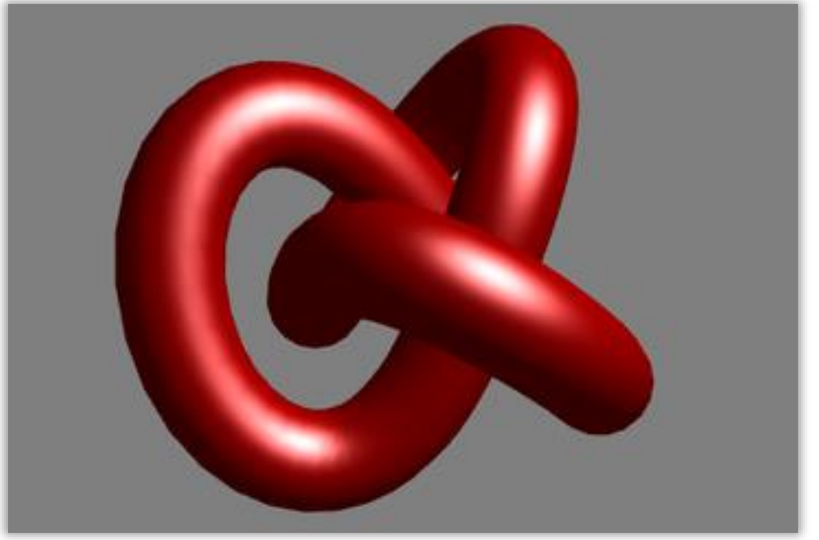
$$\vec{r} = 2(\vec{n} \cdot \vec{l})\,\vec{n} - \vec{l}$$



$\vec{n}$ and $\vec{l}$ have unit length

Lighting

SIEMENS
Ingenuity for life



```
varying vec3 normalInterp;  // Surface normal
varying vec3 vertexPos;     // Vertex position
uniform vec3 lightPos;      // Light position

uniform float Ka;   // Ambient reflection coefficient
uniform float Kd;   // Diffuse reflection coefficient
uniform float Ks;   // Specular reflection coefficient
uniform float shininess; // Shininess

// Material color
uniform vec3 ambientColor;
uniform vec3 diffuseColor;
uniform vec3 specularColor;

void main() {
   vec3 N = normalize(normalInterp);
   vec3 L = normalize(lightPos - vertexPos);

   // Lambert's cosine law
   float lambertian = max(dot(N, L), 0.0);      // use max-function to avoid negative values

   float specular = 0.0;
   if (lambertian > 0.0) {
      vec3 R = reflect(-L, N);        // Reflected light vector
      vec3 V = normalize(-vertexPos); // Vector to viewer with eye position (0, 0, 0)

      // Compute the specular term
      float specAngle = max(dot(R, V), 0.0);
      specular = pow(specAngle, shininess);
   }
   gl_FragColor = vec4(Ka * ambientColor +
                       Kd * lambertian * diffuseColor +
                       Ks * specular * specularColor, 1.0);
}
```

http://multivis.net/lecture/phong.html

Lighting

- Example
  - Ambient reflection coefficient $k_a = \frac{1}{10}$
  - Diffuse reflection coefficient $k_d = \frac{1}{2}$
  - Specular reflection coefficient $k_s = \frac{5}{6}$
  - Shininess factor $n = 2$
  - Light & object color white

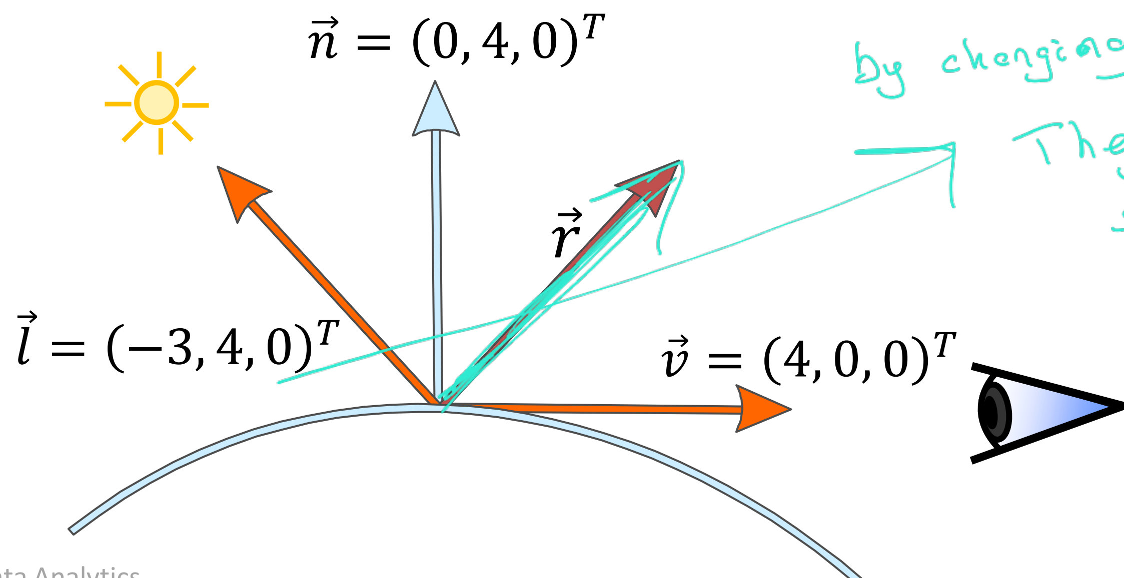
$C = k_s C_p O_d (\vec{r} \cdot \vec{v})^n$

$\frac{5}{6} C_p O_d \left(\frac{\begin{pmatrix} 3 \\ 4 \\ 0 \end{pmatrix} \begin{pmatrix} 4 \\ 0 \\ 0 \end{pmatrix}}{5 \cdot 4} \right)^2$

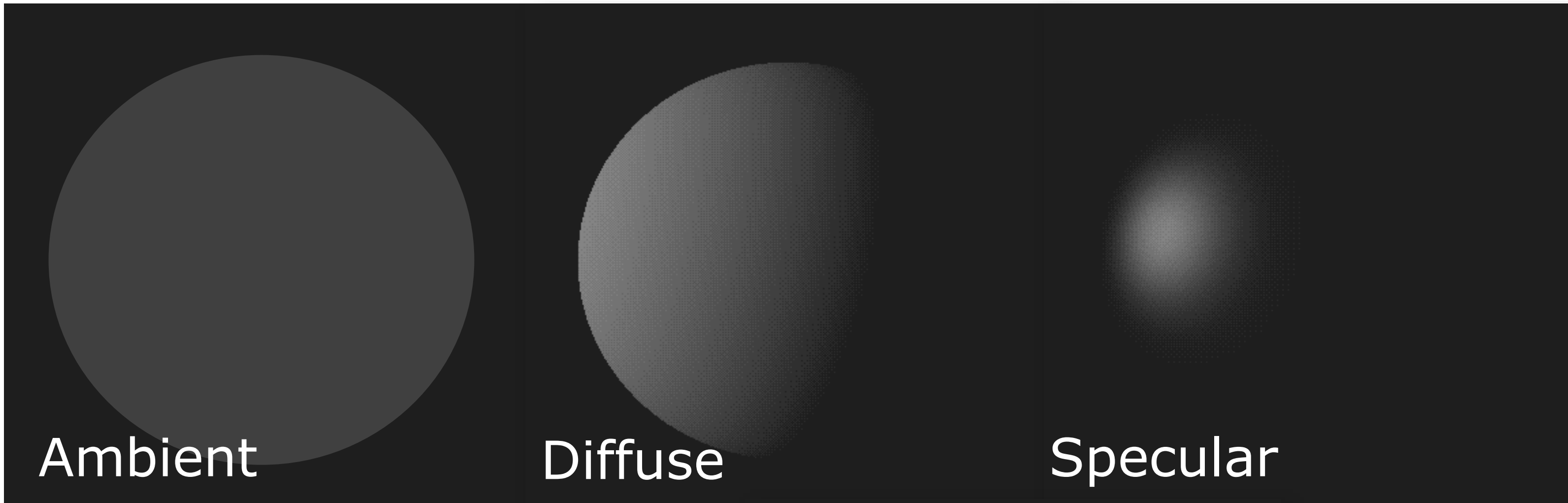
$= \frac{3}{10} C_p O_d$

We took inverse because we calculate r by changing x value

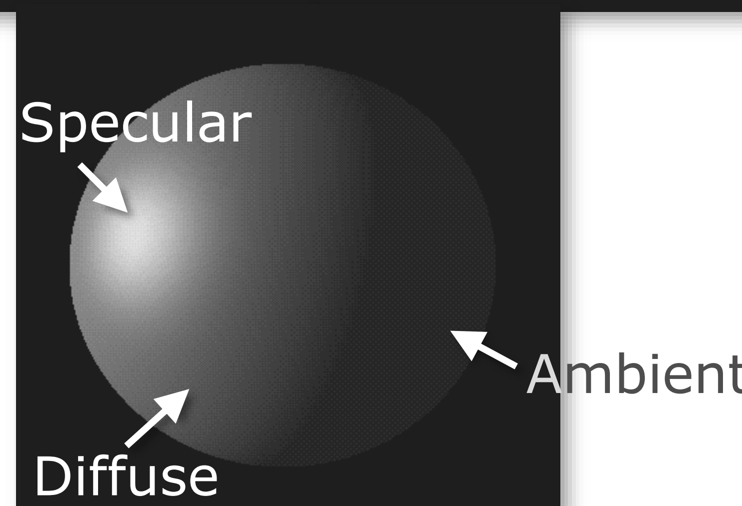
They are symetric



Lighting



$$k_a = 0.1$$
$$k_d = 0.5$$
$$k_s = 0.4$$

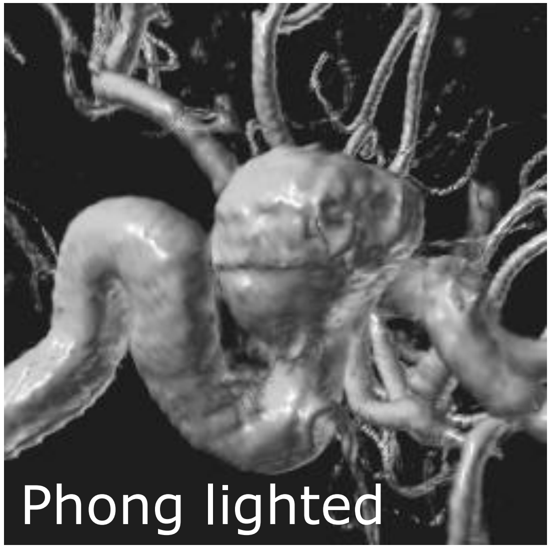


Phong lighted

Lighting

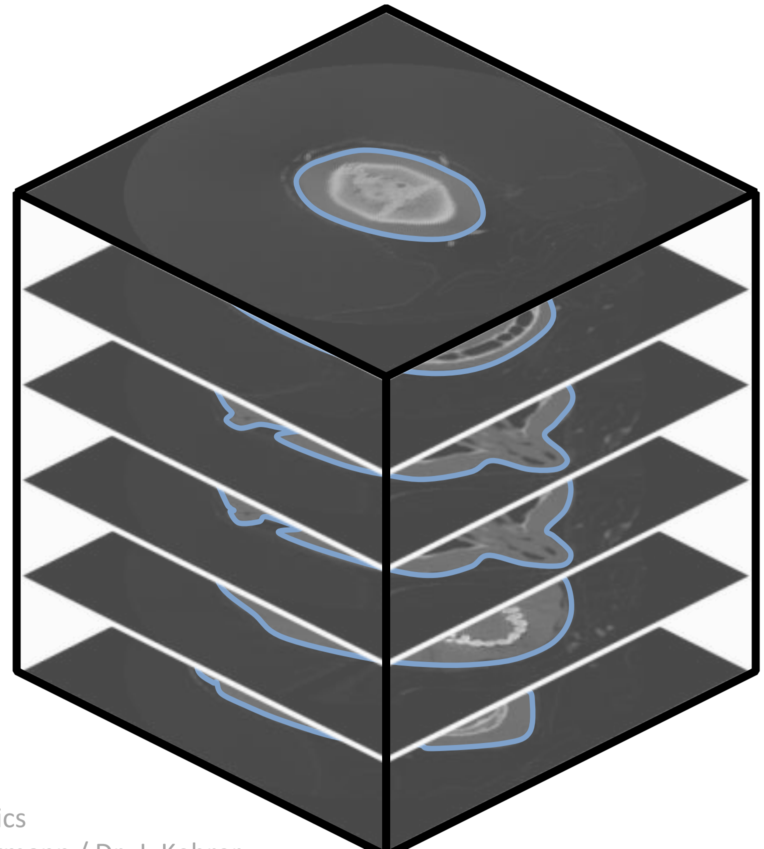
SIEMENS
Ingenuity for life

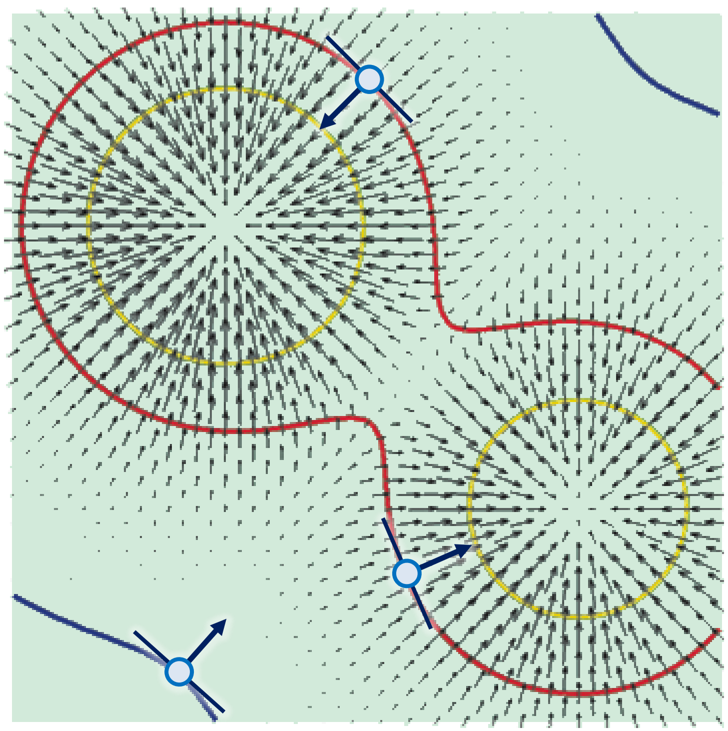
$$k_a = 0.1$$
$$k_d = 0.5$$
$$k_s = 0.4$$



Lighting

- What is the normal vector at a point on an iso-surface in a scalar field?
- It is the gradient at this point, which is perpendicular to the iso-surface





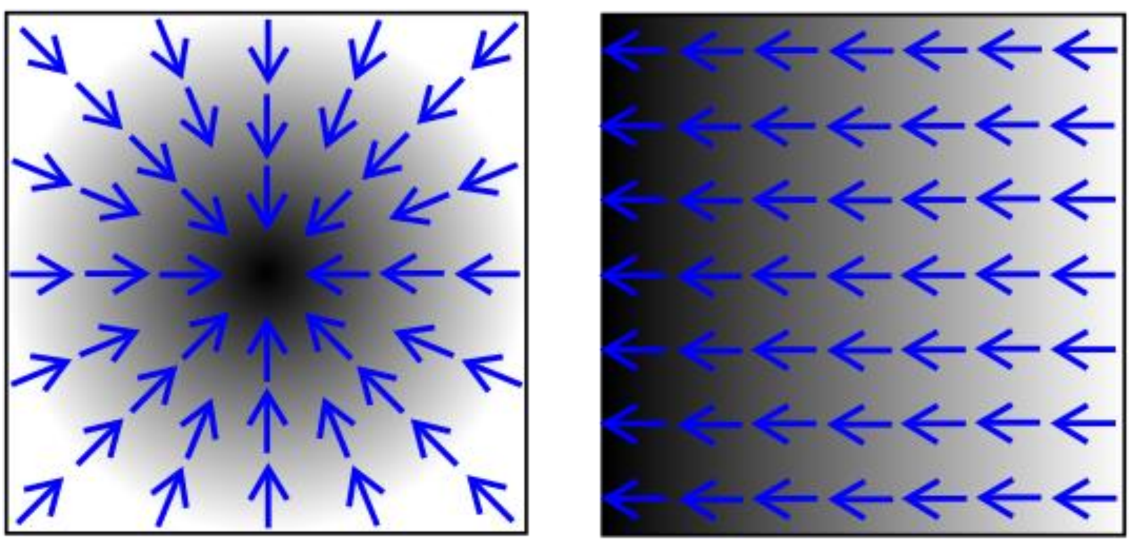
Lighting

SIEMENS
Ingenuity for life

- The gradient in a scalar field $f(\boldsymbol{x})$

$$\nabla f(\boldsymbol{x}) = \left(\frac{\partial}{\partial x} f(\boldsymbol{x}),\ \frac{\partial}{\partial y} f(\boldsymbol{x}),\ \frac{\partial}{\partial z} f(\boldsymbol{x})\right)^T$$

  - Partial derivatives of scalar field
  - The vector pointing into the direction of steepest ascent of f, with magnitude indicating the slope of the ascent
  - ∇ is the Nabla operator

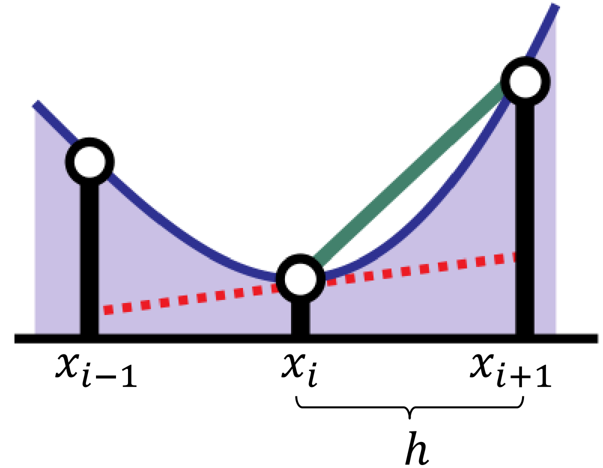


Lighting

SIEMENS
Ingenuity for life

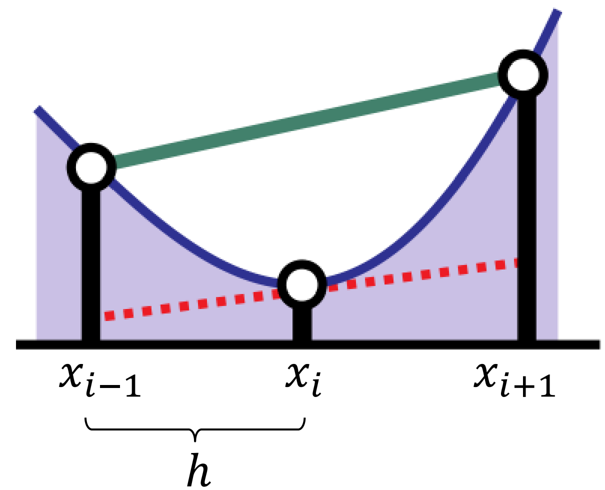
- Gradient approximation (1D)
  - Using forward differences

$$\frac{df}{dx}(x_i) \approx \frac{f(x_{i+1}) - f(x_i)}{h}$$



  - Using central differences

$$\frac{df}{dx}(x_i) \approx \frac{f(x_{i+1}) - f(x_{i-1})}{2h}$$

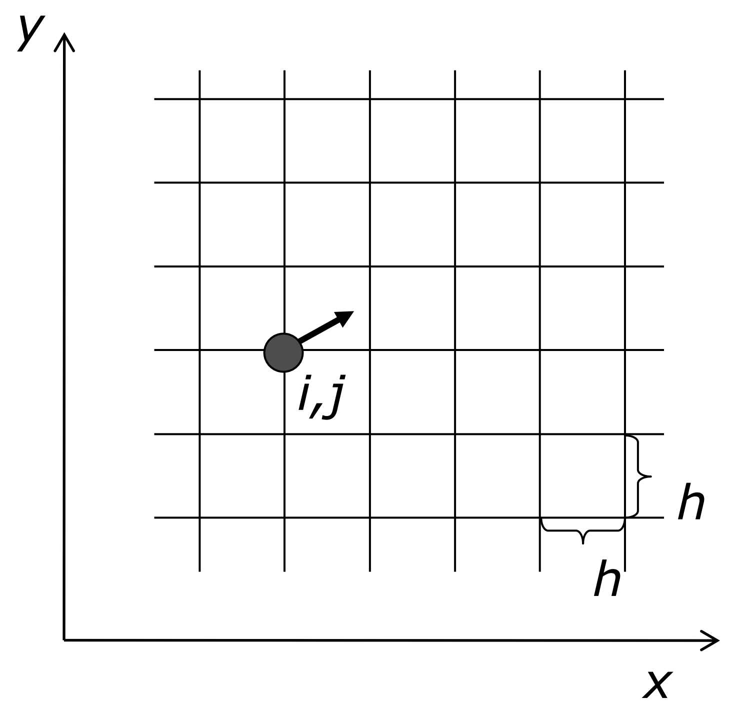


[Engel et al. 06]

Lighting

- Approximation of partial derivatives on uniform grids
  - Using forward differences

$$grad\ f = \nabla f = \begin{pmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \\ \frac{\partial f}{\partial z} \end{pmatrix} \approx \begin{pmatrix} \frac{f_{i+1,j,k} - f_{i,j,k}}{h} \\ \frac{f_{i,j+1,k} - f_{i,j,k}}{h} \\ \frac{f_{i,j,k+1} - f_{i,j,k}}{h} \end{pmatrix}$$

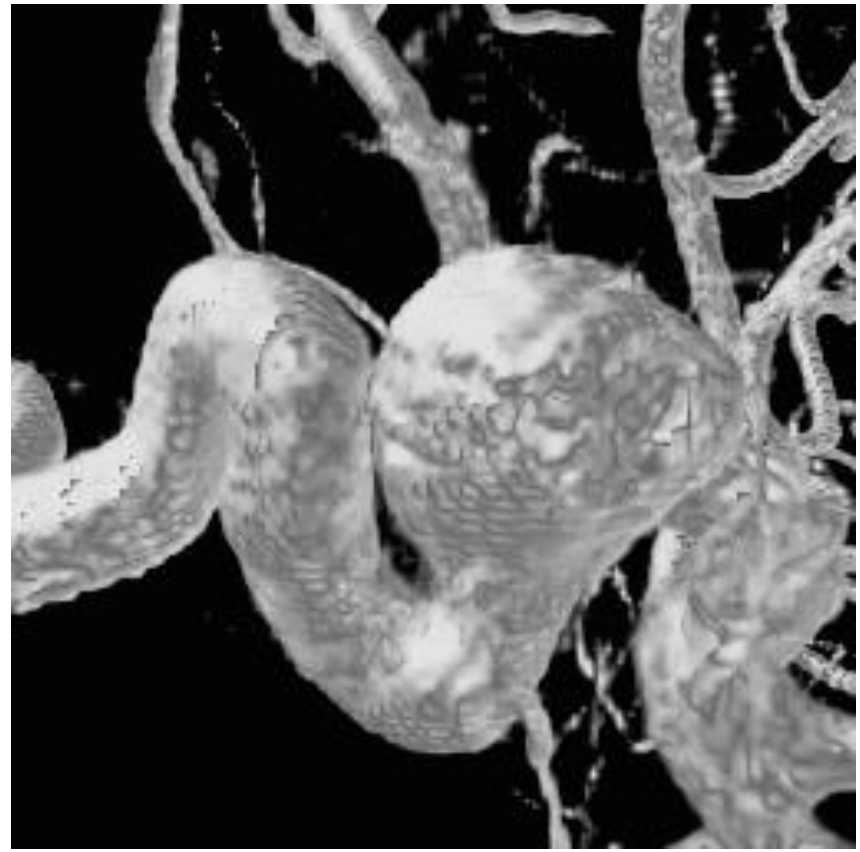


  - Using central differences

$$grad\ f = \nabla f = \begin{pmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \\ \frac{\partial f}{\partial z} \end{pmatrix} \approx \begin{pmatrix} \frac{f_{i+1,j,k} - f_{i-1,j,k}}{2h} \\ \frac{f_{i,j+1,k} - f_{i,j-1,k}}{2h} \\ \frac{f_{i,j,k+1} - f_{i,j,k-1}}{2h} \end{pmatrix}$$

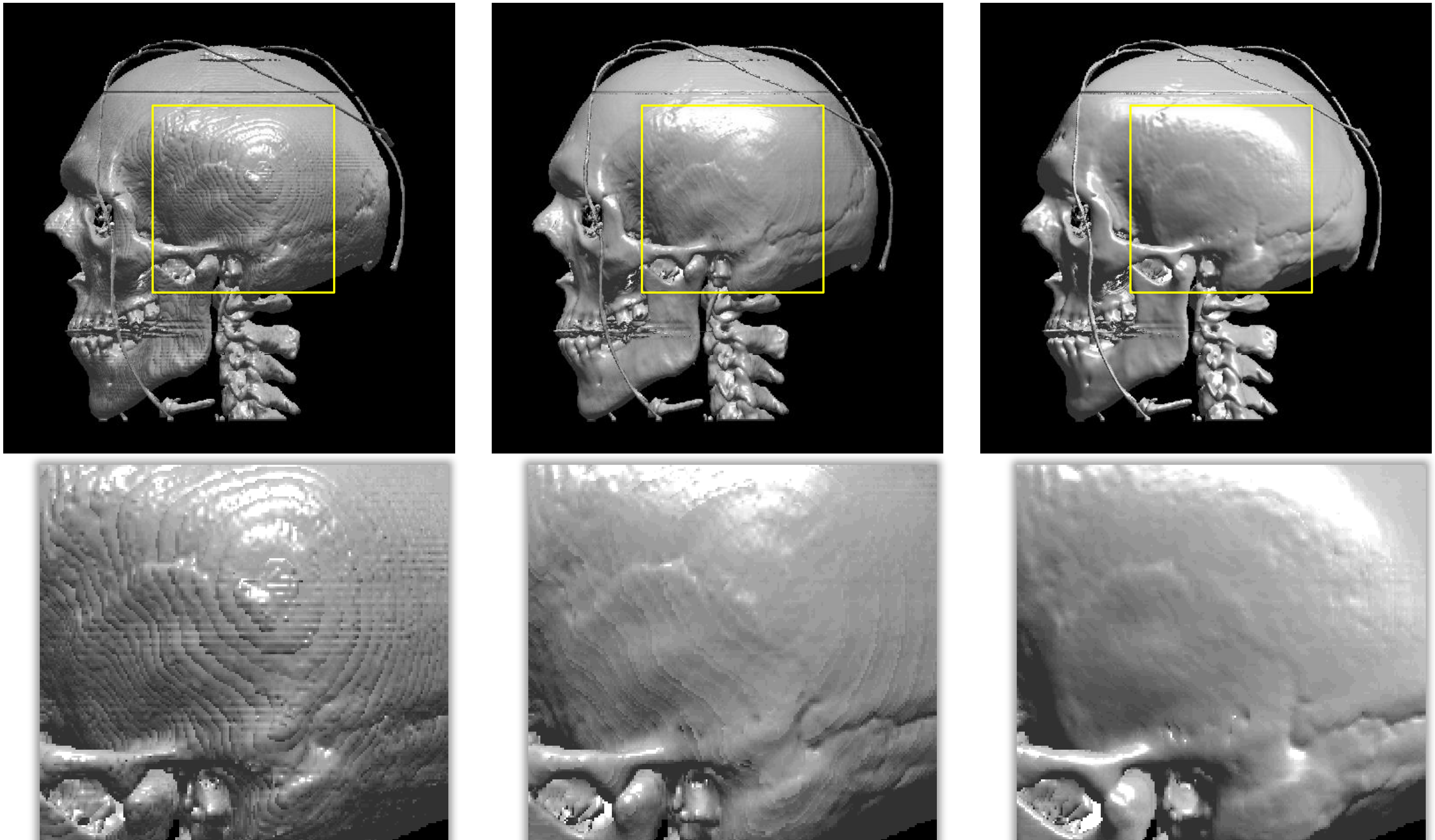
Lighting

Central differences



Forward differences

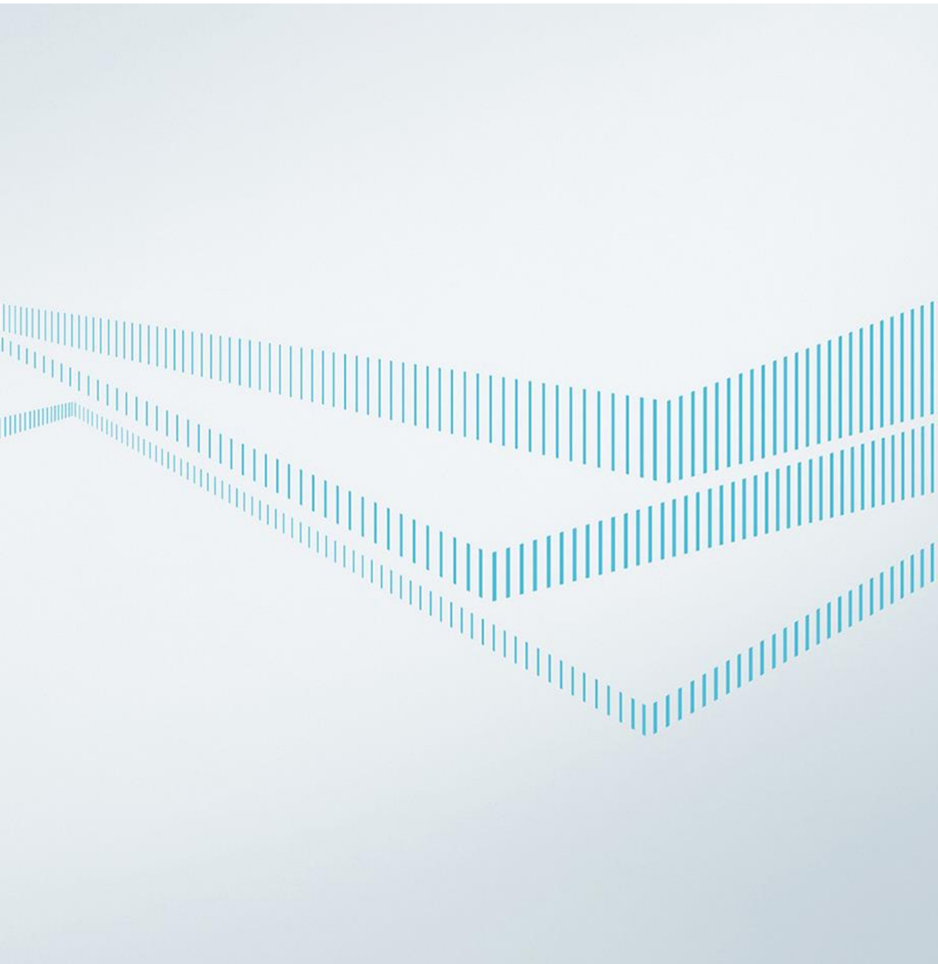
Lighting



Forward differences | Central differences | Sobel operator

Contact information

SIEMENS
Ingenuity for life



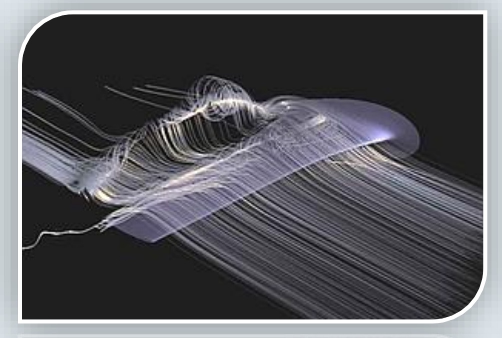
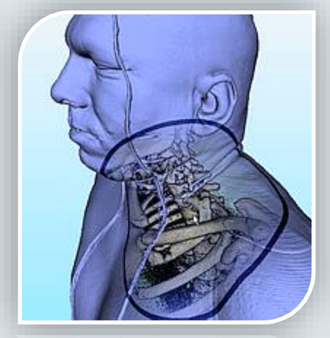
Dr. Johannes Kehrer

Siemens Technology
T DAI HCA-DE
Otto-Hahn-Ring 6
81739 München, Deutschland

E-mail:
kehrer.johannes@siemens.com

Internet
siemens.com/innovation

Intranet
intranet.ct.siemens.com


Visual Data Analytics

Isolines & Isosurfaces

Dr. Johannes Kehrer

Disclaimer

These lecture slides and materials are for personal use by students and are intended for educational purposes only. Distribution or reproduction of the material is prohibited.

The views and opinions expressed by any individual during the course of these lecturers are their own and do not necessarily represent the views, opinions, or positions of Siemens or the Technical University of Munich.

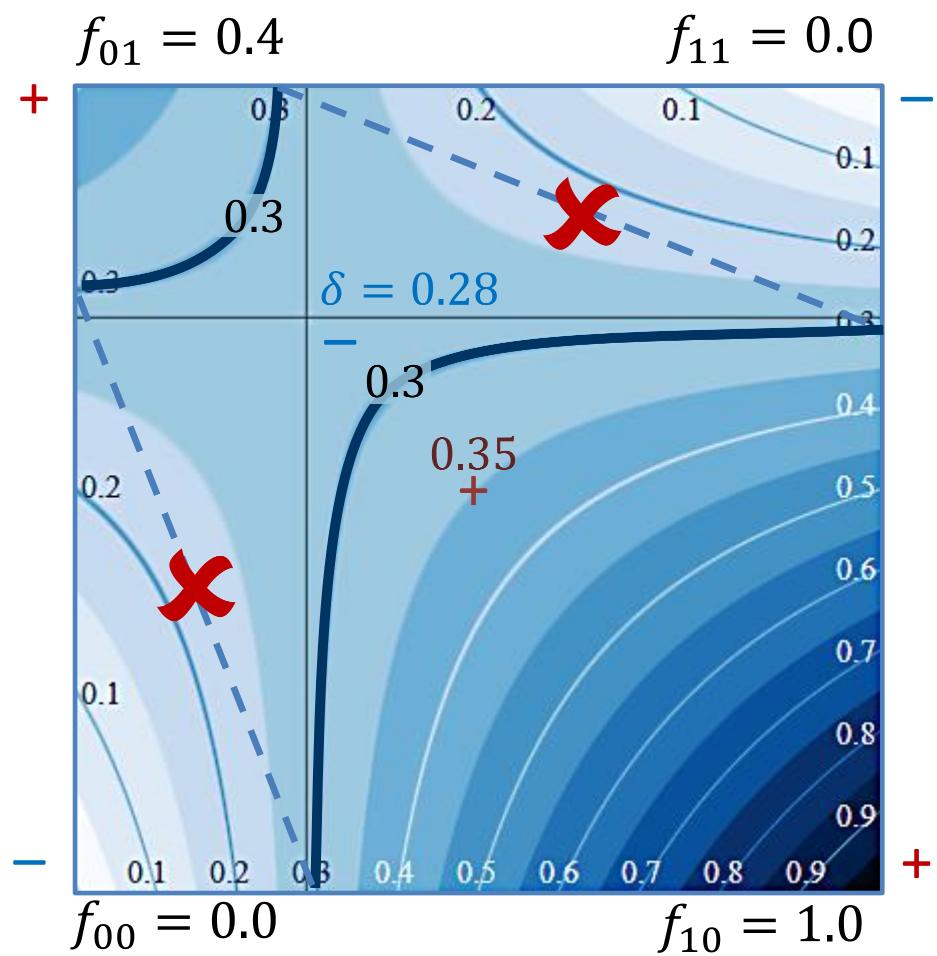
Although reasonable efforts have been made to ensure the information in these lecture slides is current and accurate, we do not assume any responsibility for the content of the posted material.

These slides also contain links to websites of third parties. As the content of these websites is not under our control, we cannot assume any liability for such external content.

Isolines

SIEMENS
Ingenuity for life

- When does the midpoint decider **not** work
  - **Example:** Iso-line for $c = 0.3$
  - Value at midpoint: 0.35
  - Evaluating the asymptotic decider, we get $\delta = 0.28$ at intersection of asymptotes
  - The midpoint decider would give us wrong isolines

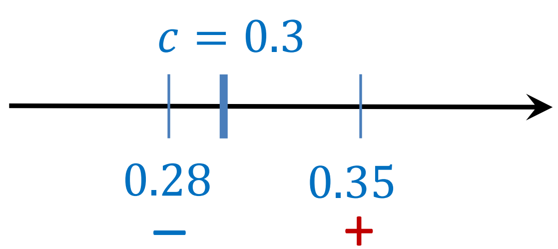


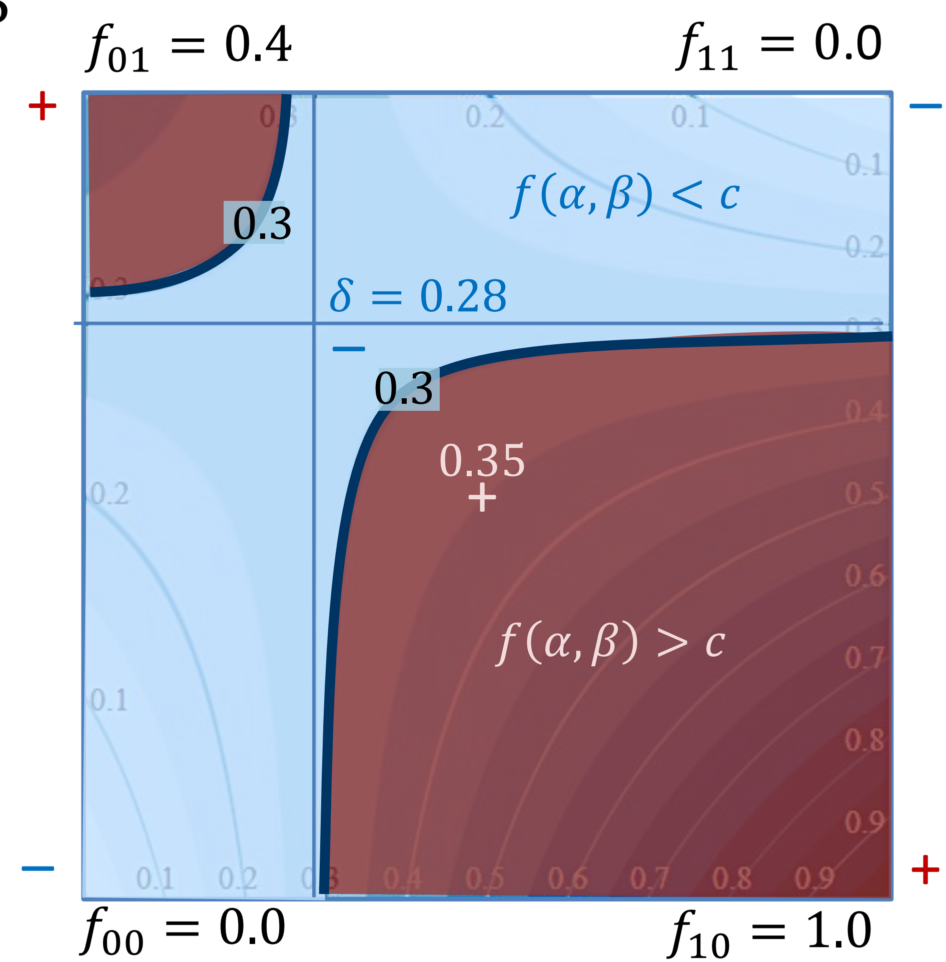
multivis.net/lecture/bilinear.html

Isolines

SIEMENS
Ingenuity for life

- When does the midpoint decider **not** work
  - **Example:** Iso-line for $c = 0.3$
  - Value at midpoint: 0.35
  - Evaluating the asymptotic decider, we get $\delta = 0.28$ at intersection of asymptotes
  - The midpoint decider would give us wrong isolines

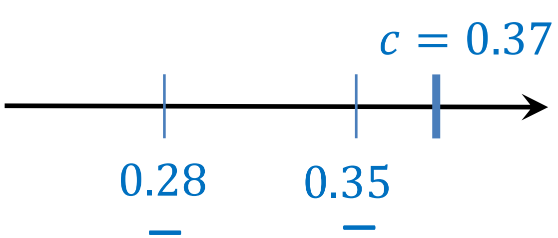


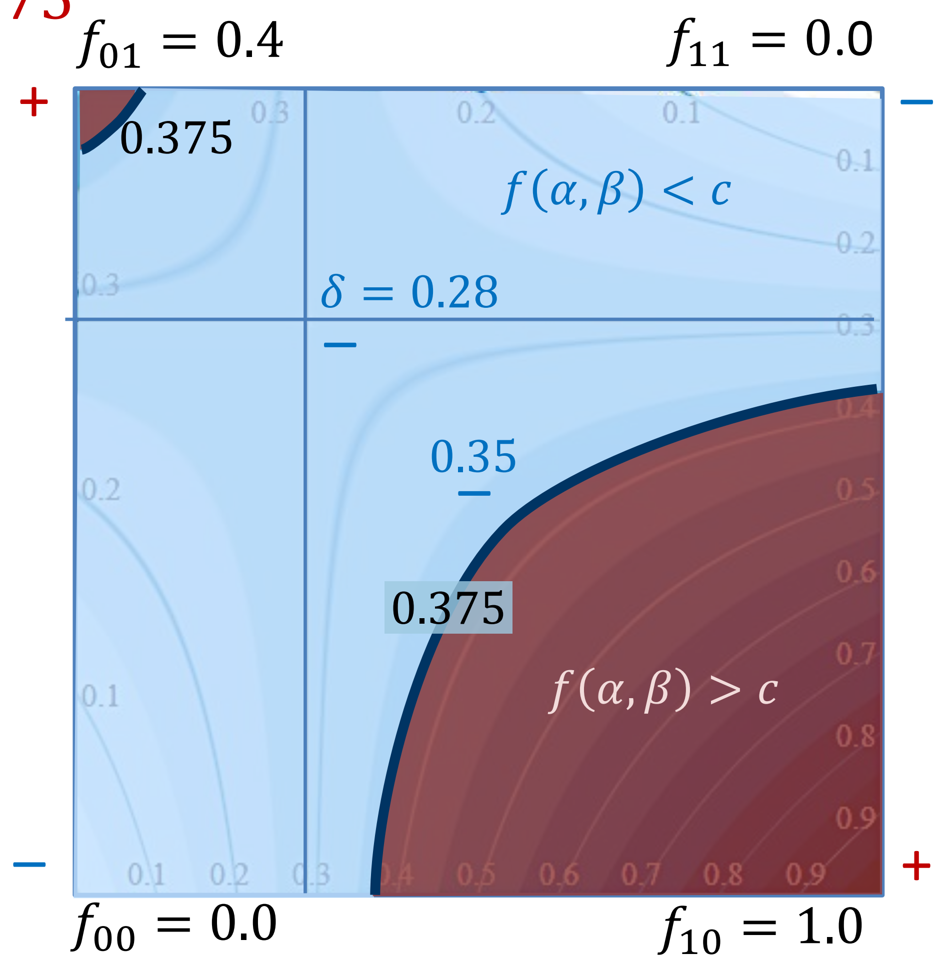


multivis.net/lecture/bilinear.html

Isolines

- When does the midpoint decider work
  - **Example:** Iso-line for $c = 0.375$
  - Value at midpoint: 0.35
  - Evaluating the asymptotic decider, we get $\delta = 0.28$ at intersection of asymptotes
  - The midpoint decider would give us the correct isolines



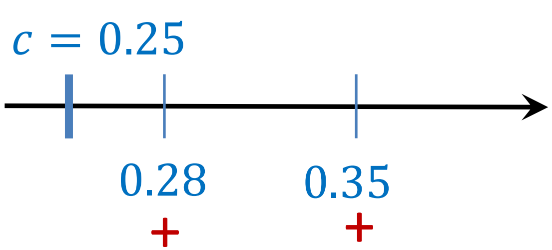


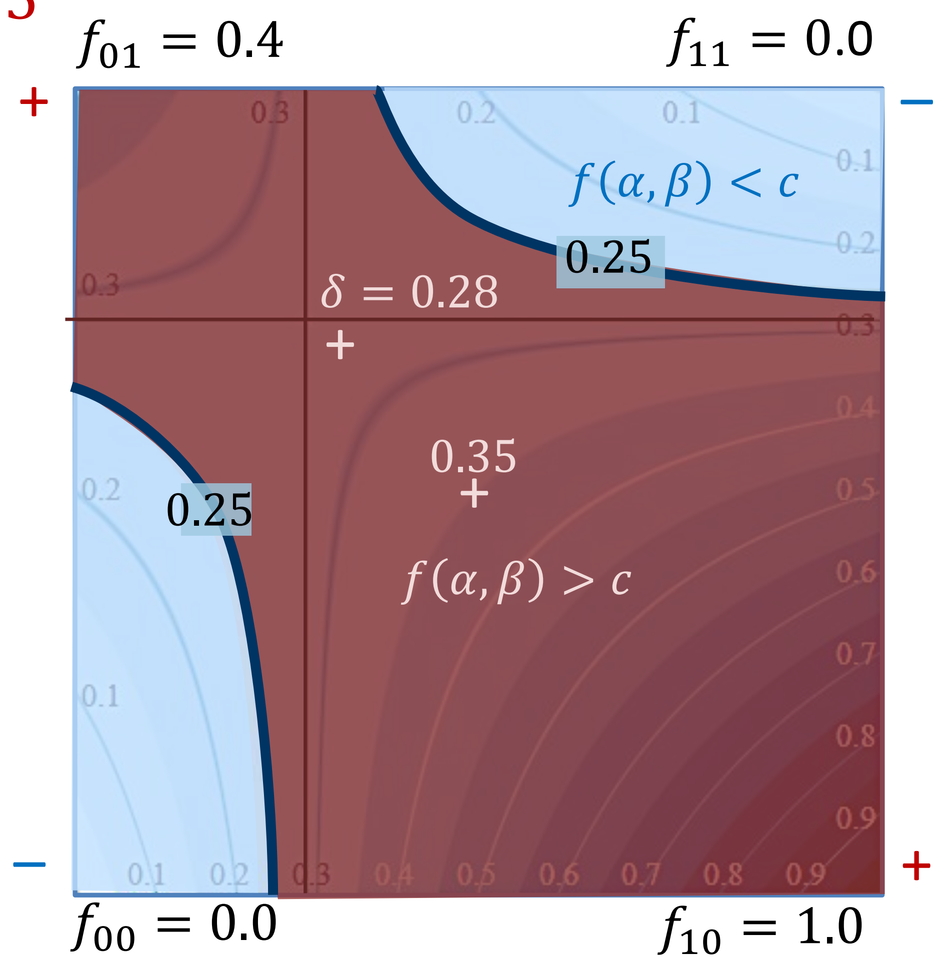
multivis.net/lecture/bilinear.html

Isolines

SIEMENS
Ingenuity for life

- When does the midpoint decider work
  - **Example:** Iso-line for $c = 0.25$
  - Value at midpoint: 0.35
  - Evaluating the asymptotic decider, we get $\delta = 0.28$ at intersection of asymptotes
  - The midpoint decider would give us the correct isolines

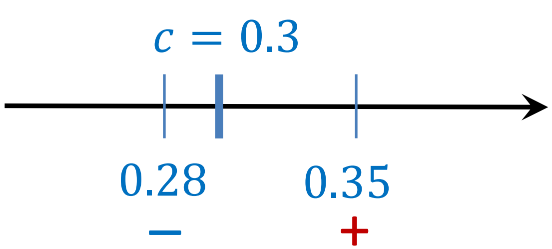




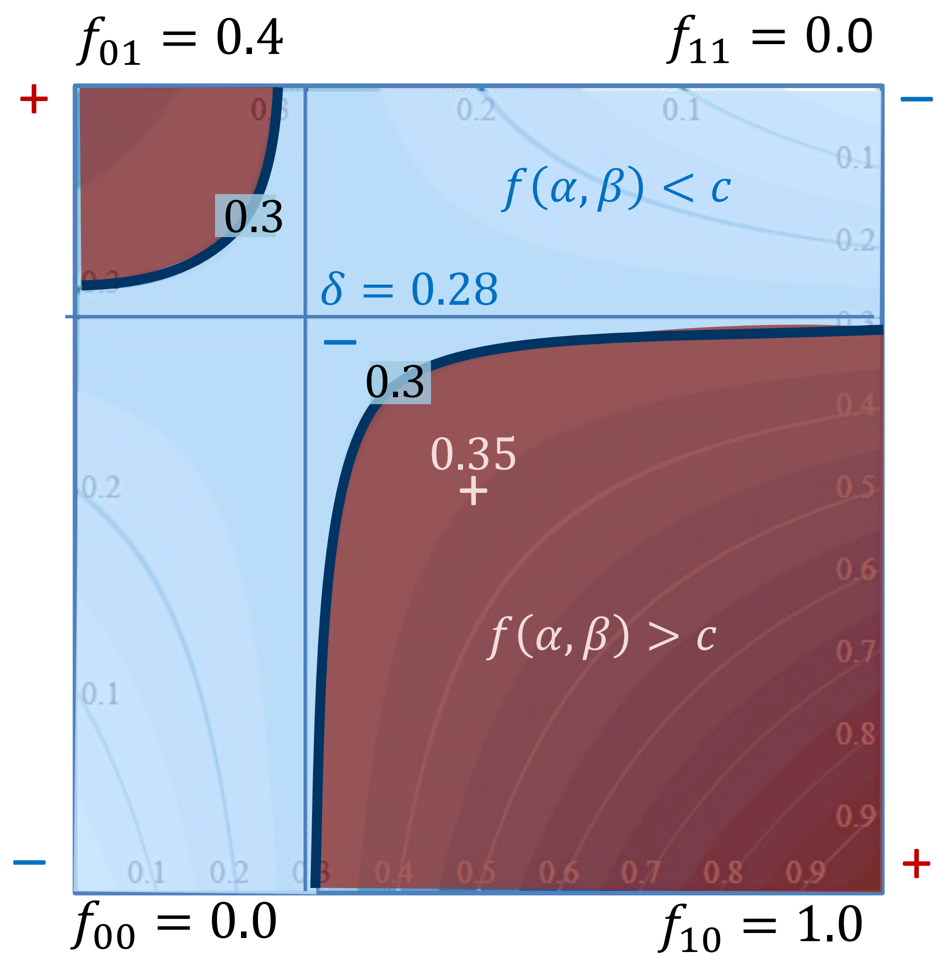
multivis.net/lecture/bilinear.html

Isolines

- When does the midpoint decider **not** work
  - **Example:** Iso-line for $c = 0.3$
  - Value at midpoint: 0.35
  - Evaluating the asymptotic decider, we get $\delta = 0.28$ at intersection of asymptotes
  - The midpoint decider would give us wrong isolines



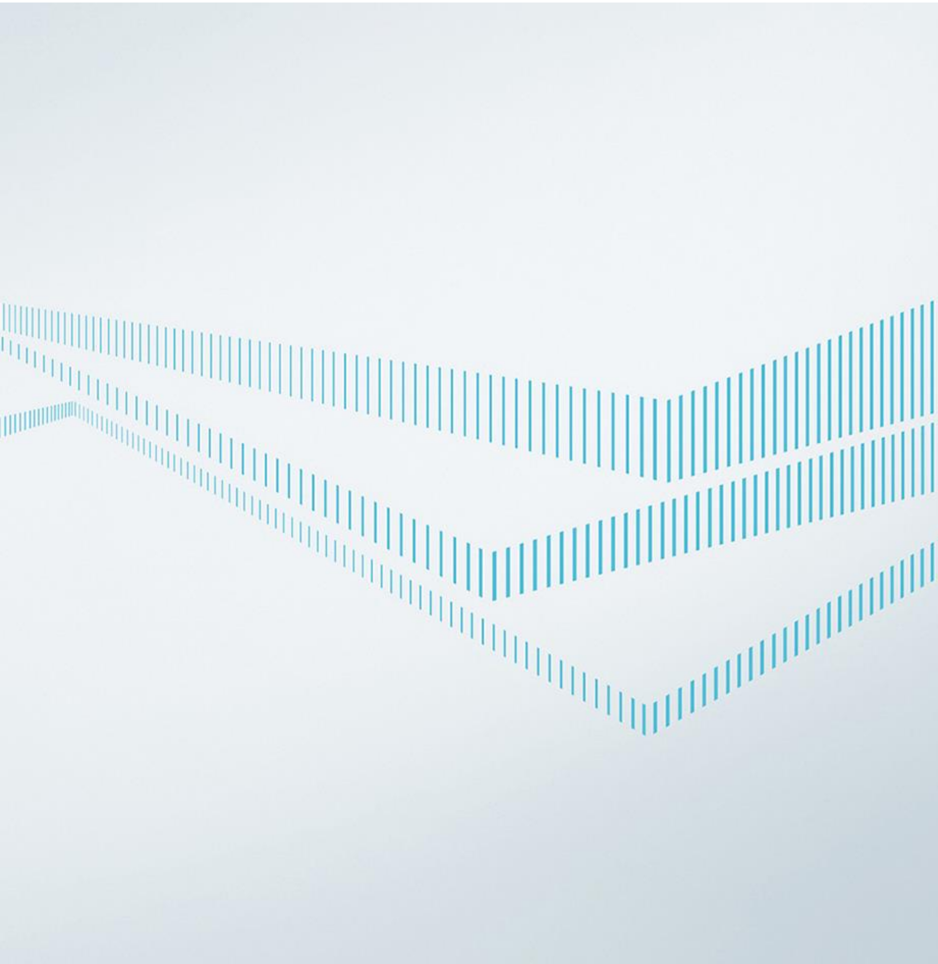
  - **Incorrect results for $0.28 < c < 0.35$**



multivis.net/lecture/bilinear.html

Contact information

SIEMENS
Ingenuity for life



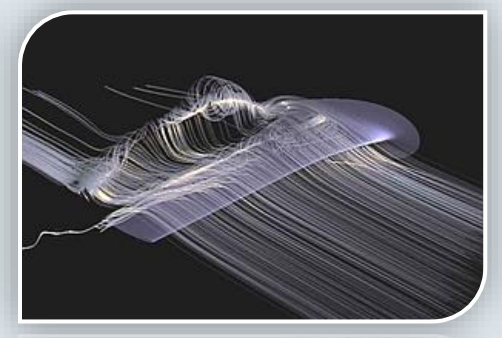
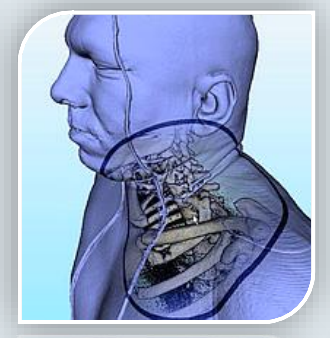
Dr. Johannes Kehrer

Siemens Technology
T DAI HCA-DE
Otto-Hahn-Ring 6
81739 München, Deutschland

E-mail:
kehrer.johannes@siemens.com

Internet
siemens.com/innovation

Intranet
intranet.ct.siemens.com

Visual Data Analytics

Volume Visualization

Dr. Johannes Kehrer

Disclaimer

These lecture slides and materials are for personal use by students and are intended for educational purposes only. Distribution or reproduction of the material is prohibited.

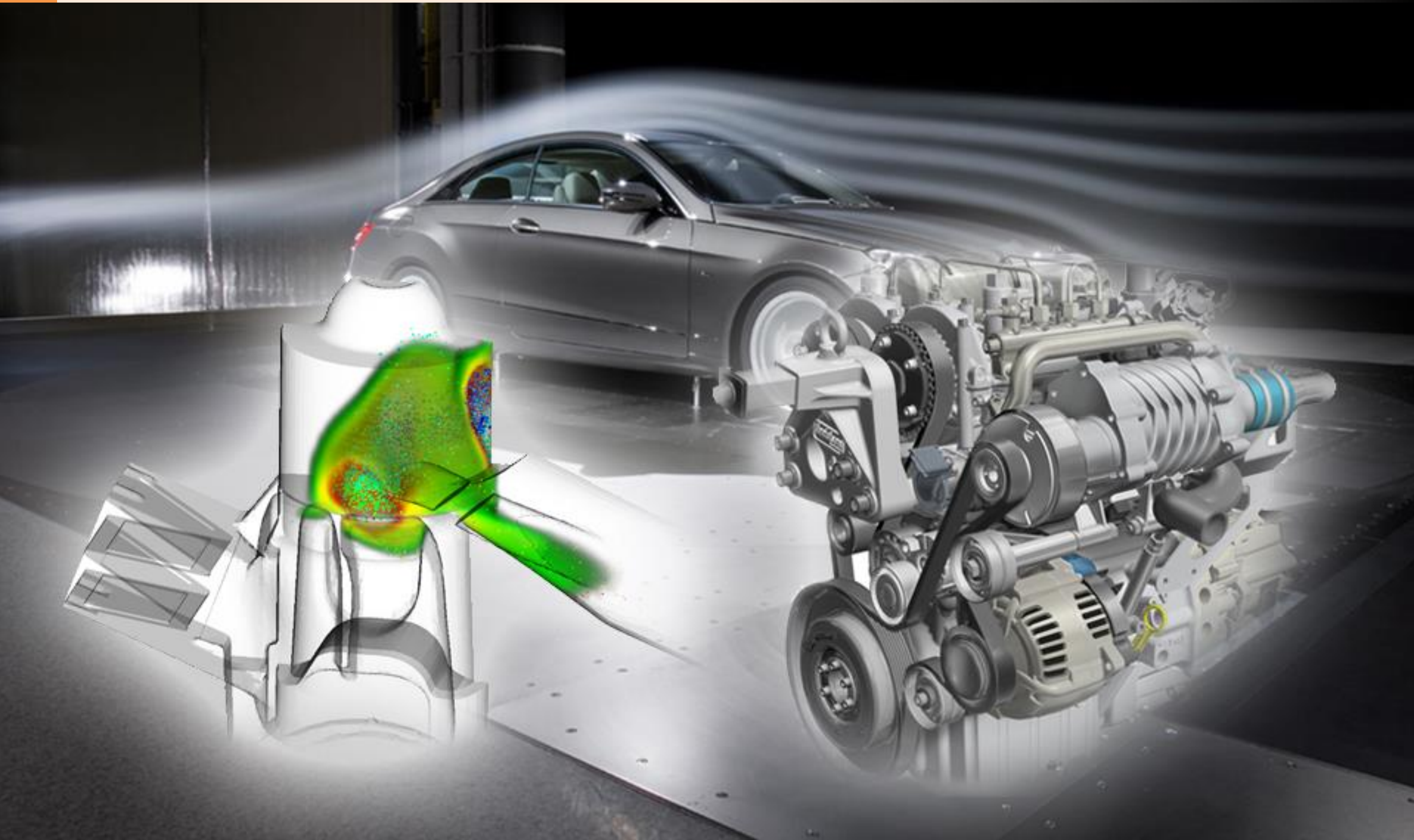
The views and opinions expressed by any individual during the course of these lecturers are their own and do not necessarily represent the views, opinions, or positions of Siemens or the Technical University of Munich.

Although reasonable efforts have been made to ensure the information in these lecture slides is current and accurate, we do not assume any responsibility for the content of the posted material.

These slides also contain links to websites of third parties. As the content of these websites is not under our control, we cannot assume any liability for such external content.

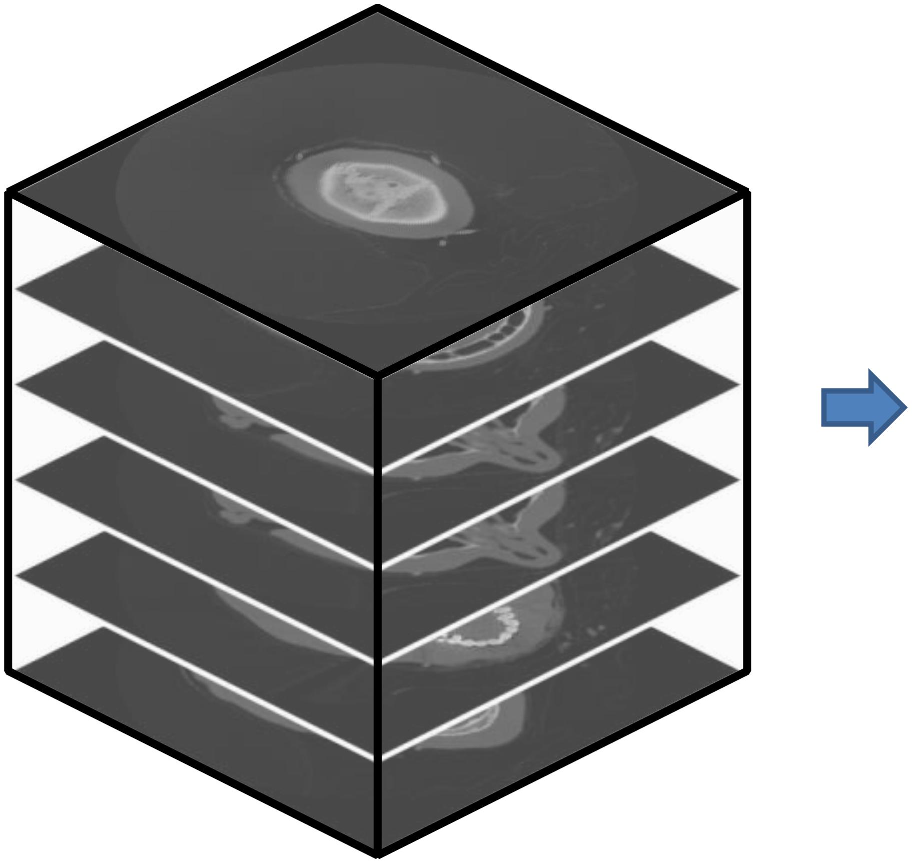
Medical scanners

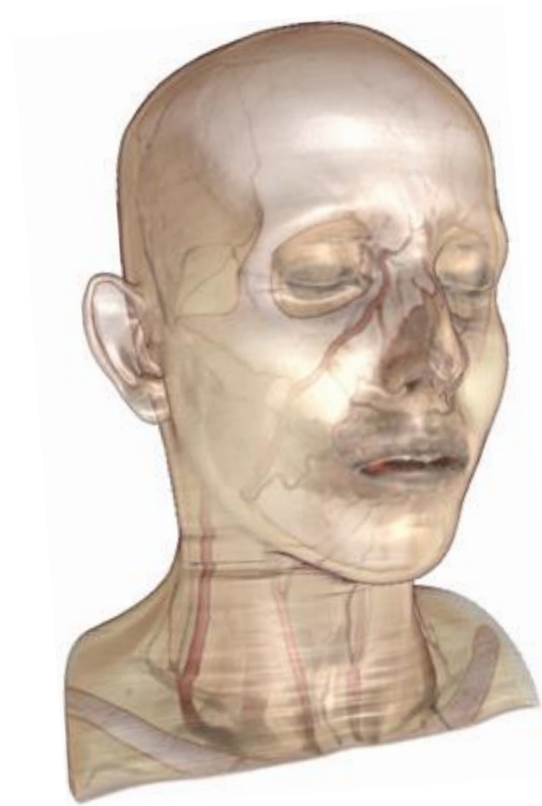
Automotive Engineering



Volume visualization

SIEMENS
Ingenuity for life

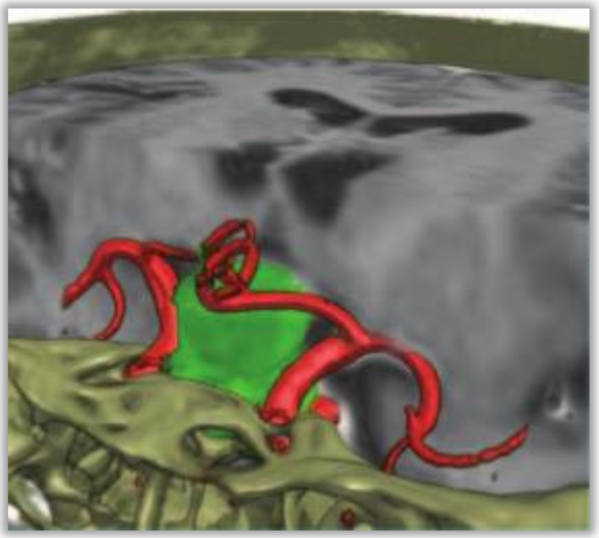


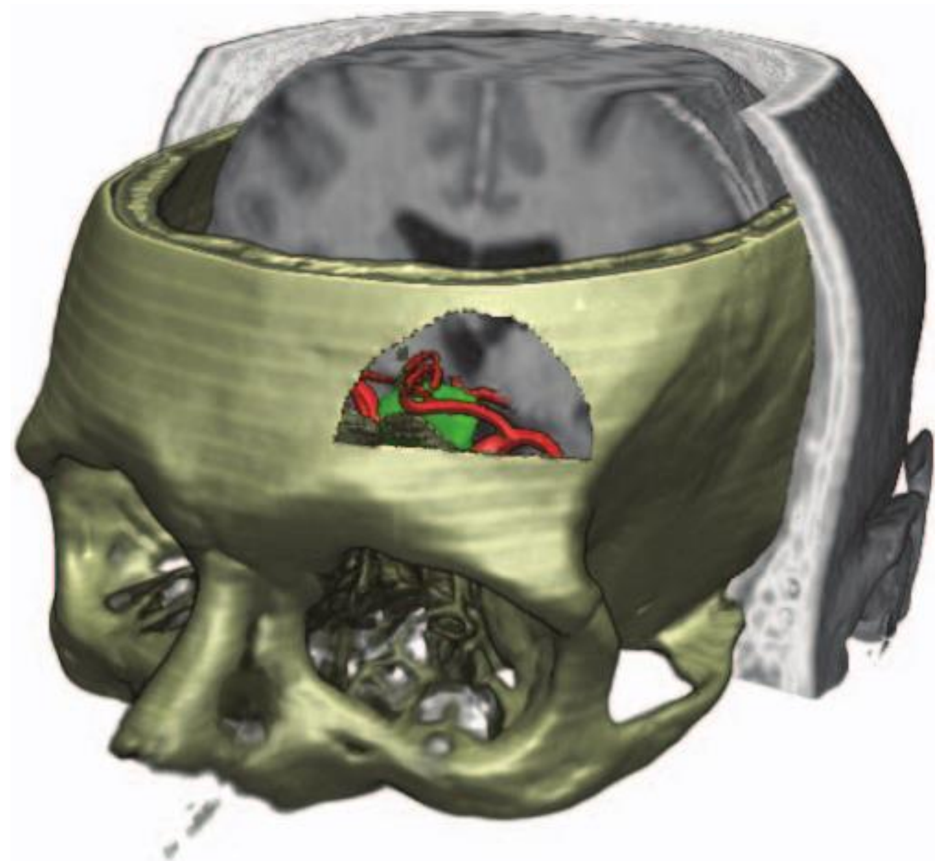


Visualization – Examples

SIEMENS
Ingenuity for life

- Preoperative planning of a tumor resection





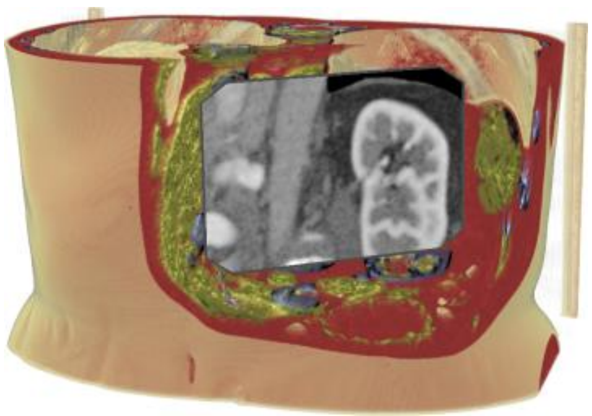
[Beyer et al. 2007]

Black/white: brain – Magnetic Resonance Imaging (MRI)
Green: tumor – MRI
Red: vessels – Magnetic Resonance Angiogram (MRA)
Brown: skull – Computer Tomography (CT)

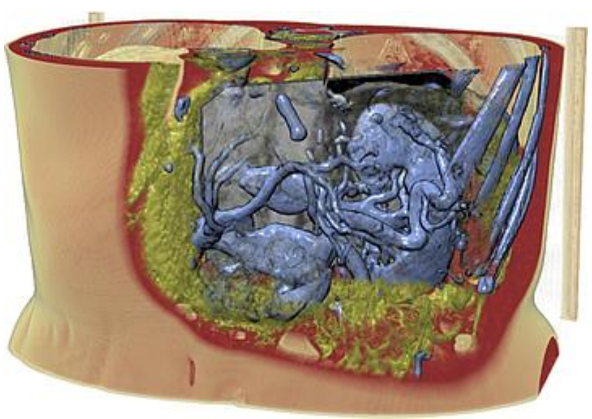
Visualization – Examples

SIEMENS
Ingenuity for life

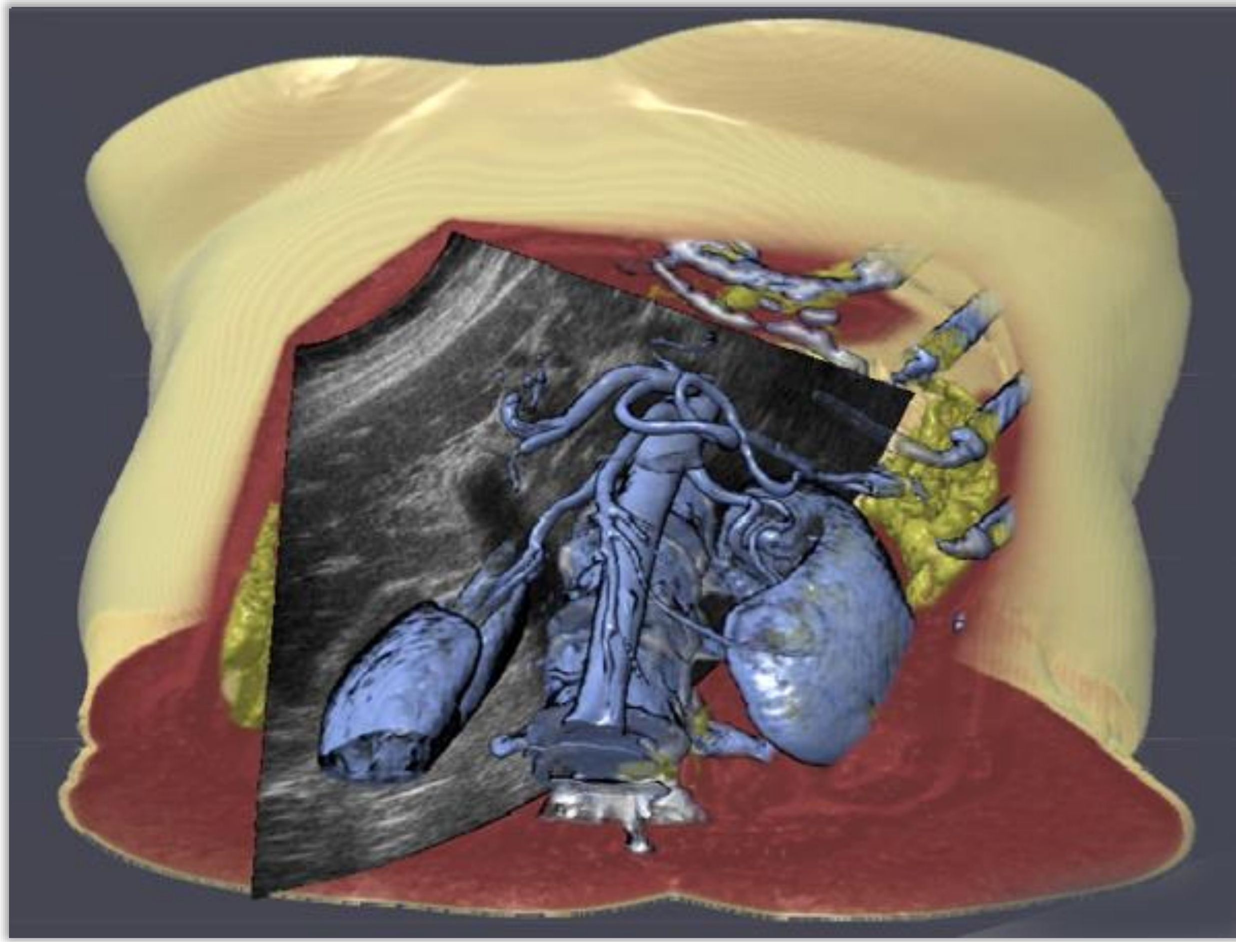
- CT scan with embedded Ultrasound data



Traditional cutaway



Contextual cutaway

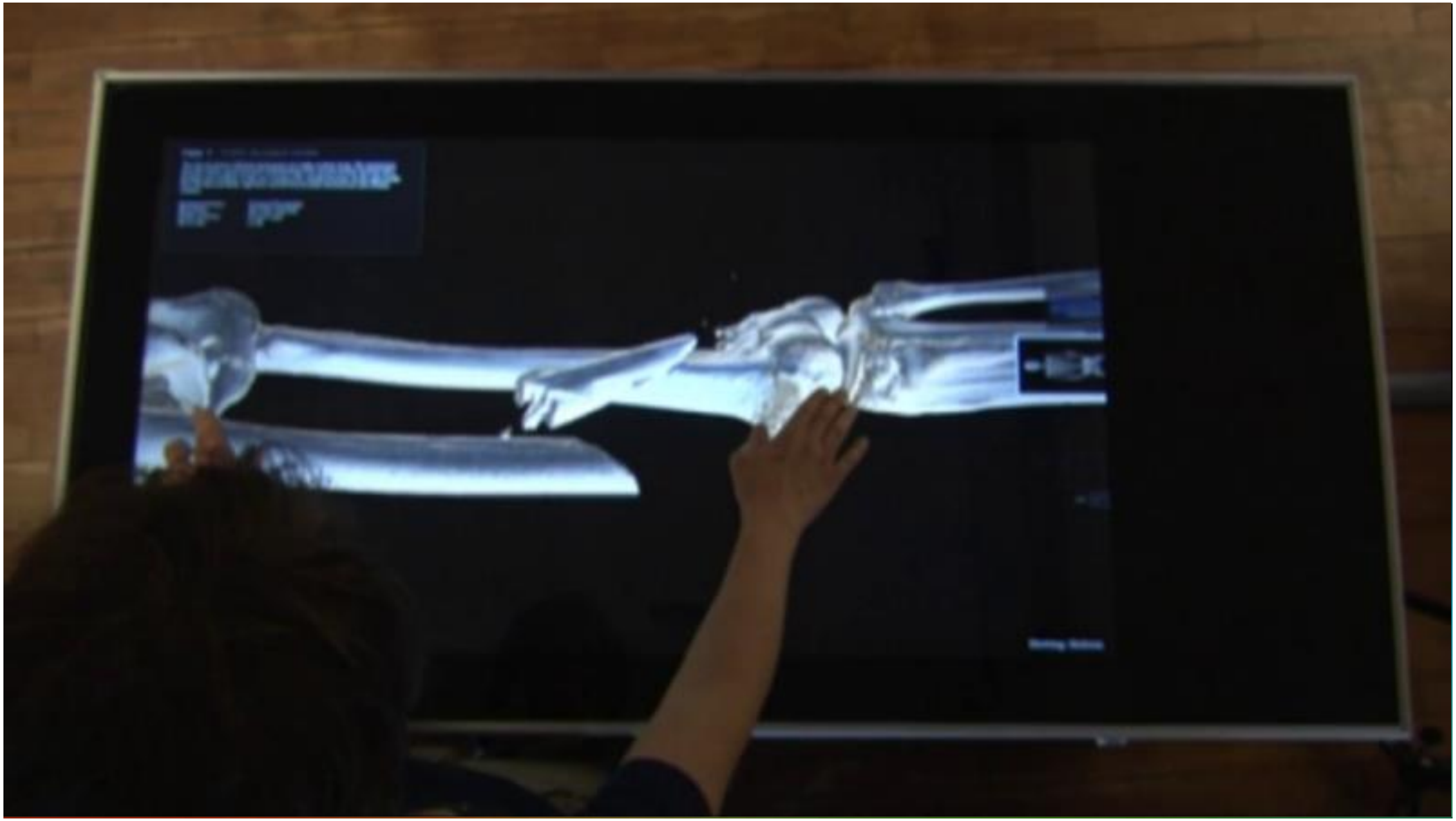


[Burns et al. 2007]

Visualization – Examples

SIEMENS
Ingenuity for life

- Virtual Autopsy

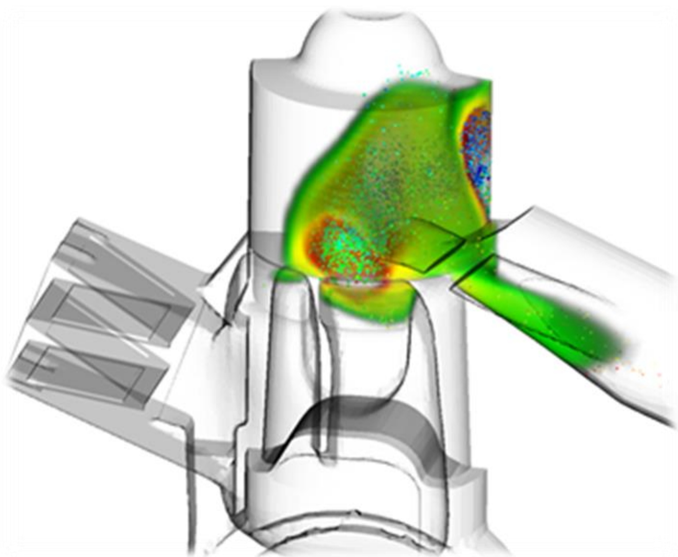


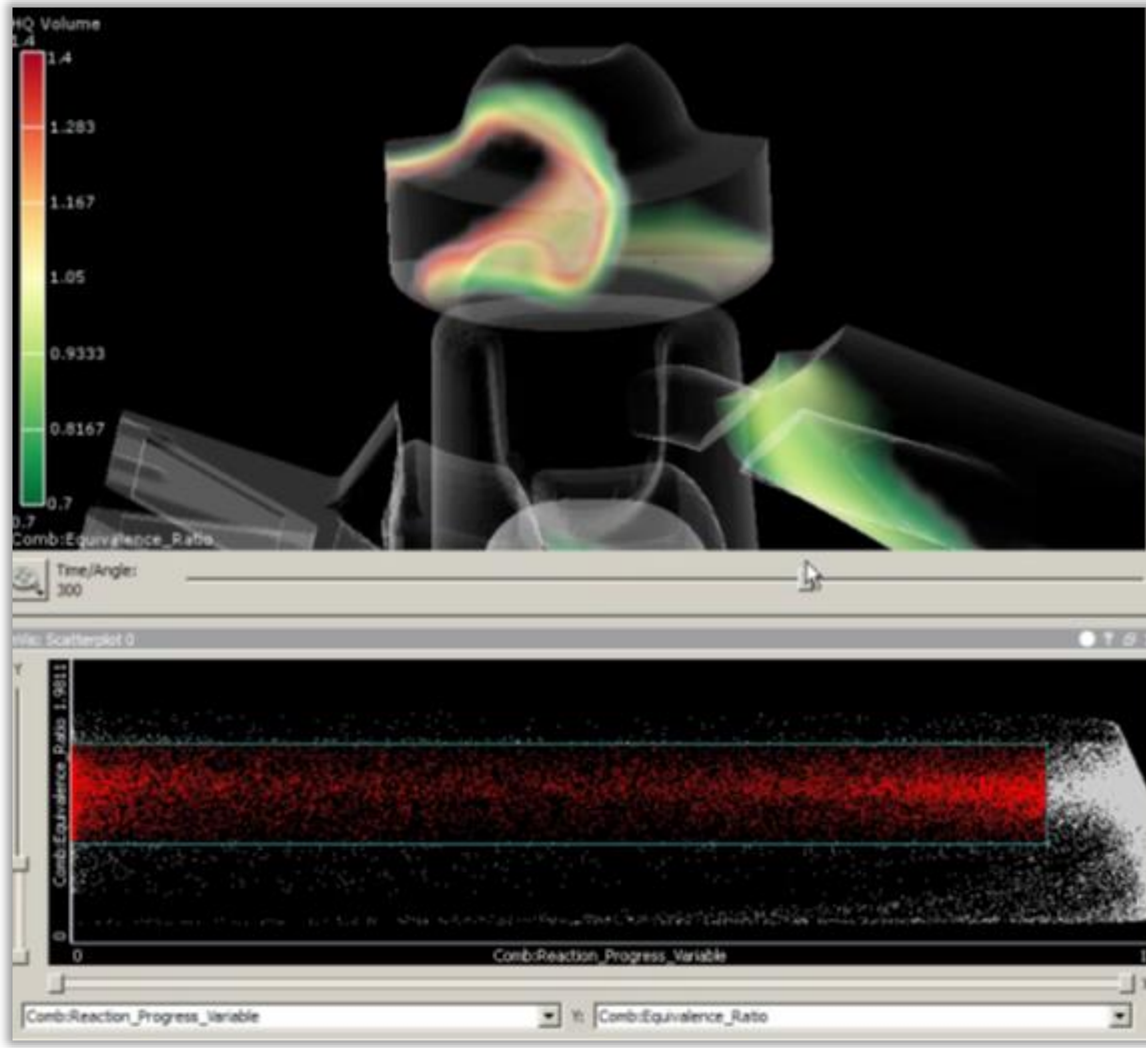
https://vimeo.com/6866296

Visualization – Examples

SIEMENS
Ingenuity for life

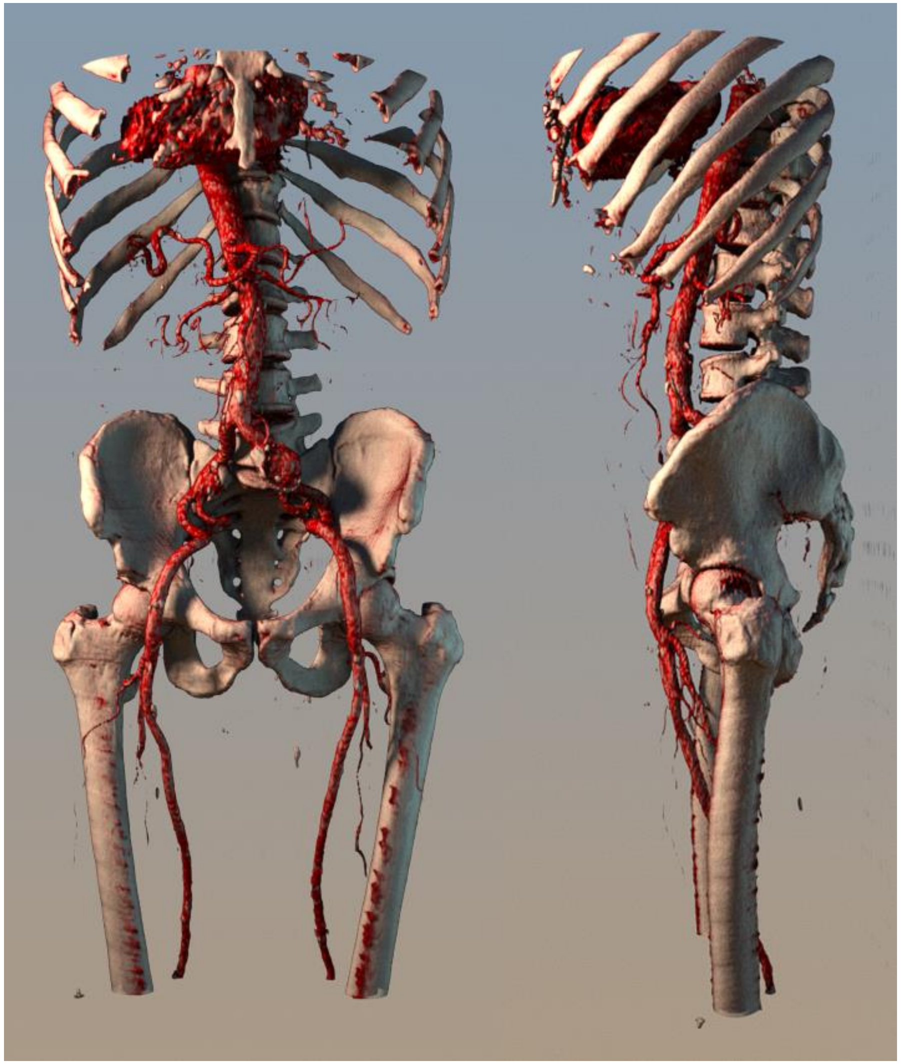
- Combustion process in a two-stroke engine





Visualization – Examples

- Physically-based lighting



[Kroes et al. 2012]

Volume visualization

- Volume rendering techniques
  - Techniques for 2D scalar fields (e.g. slicing)
  - Indirect volume rendering techniques (e.g. surface fitting)
    - Convert/reduce volume data to intermediate representation (surface representation), which can be rendered with traditional techniques – MC-algorithm
  - Direct volume rendering techniques
    - Consider the data as a semi-transparent gel with physical properties and directly get a 3D representation of it

Direct ve indirect volume rendering arasındaki fark nedir ?

Direct volume rendering, veri setinin doğrudan ekrana yansıtılmasını sağlar. Bu yöntem, veri setinin voxel değerlerine göre bir renk ataması yaparak, görüntüde gösterilen nesnelerin kalınlıklarını ve şekillerini belirler. Bu yöntem, veri setinin tamamını bir anda görüntüler ve katmanlı görüntüler oluşturmaz.

Indirect volume rendering ise, veri setinin önceden hesaplanan görüntülerini kullanır. Bu görüntüler, veri setinin voxel değerlerine göre oluşturulan "örtü"lerdir. Örtüler, veri setinin bir kesiti olarak düşünülebilir ve her bir örtü, veri setinin bir zaman diliminde ne gibi değişiklikler gösterdiğini gösterir. Bu örtüler arasında geçiş yaparak, veri setinin değişimlerini izleyebilirsiniz.

Direct volume rendering yöntemi, veri setinin tamamını bir anda görüntülerken, indirect volume rendering yöntemi ise veri setinin değişimlerini zaman içinde izlemek için kullanılır.

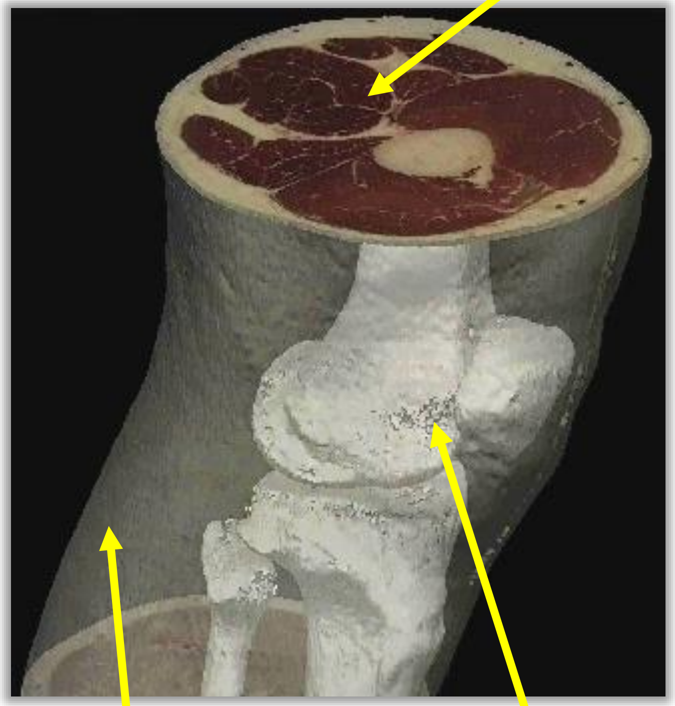
Voxel değerleri nedir ve direct volume renderingde nasıl kullanılır

Voxel, veri setlerinde kullanılan 3 boyutlu piksellere verilen isimdir. Her voxel, veri setinin bir noktasını temsil eder ve bu noktanın koordinatları ile birlikte, bir değer içerir. Bu değer, voxelin belirli bir özelliğini temsil eder. Örneğin, bir CT görüntüsünde voxel değerleri, voxelin belirli bir bölgede bulunan doku yoğunluğunu gösterir.

Direct volume rendering yönteminde, voxel değerlerine göre bir renk ataması yapılır. Bu renk ataması, voxelin özelliğini göstermek için kullanılır. Örneğin, bir CT görüntüsünde, düşük yoğunlukta olan dokuların daha açık renklerle, yüksek yoğunlukta olan dokuların daha koyu renklerle gösterilmesi mümkündür. Bu renk ataması sayesinde, veri setindeki nesnelerin kalınlıkları ve şekilleri belirlenir ve görüntüde görülebilir hale gelir.

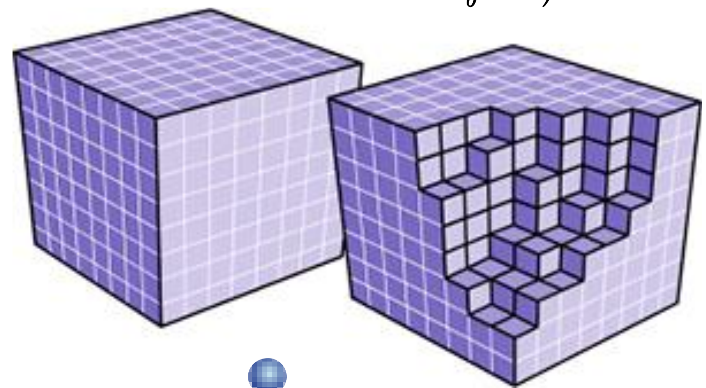
Volume visualization

- Slicing
  Display volume data, mapped to colors, on a slice plane
- Iso-surfacing
  Generate opaque/semi-opaque surfaces (e.g., via the MC-Algorithm)
- Direct Volume Rendering
  Volume material attenuates reflected or emitted light

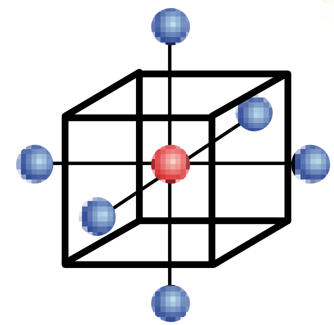


Volume visualization

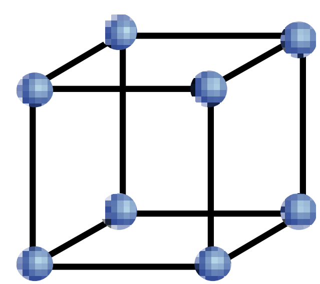
SIEMENS
Ingenuity for life



- Data values are initially given at vertices of a 3D grid
  - These are called voxels (volume elements)
  - Voxel = point sample in 3D



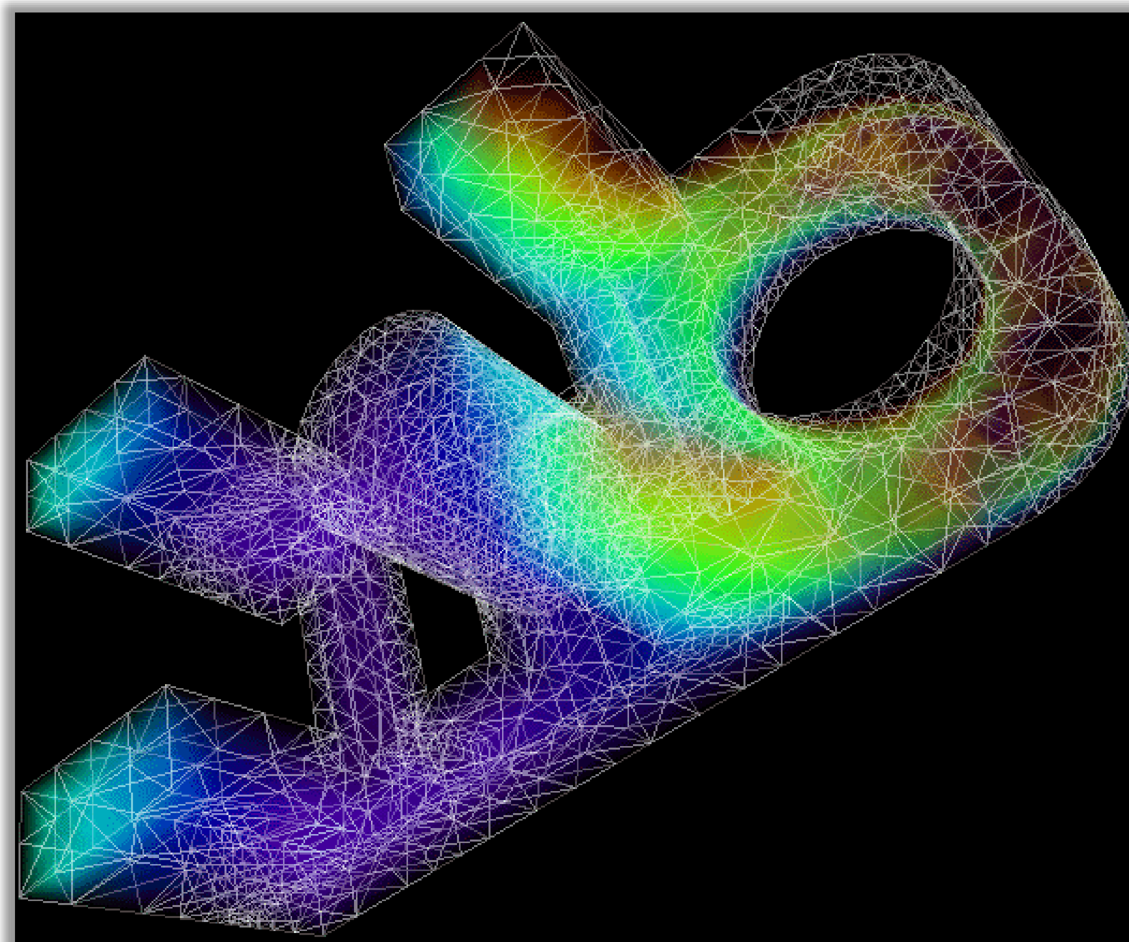
- "Adjacent" grid vertices make up a cell
  - Use interpolation for data reconstruction in a cell
  - Corners: 8 voxel



Volume visualization

SIEMENS
Ingenuity for life

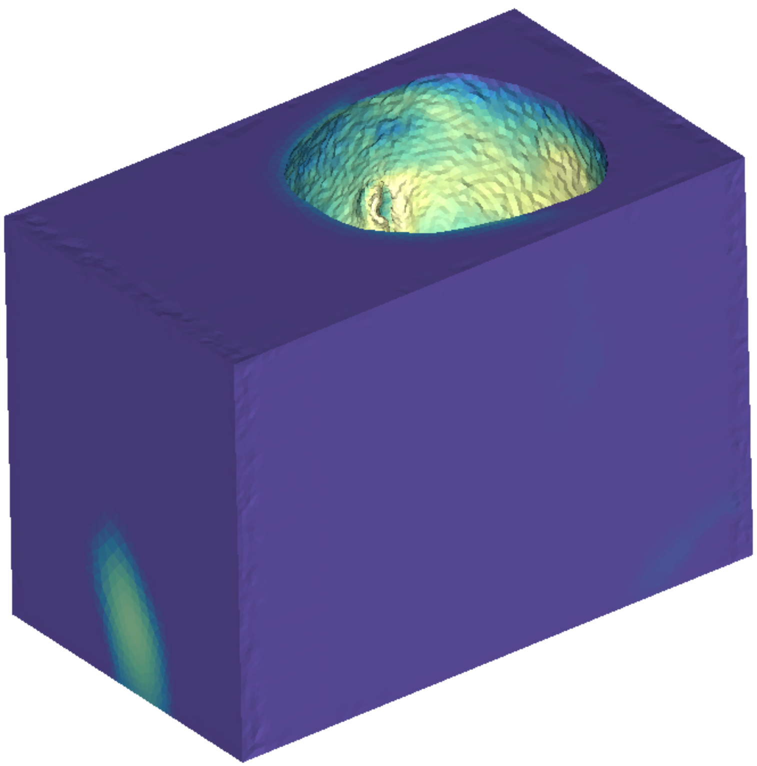
- A volumetric object has…
  - a shape (given by the geometry of the 3D grid)
  - an appearance (given by the data values)

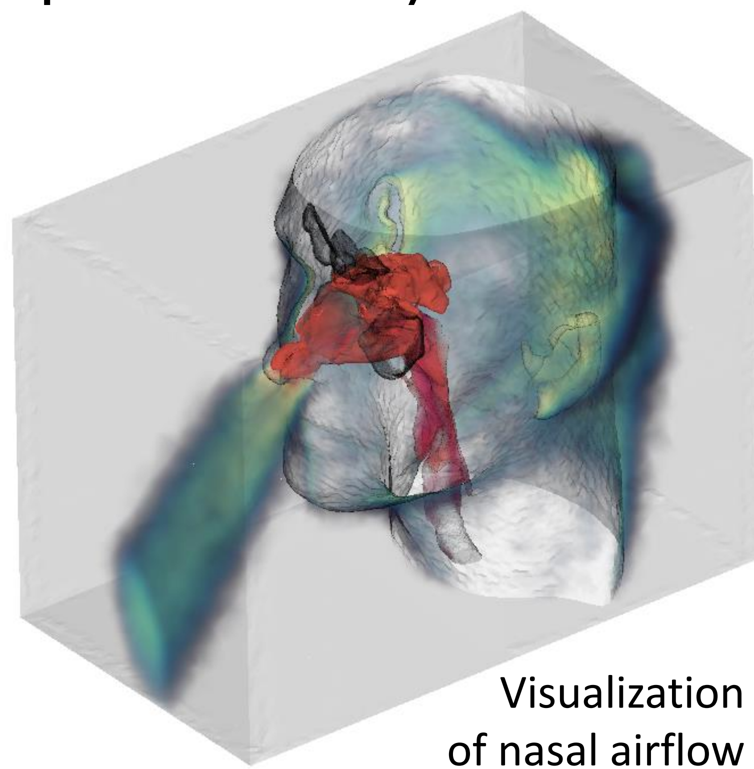


Volume visualization

SIEMENS
Ingenuity for life

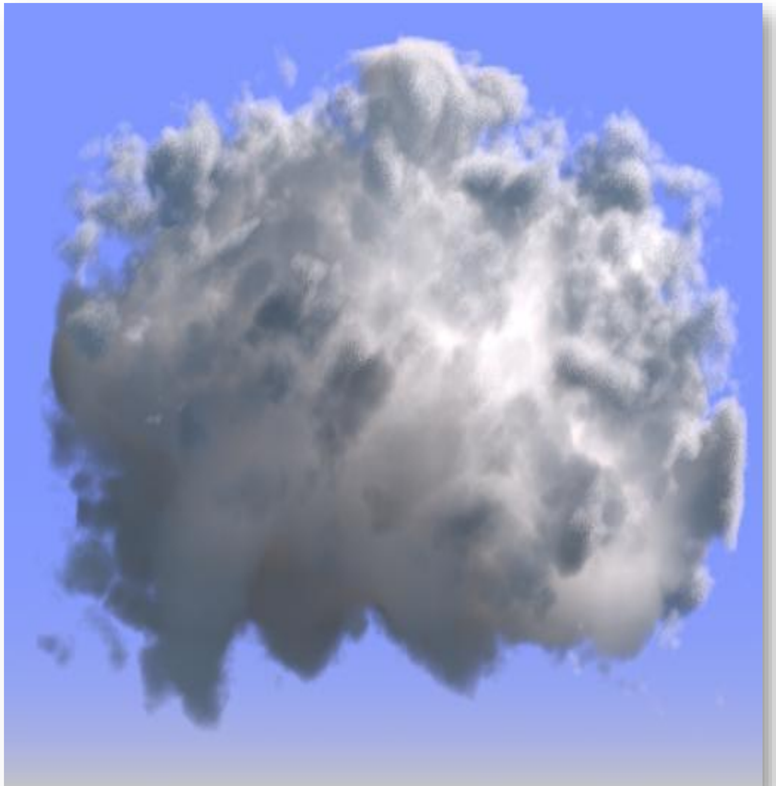
- Some characteristics of volume data
  - Essential information in the interior
  - Often cannot be described by a surface in general (e.g., fire, clouds, gaseous phenomena)



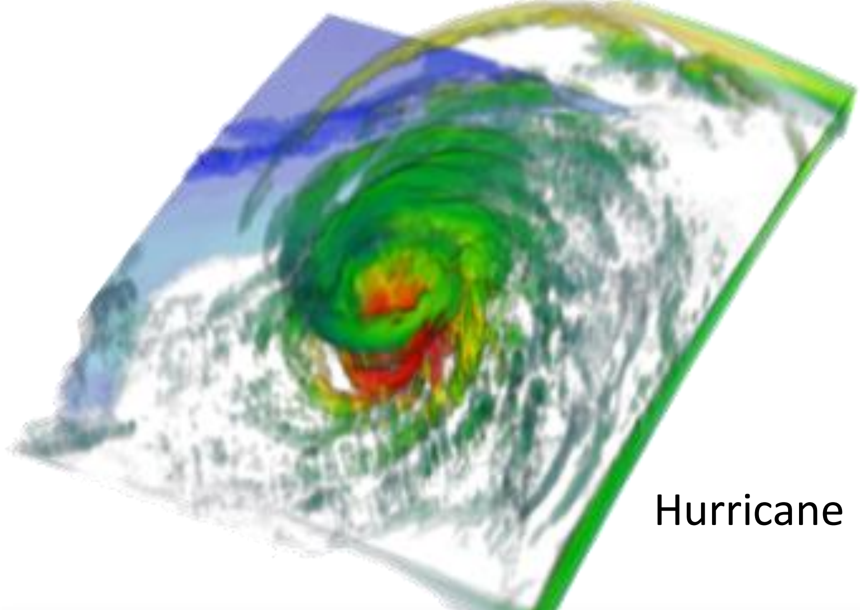


Visualization
of nasal airflow

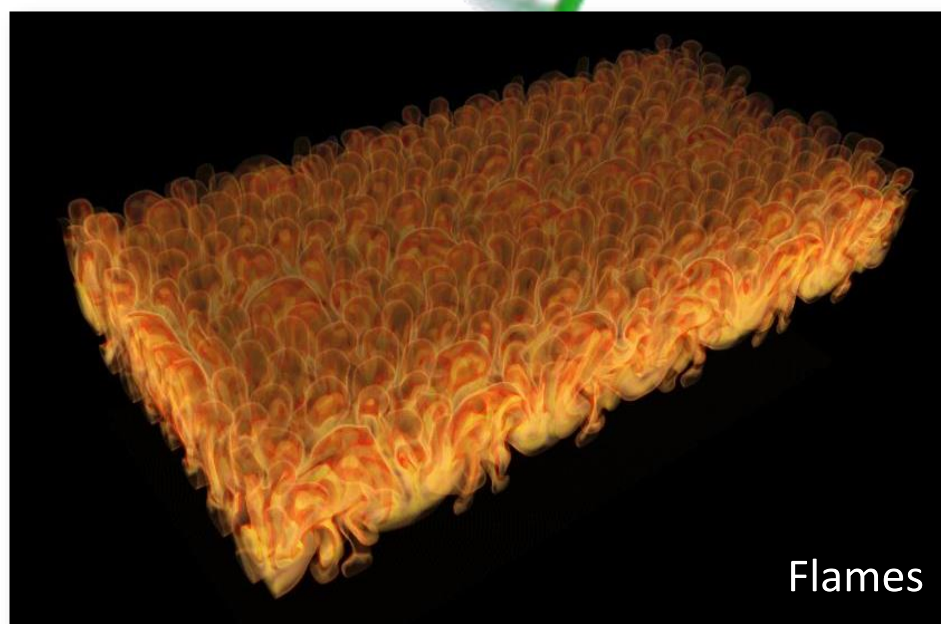
Volume visualization



Clouds

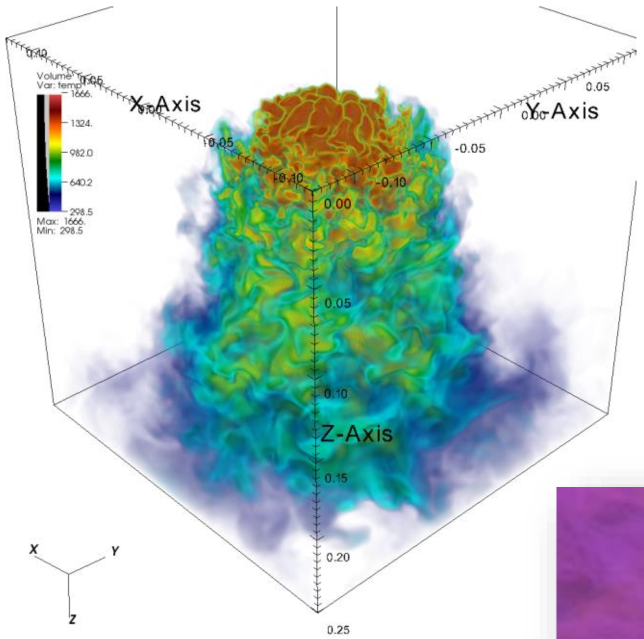


Hurricane

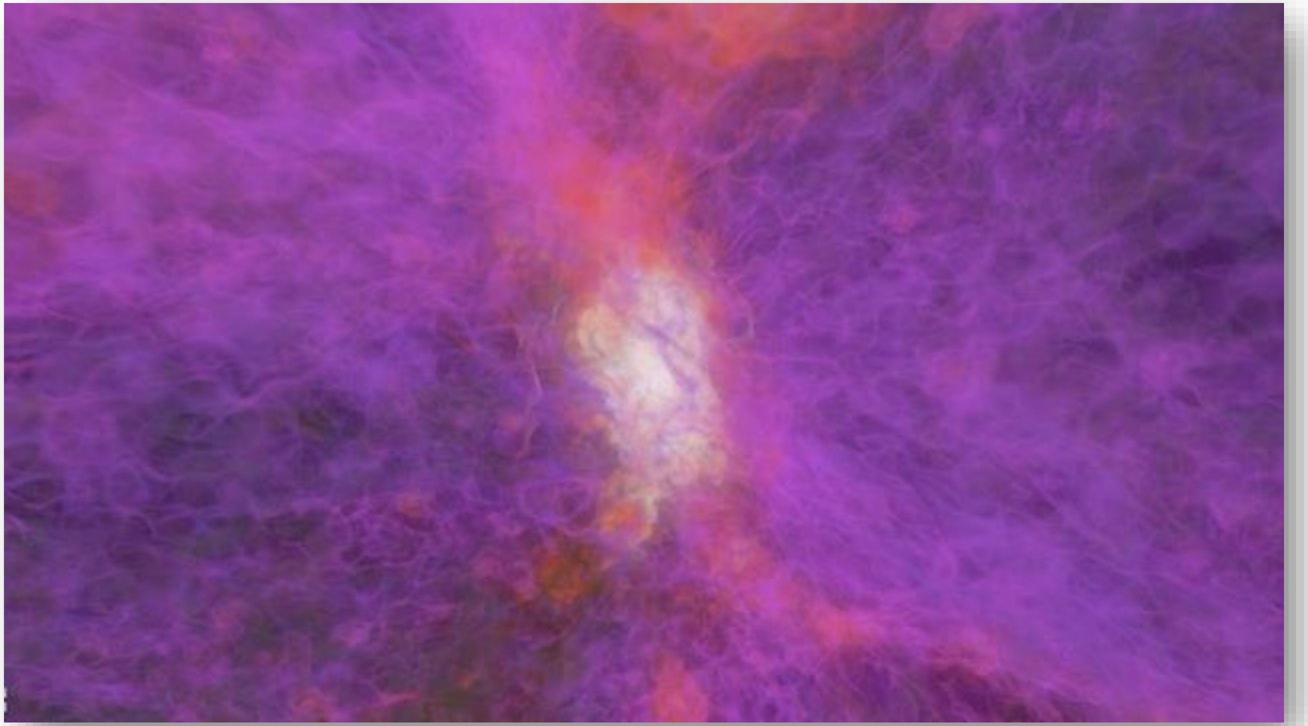


Volume visualization

SIEMENS
Ingenuity for life

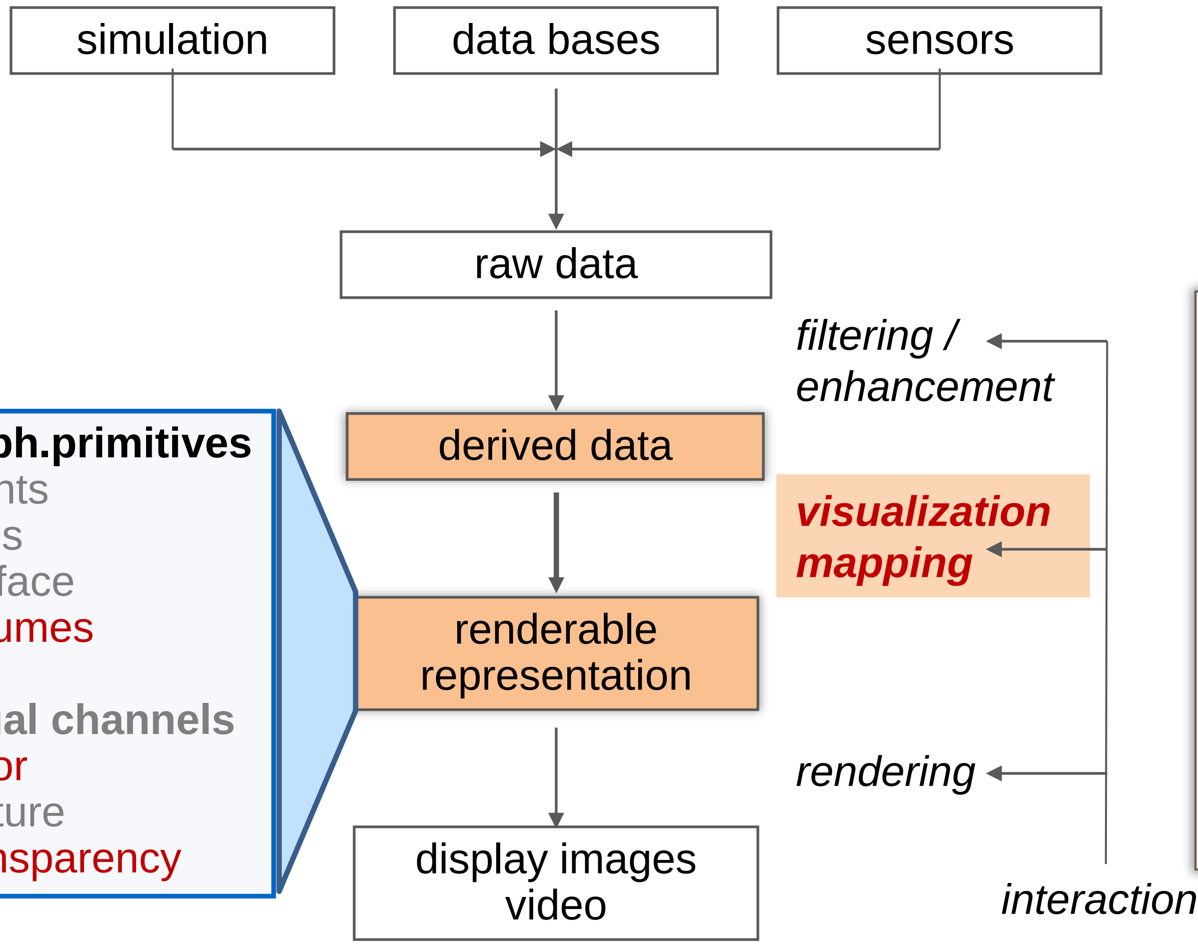


Hydrogen flame



Millennium Simulation: Distribution of matter in the Universe

Volume Visualization



Graph.primitives
- points
- lines
- surface
- volumes

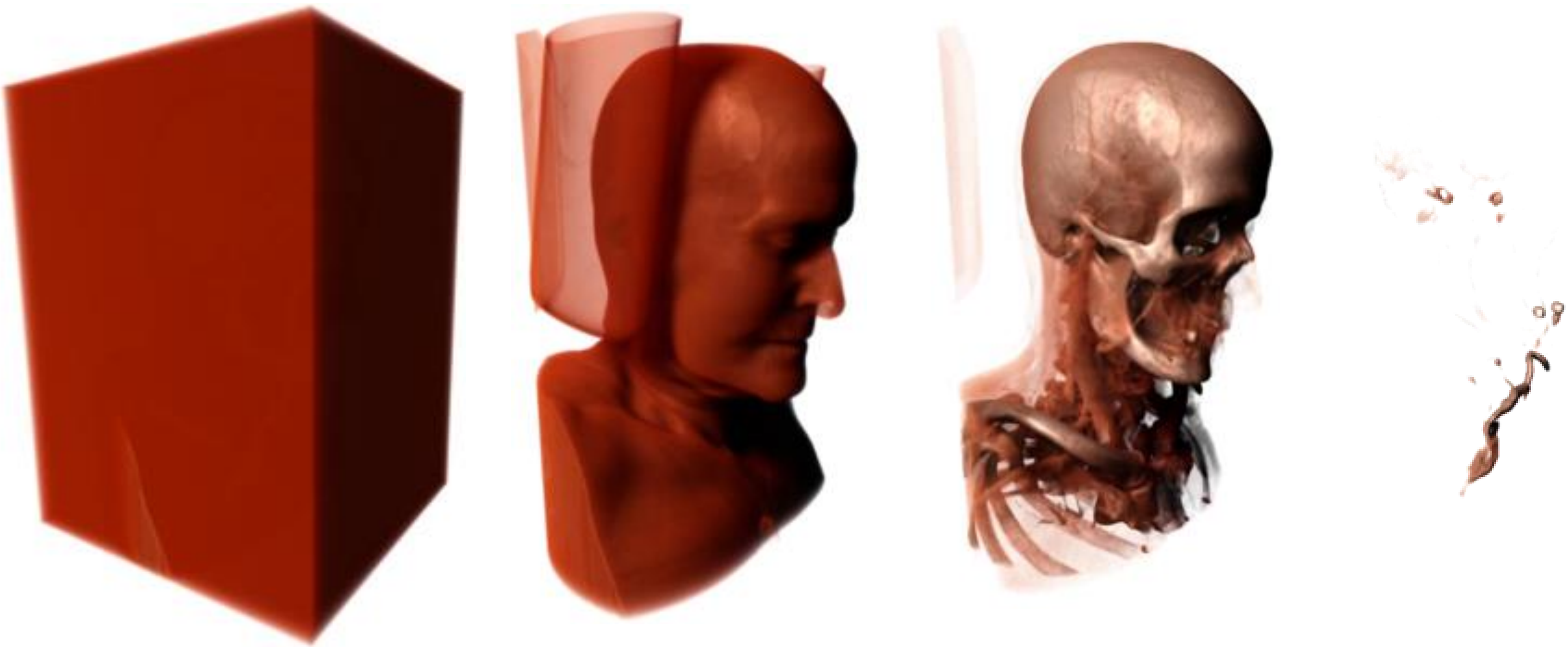
Visual channels
- color
- texture
- transparency

Derived data
→ graphical primitives
- scalar field → isosurface
- 2D field → height field
- vector field → vectors
- tensor field → glyphs
- 3D field → volume (C, α)

Decide
- Which parts of the data are shown?
- How to represent them?

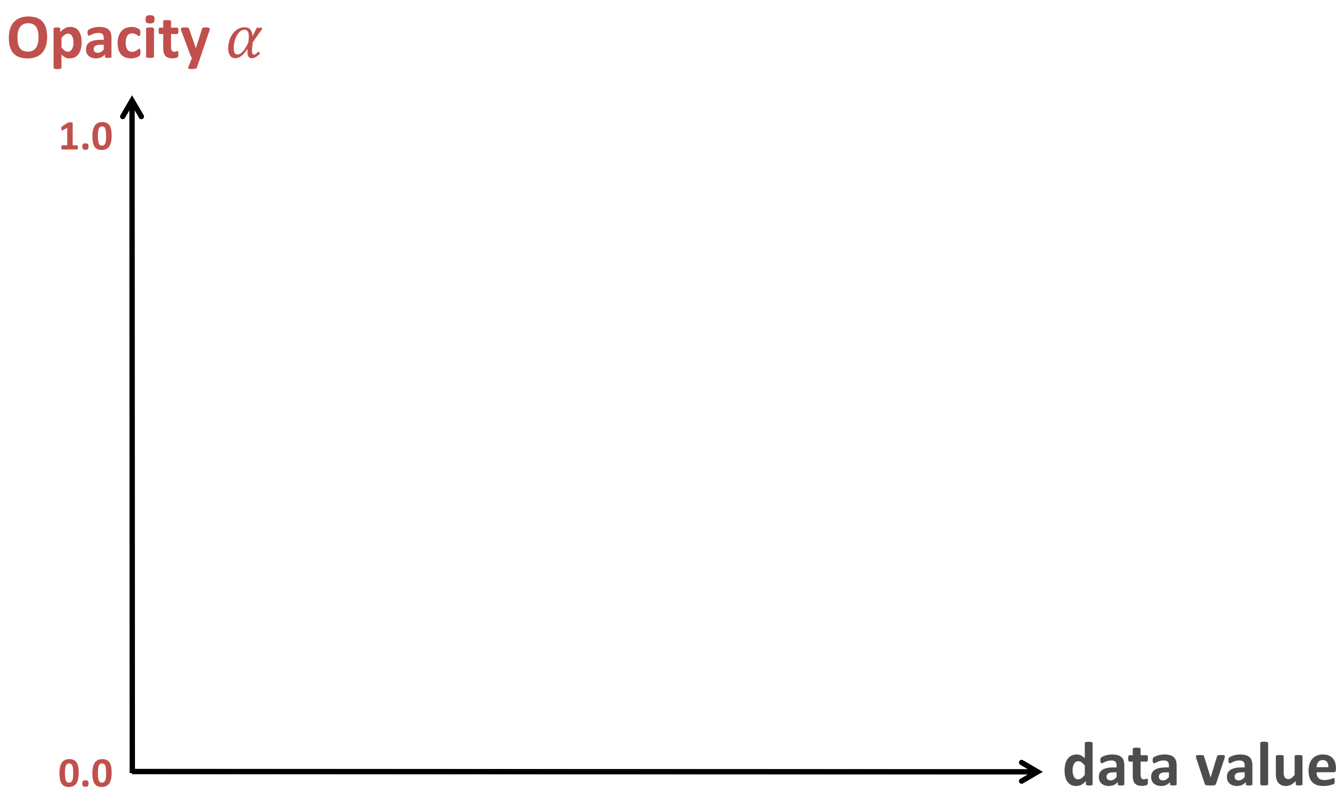
Volume visualization

- Before data is visualized, it has to be classified, i.e., data values are mapped to visual properties like color and opacity
- Mapping is performed via a transfer function



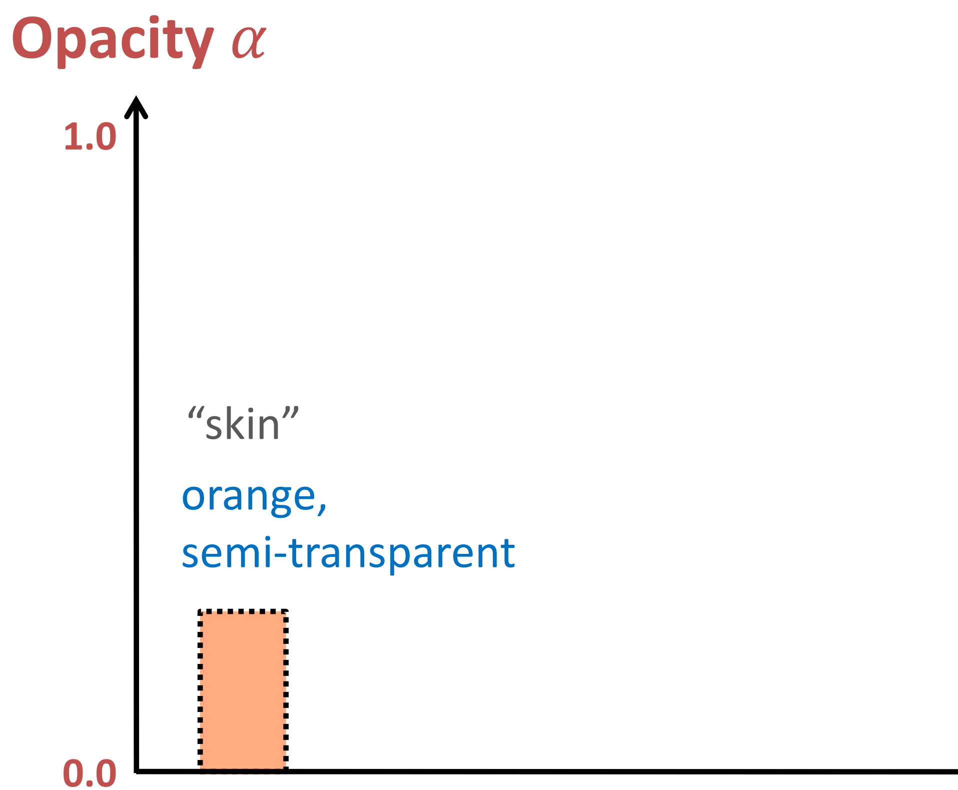
Volume visualization

- Transfer function
  - Associate distinct materials (value ranges) to distinct properties (color & opacity)



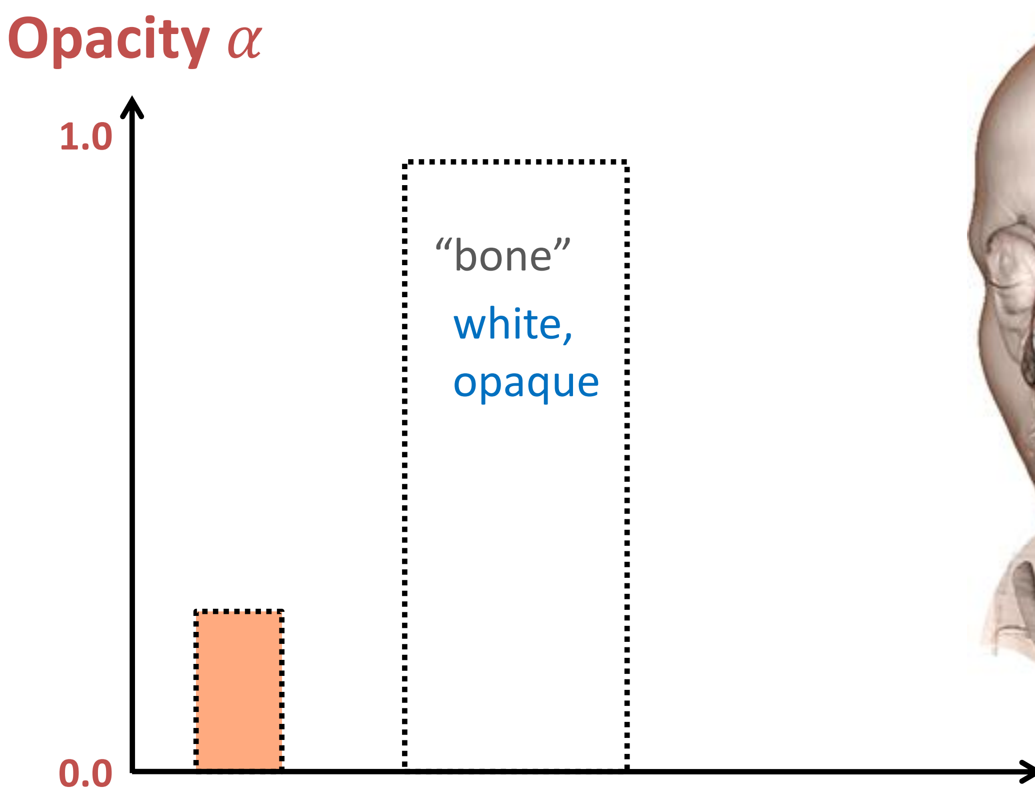
Volume visualization

- Transfer function
  - Data value → color: $f(\boldsymbol{x}) \rightarrow C(\boldsymbol{x})$
  - Data value → opacity: $f(\boldsymbol{x}) \rightarrow \alpha(\boldsymbol{x})$



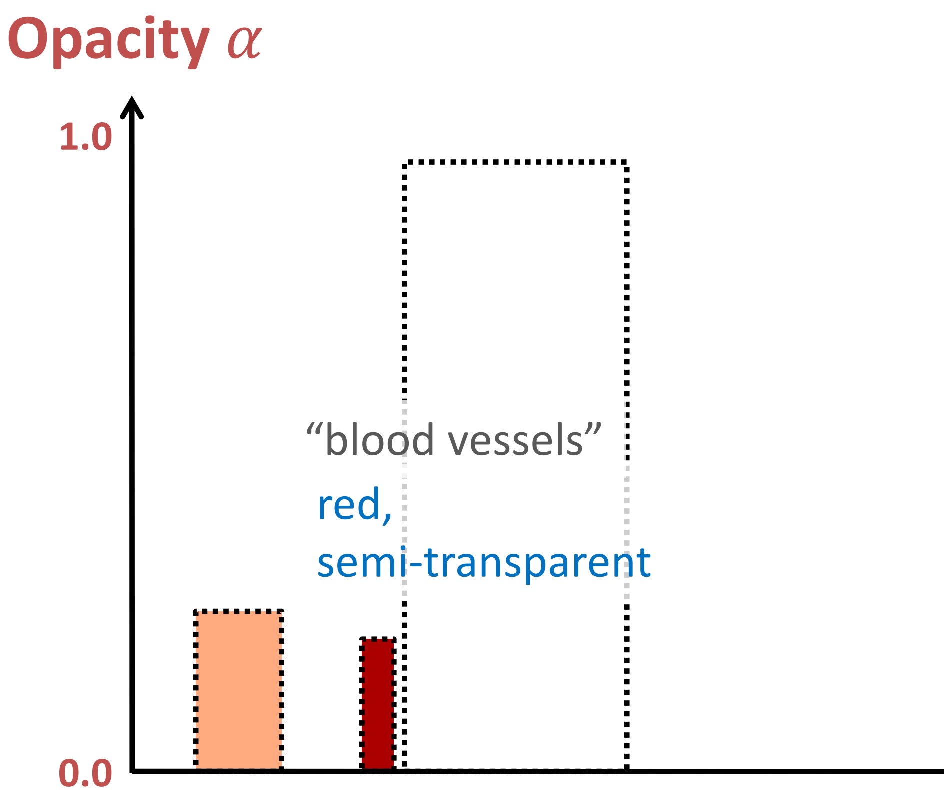
Volume visualization

- Transfer function
  - Data value → color: $f(\boldsymbol{x}) \rightarrow C(\boldsymbol{x})$
  - Data value → opacity: $f(\boldsymbol{x}) \rightarrow \alpha(\boldsymbol{x})$



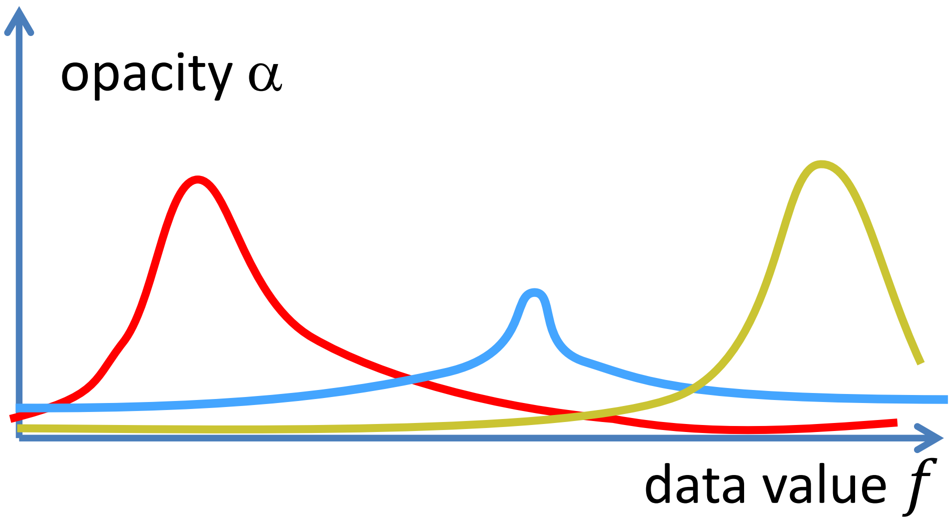
Volume visualization

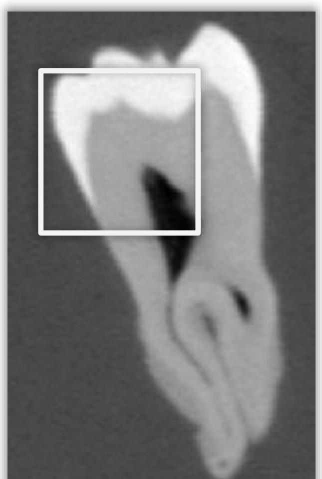
- Transfer function
  - Data value → color: $f(\boldsymbol{x}) \rightarrow C(\boldsymbol{x})$
  - Data value → opacity: $f(\boldsymbol{x}) \rightarrow \alpha(\boldsymbol{x})$



Volume visualization

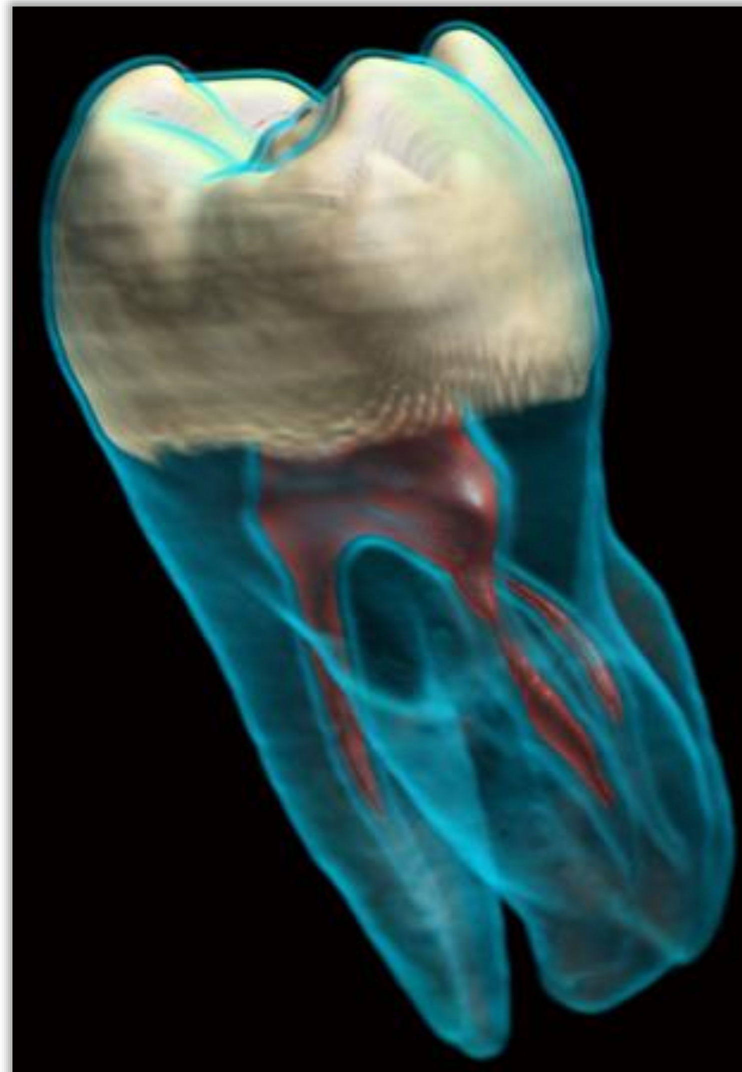
- Transfer function design





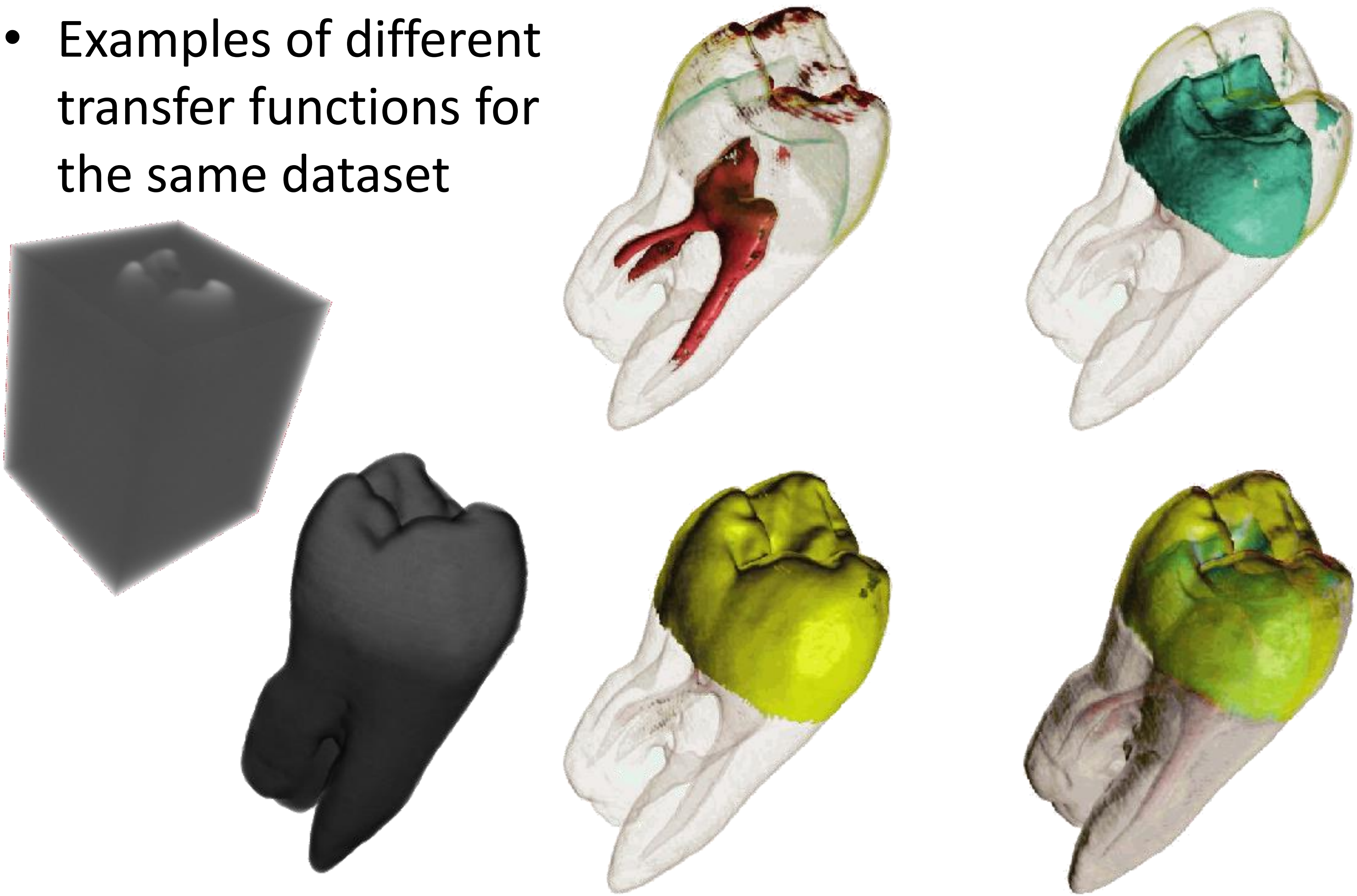
color

opacity

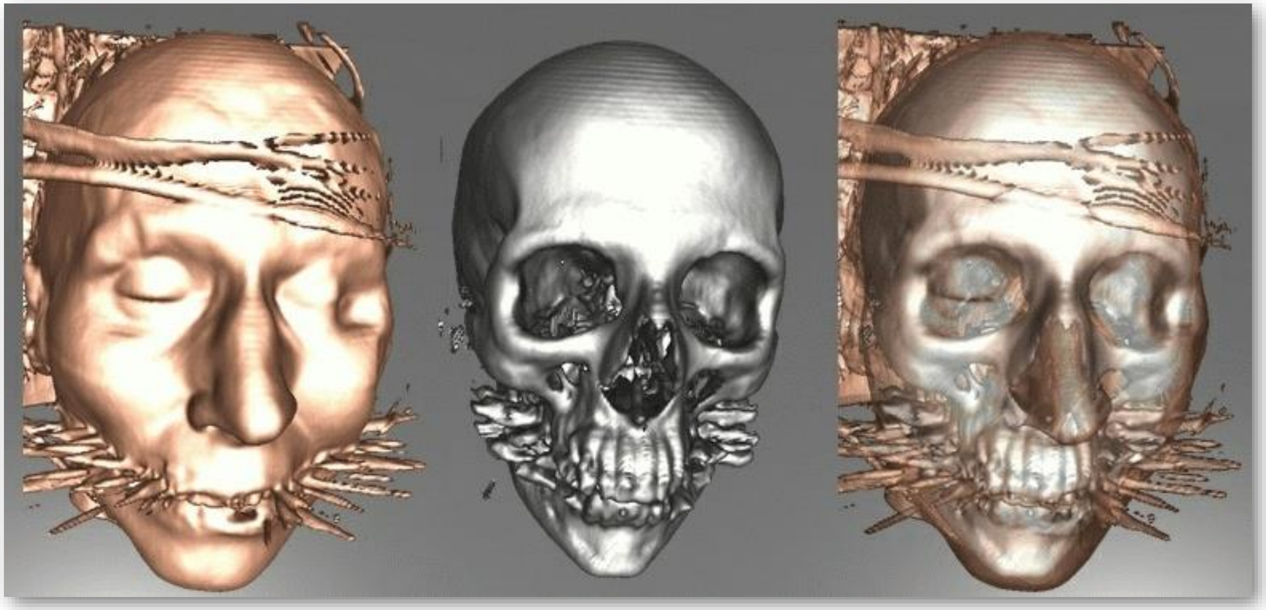


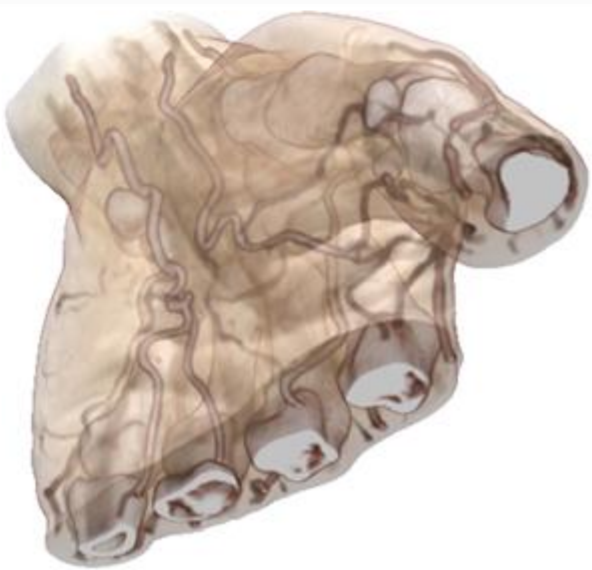
Volume visualization

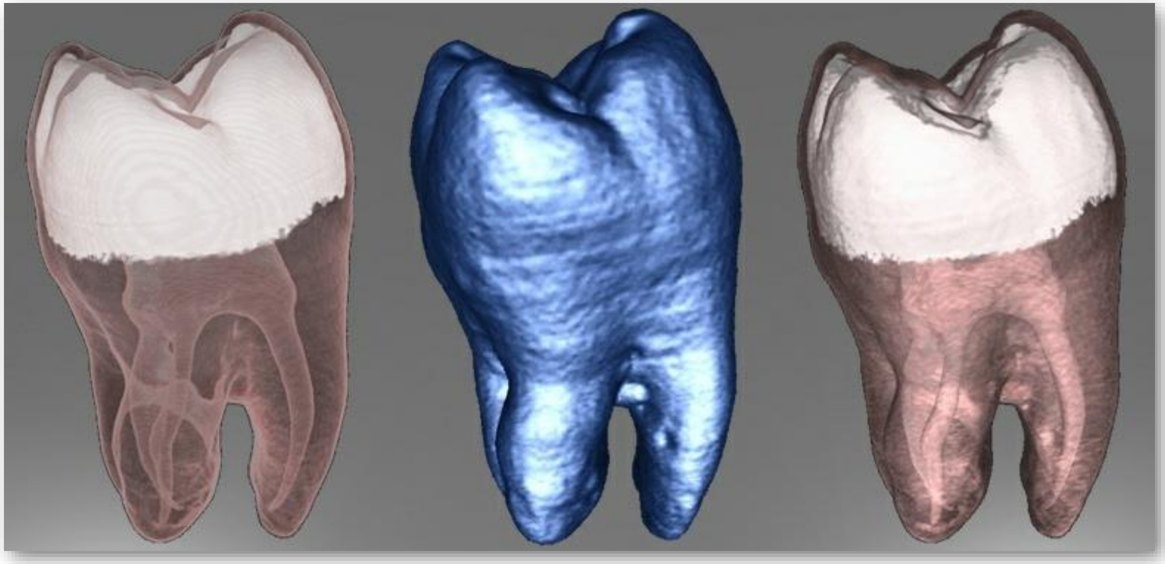
- Examples of different transfer functions for the same dataset



Volume visualization





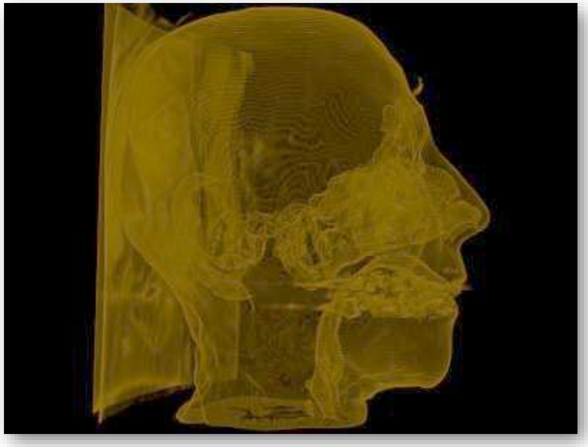
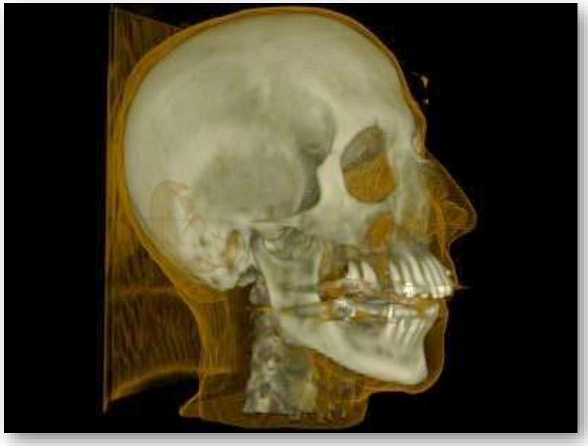


[Engel et al. 06, Krüger & Westermann 03]

Volume visualization

SIEMENS
Ingenuity for life

- Note that color is a material property
- Note that opacity α is also such a property and can be assigned as well!
- Color is then a quadrupel (RGBα)

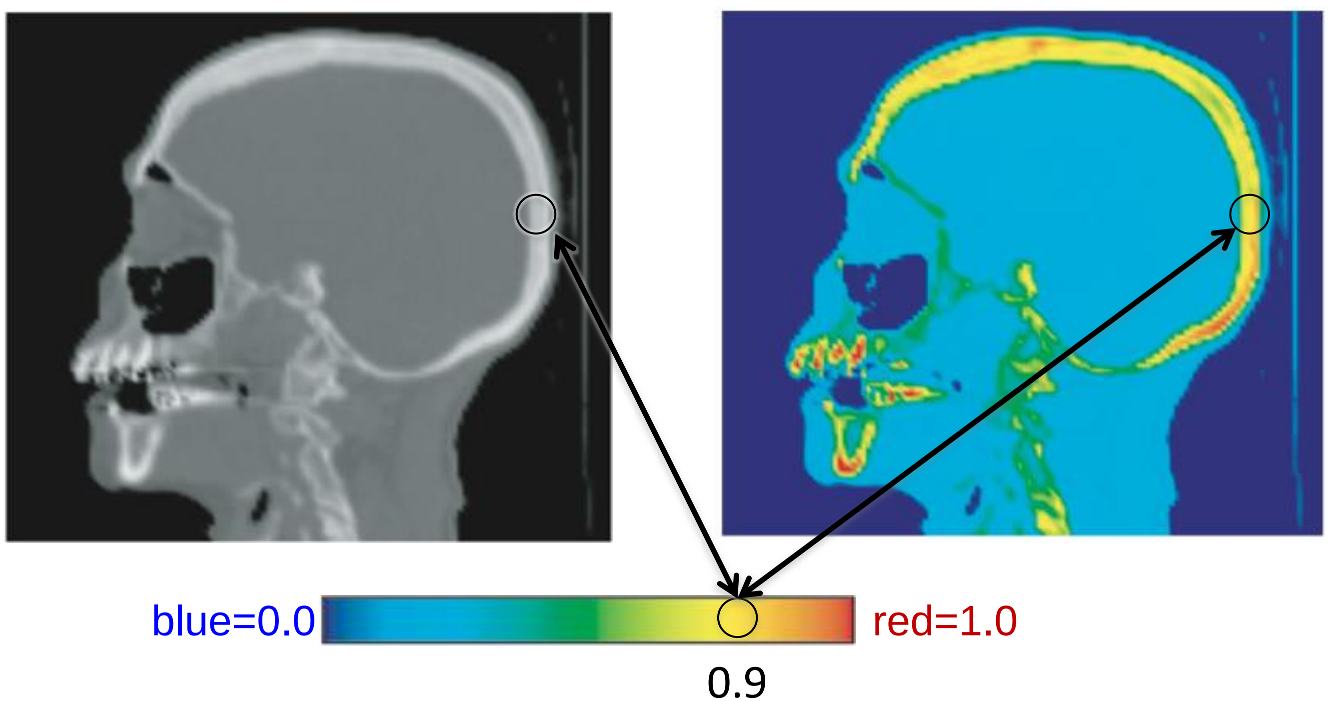




Volume visualization

SIEMENS
Ingenuity for life

- Color mapping for scalar data
  - Assign a different color to each scalar value
  - Assignment via a so-called **transfer function** *T*

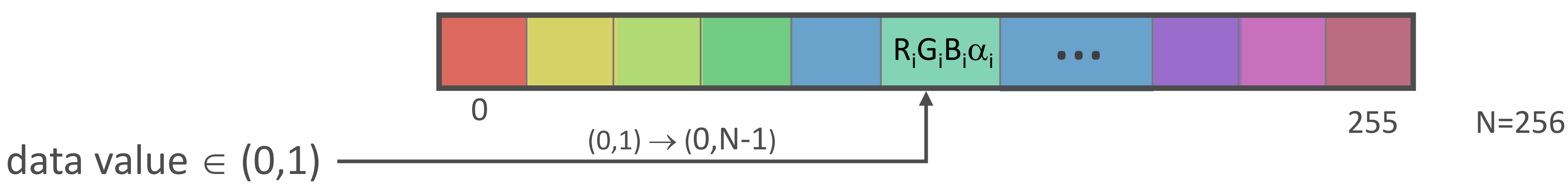
    T : scalar value → color + α-value



Volume visualization

- Color mapping for scalar data
  - Assign a different color to each scalar value
  - Assignment via a so-called **transfer function** *T*

    T : scalar value → color + α-value
  - Realization
    - RGBA-color values are stored in a color table
    - For a data value, the color at the corresponding entry in the table is used



Direct volume rendering

- Get a 3D representation of the volume data taking into account emission and absorption

Direct volume rendering

SIEMENS
Ingenuity for life

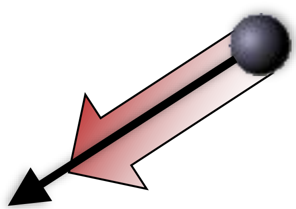
- Direct volume rendering (DVR) considers the physics of light transport in a dense medium
  - Optical properties are mapped to each voxel (emission = color, absorption = opacity)
  - The light reaching the viewer is simulated by ray casting

No need to extract intermediate explicit geometry (iso-surface)

Direct volume rendering

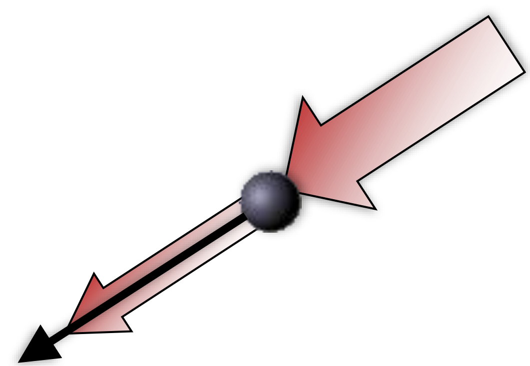
- Direct volume rendering
  - The collection of light along straight rays is based on an emission/absorption model
  - Assumption: Volume consists of small particles
  - Each voxel emits light of the color which is assigned to it, and absorbs light according to it's opacity

Emission



Energy increase

Absorption

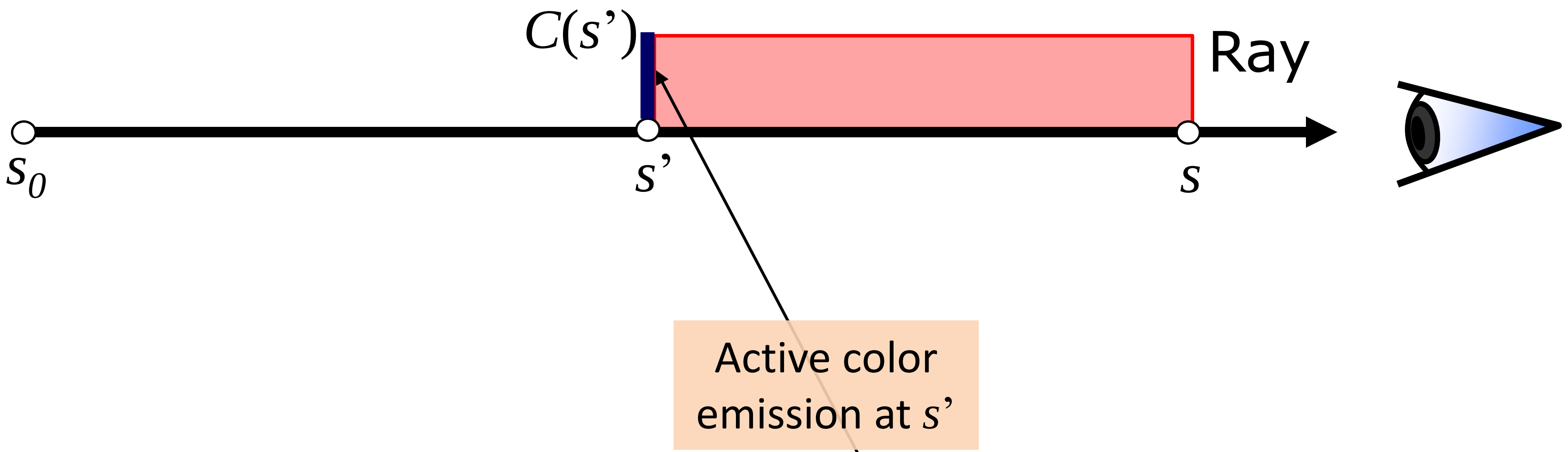


Energy decrease

Light emission and attenuation

SIEMENS
Ingenuity for life

- The volume rendering integral

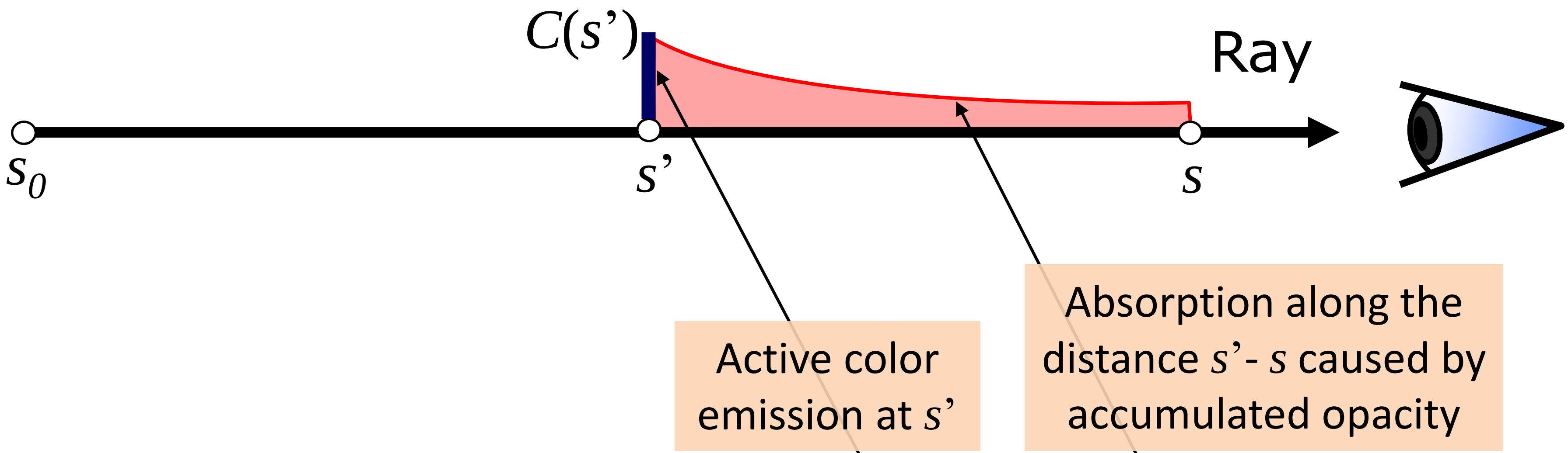


$$C(s) = \boxed{C(s')}$$

Without absorption the color emitted at point s' would reach point s

Light emission and attenuation

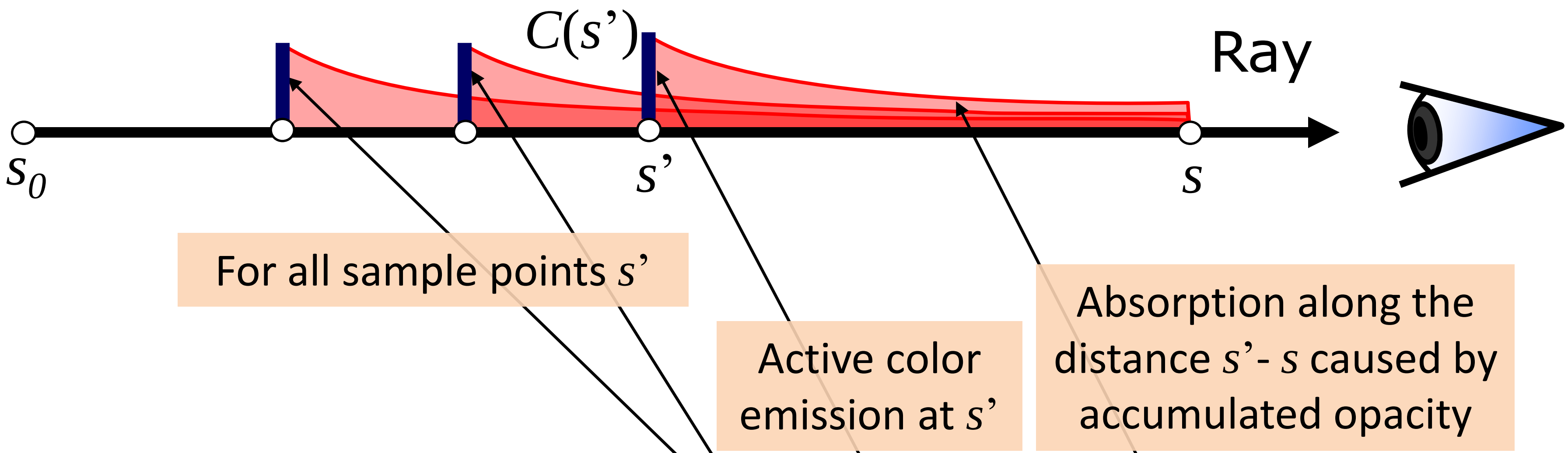
- The volume rendering integral



$$C(s) = \boxed{C(s')} \cdot \boxed{e^{-\int_{s'}^{s} \alpha(t)\, dt}}$$

Light emission and attenuation

- The volume rendering integral

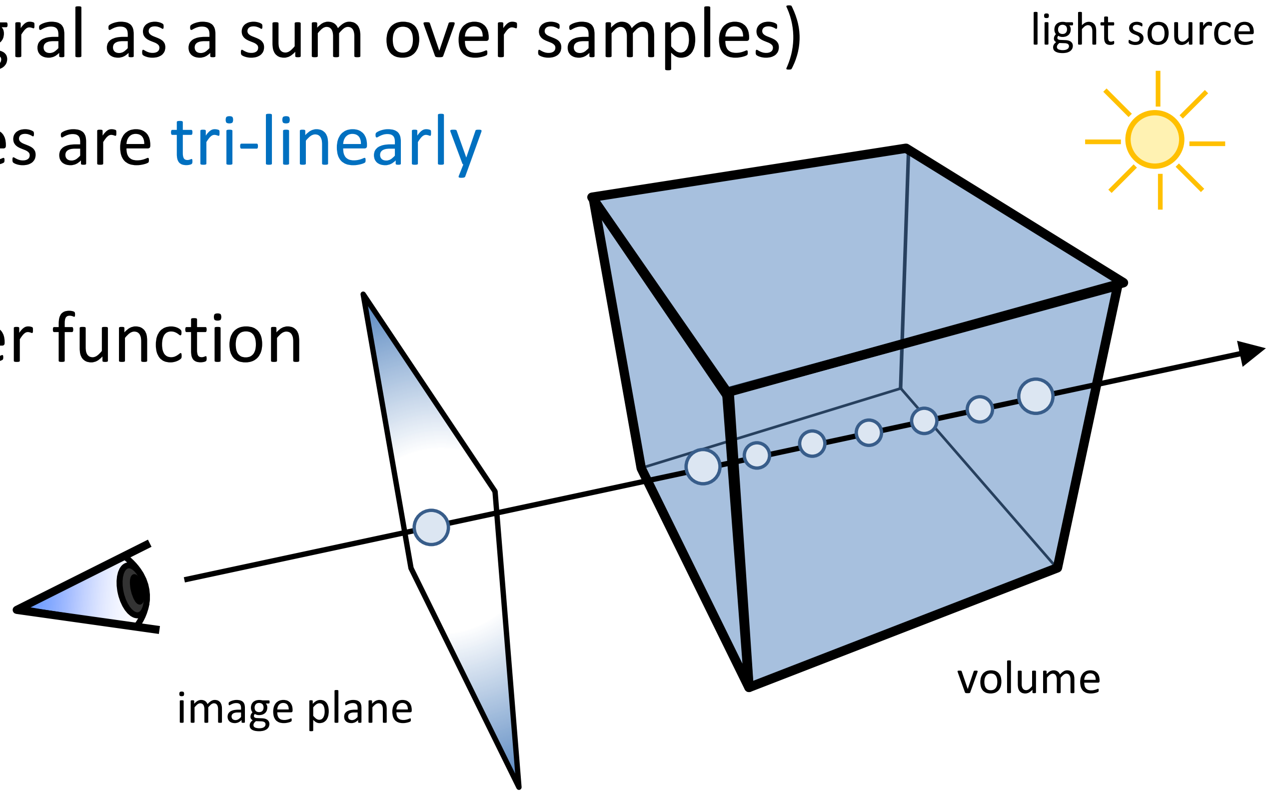


$$C(s) = \int_{s_0}^{s} \boxed{C(s')} \cdot \boxed{e^{-\int_{s'}^{s} \alpha(t)\, dt}}\, ds'$$

Direct volume rendering

SIEMENS
Ingenuity for life

- **Ray-casting:** Numerical approximation of the volume rendering integral
  - A ray is cast into volume for each output pixel
  - Volume is resampled at equidistant intervals along the ray (integral as a sum over samples)
  - Sample values are tri-linearly interpolated
  - Apply transfer function



Direct volume rendering

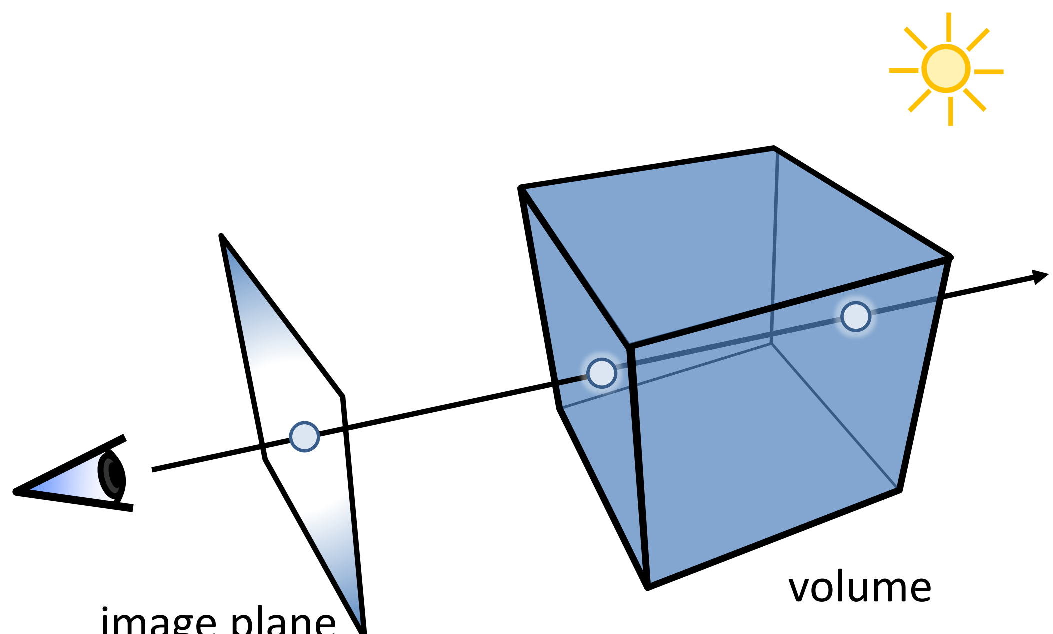
SIEMENS
Ingenuity for life

- Ray-casting method
  - Defines a virtual image plane where viewer is looking through
  - Cast a ray through every pixel on the screen

for each pixel on the image plane

compute entry- and exit-point in volume

end for



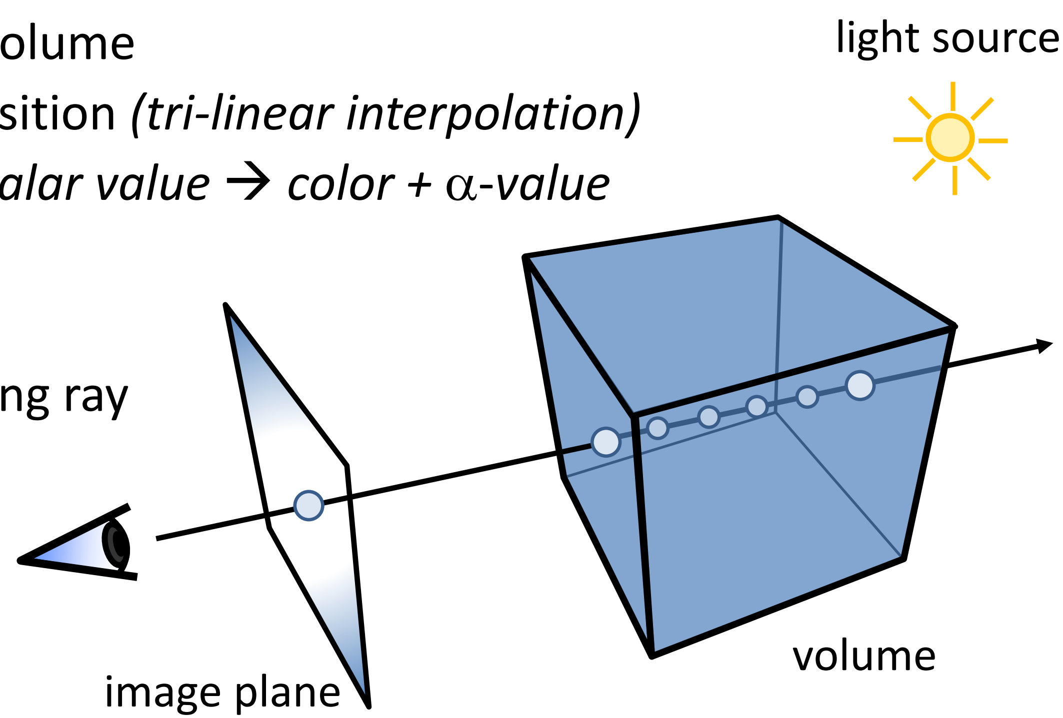
Direct volume rendering

SIEMENS
Ingenuity for life

- Ray-casting method
  - Defines a virtual image plane where viewer is looking through
  - Cast a ray through every pixel on the screen

for each pixel on the image plane
  compute entry- and exit-point in volume
  while current position inside volume
    read density at current position *(tri-linear interpolation)*
    apply transfer function: *scalar value → color + α-value*

    compute new position along ray
  end while
end for

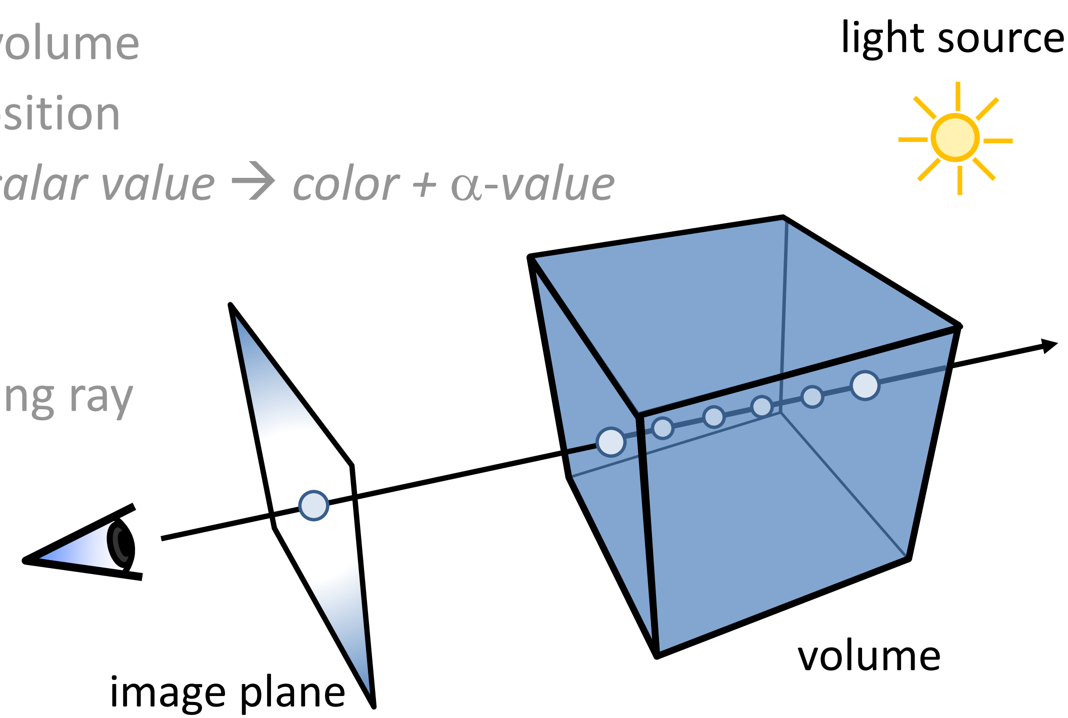


Direct volume rendering

SIEMENS
Ingenuity for life

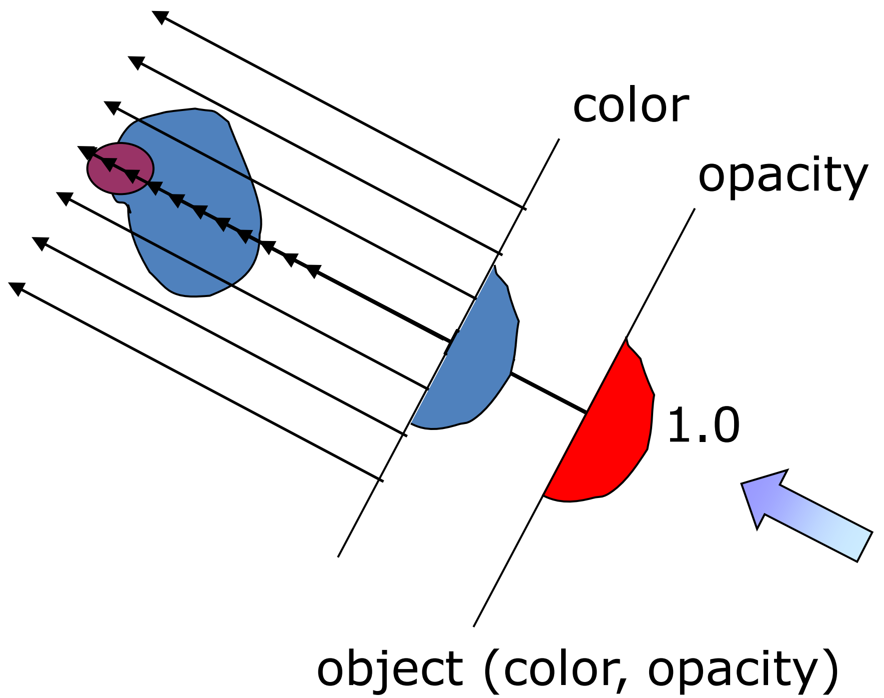
- Ray-casting method
  - Defines a virtual image plane where viewer is looking through
  - Cast a ray through every pixel on the screen

```
for each pixel on the image plane
    compute entry- and exit-point in volume
    while current position inside volume
        read density at current position
        apply transfer function: scalar value → color + α-value
        compute shading
        apply compositing
        compute new position along ray
    end while
    set pixel color in image plane
end for
```

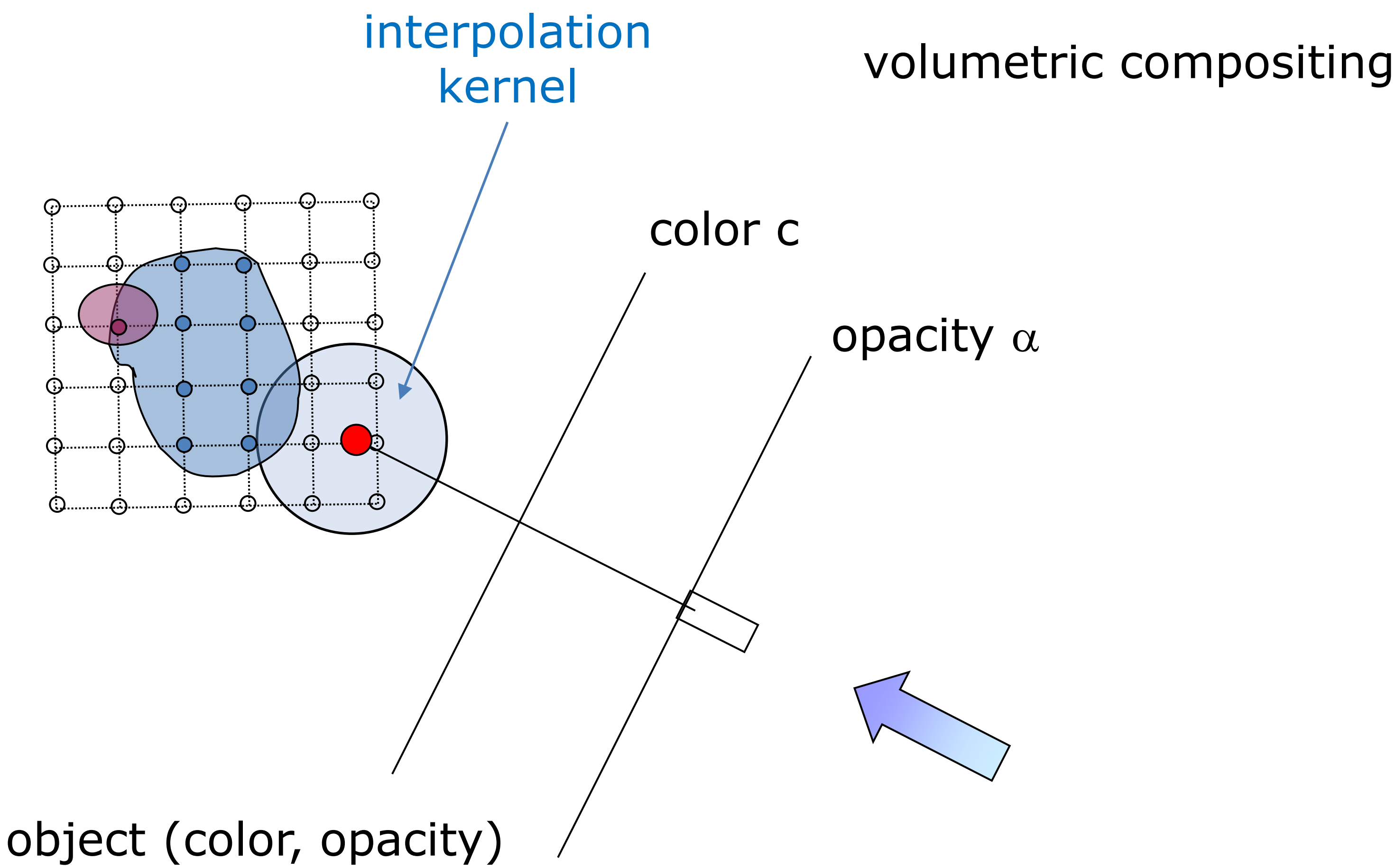


Ray casting

- Volumetric ray integration:
  - Compositing: accumulation of color & opacity along rays



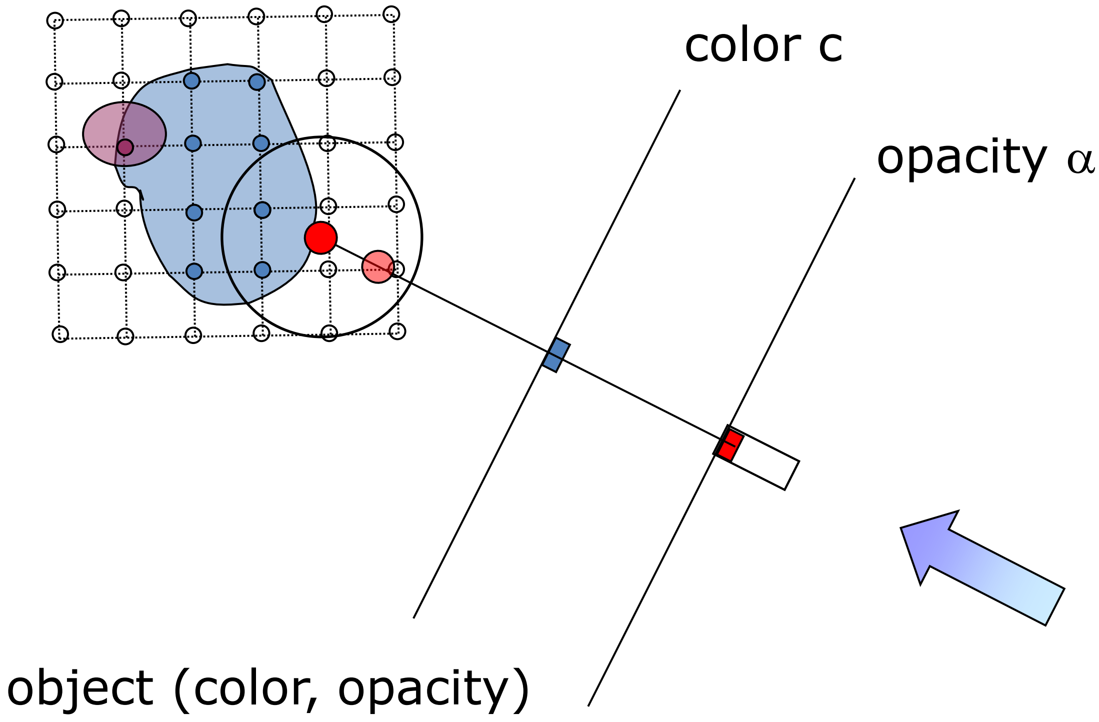
Ray casting



interpolation
kernel
volumetric compositing
color c
opacity α
object (color, opacity)

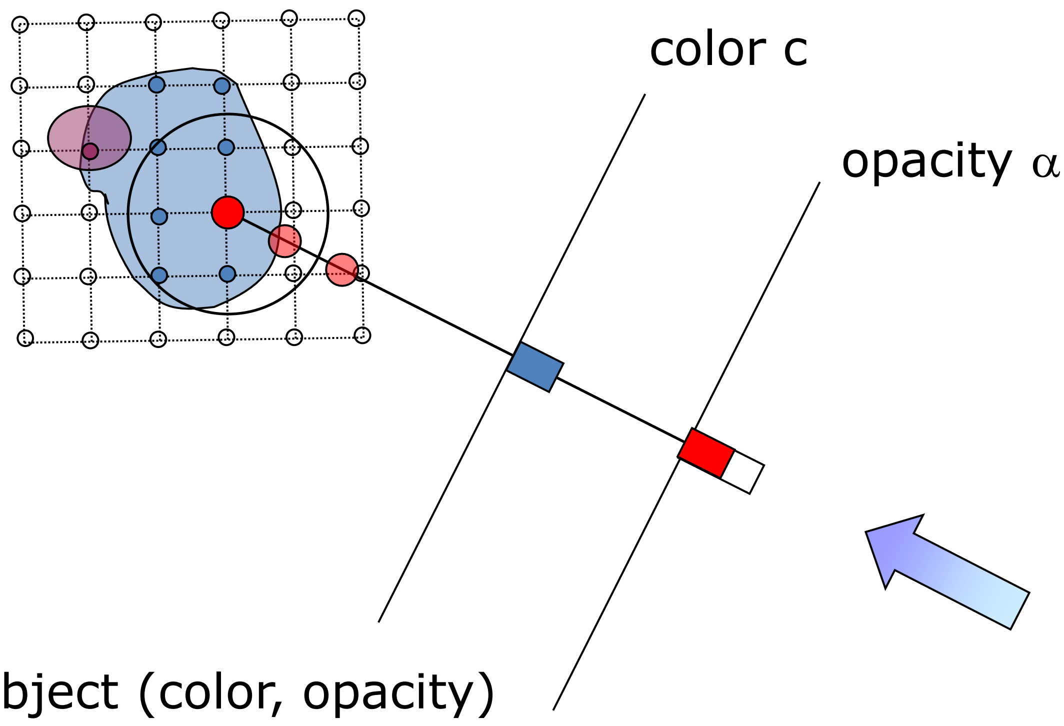
Ray casting

volumetric compositing

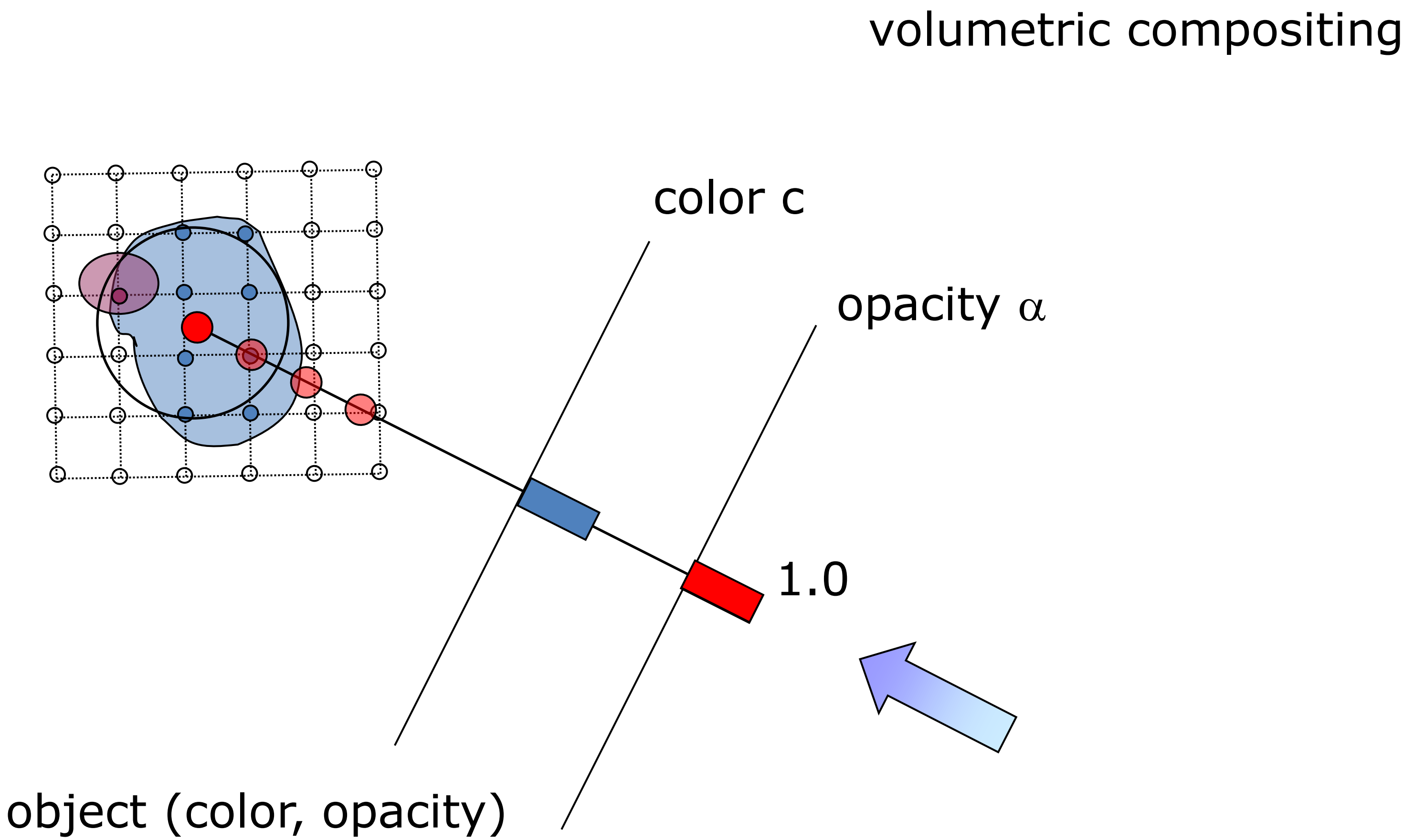


Ray casting

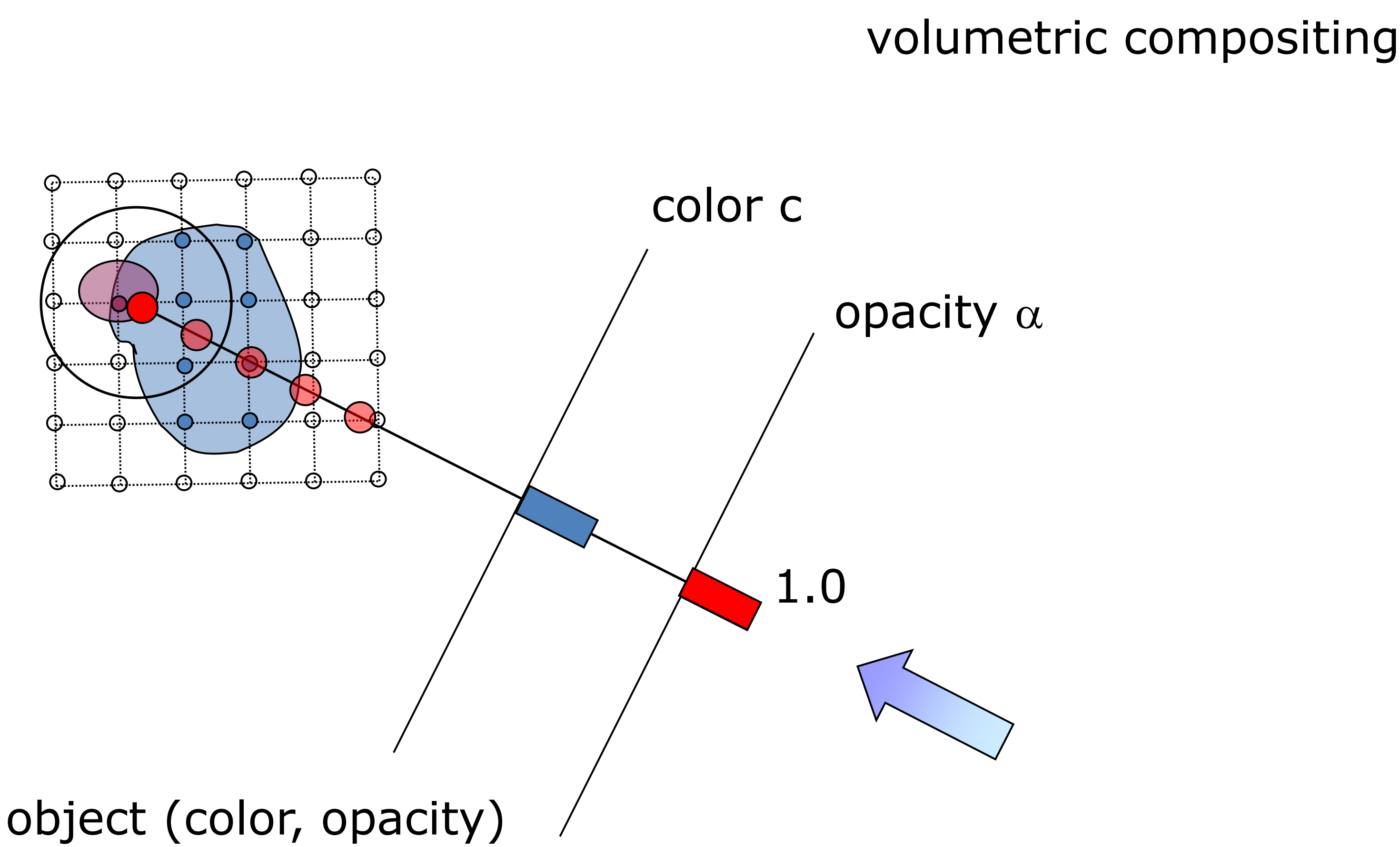
volumetric compositing



Ray casting


volumetric compositing
color c
opacity α
1.0
object (color, opacity)

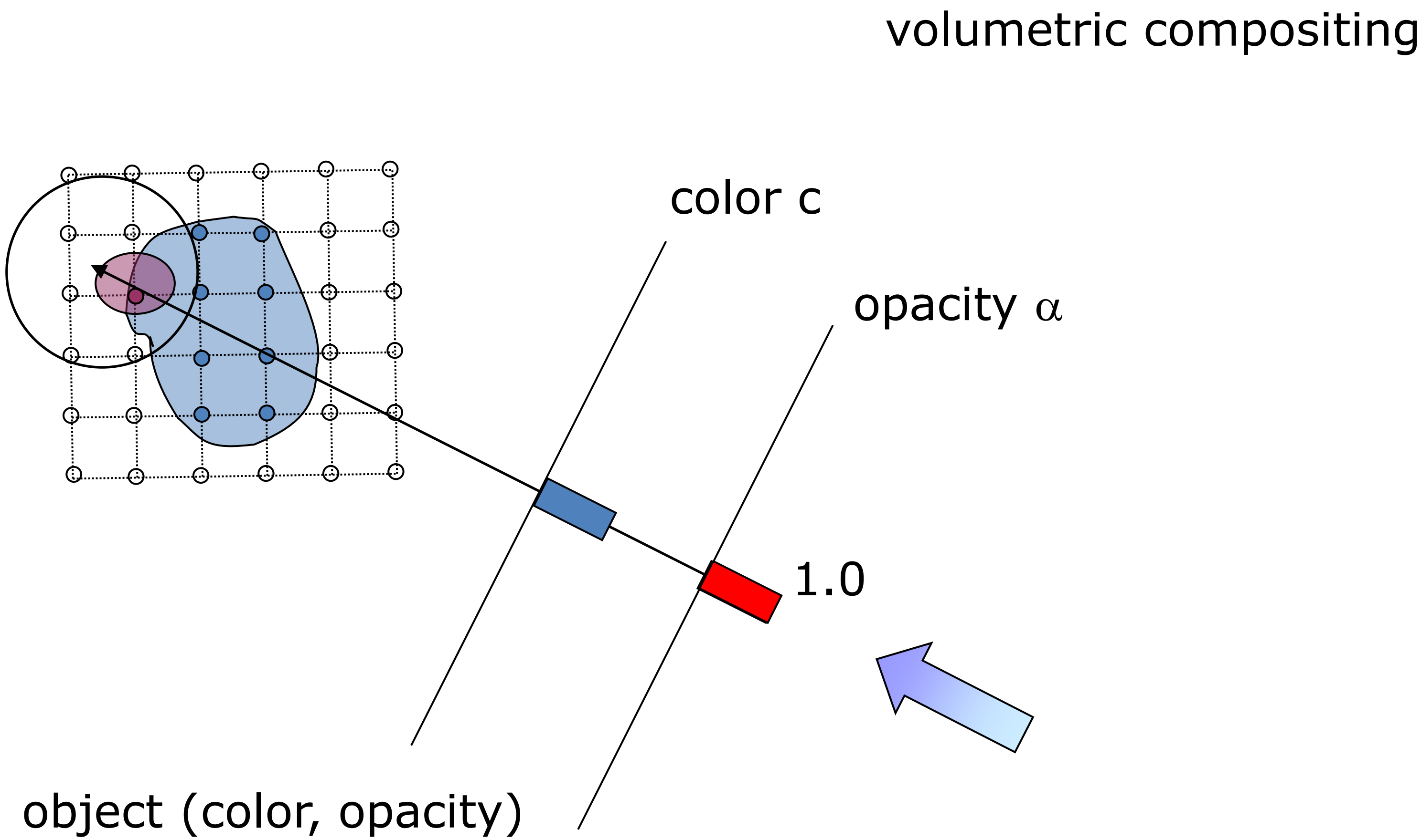
Ray casting



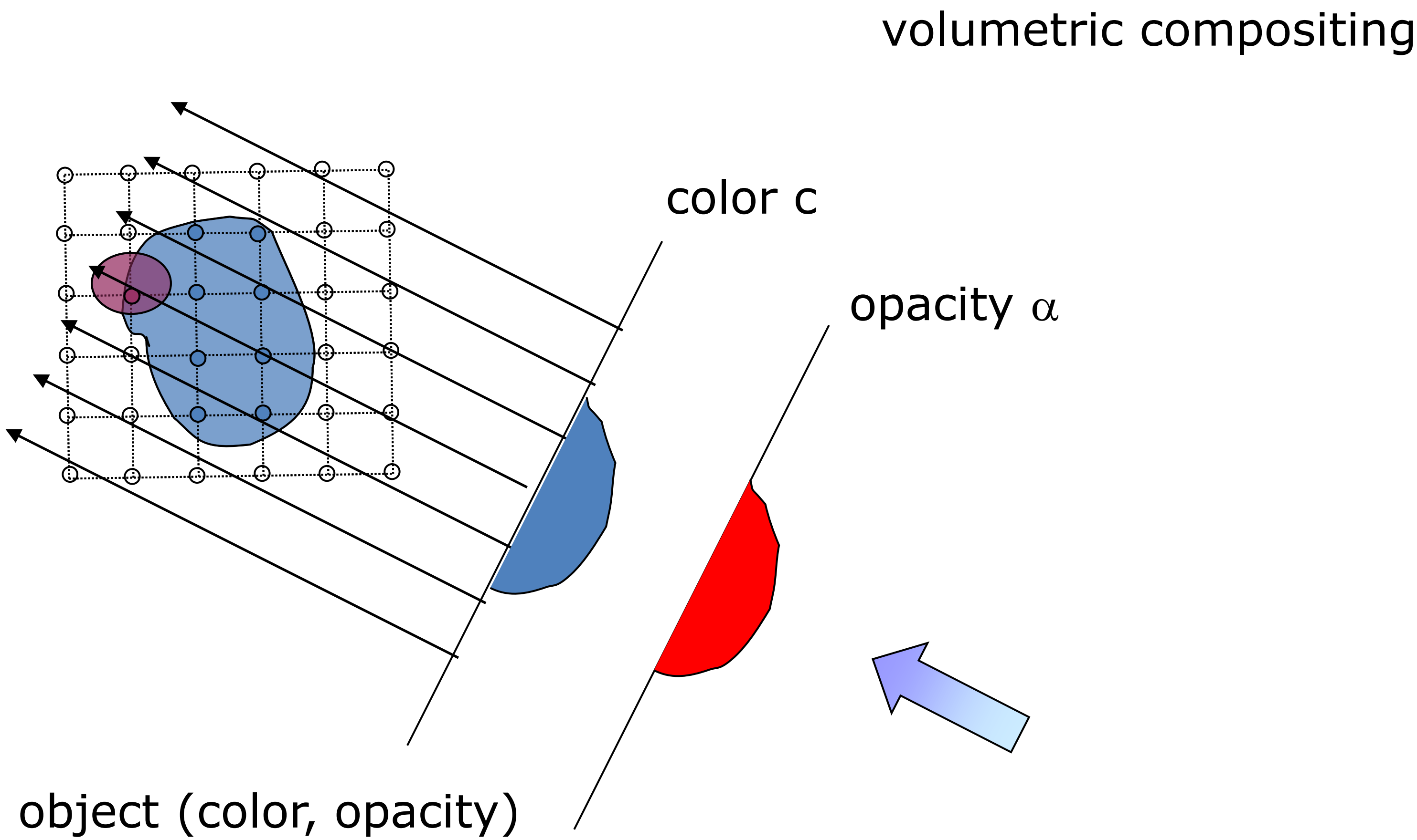
volumetric compositing
color c
opacity α
1.0
object (color, opacity)

Ray casting

SIEMENS
Ingenuity for life



Ray casting

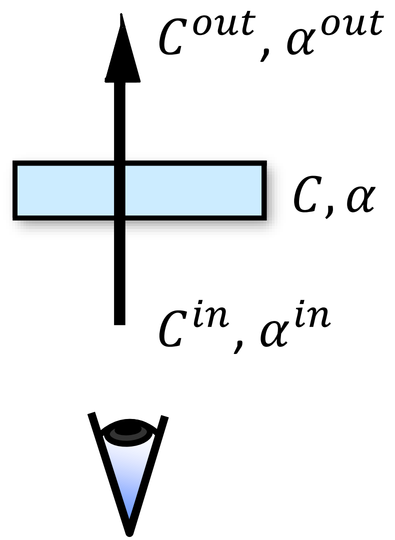


volumetric compositing
color c
opacity α
object (color, opacity)

Ray casting

- Compositing of semi-transparent voxels
  - Physical model: emissive gas with color C and opacity α
  - Front-to-back strategy, starts with $C^{in} = (0,0,0)$ and $\alpha^{in} = 0$

$$C^{out} = C^{in} + \left(1 - \alpha^{in}\right) \alpha C$$
$$\alpha^{out} = \alpha^{in} + \left(1 - \alpha^{in}\right) \alpha$$



front-to-back strategy

Ray casting

- Example

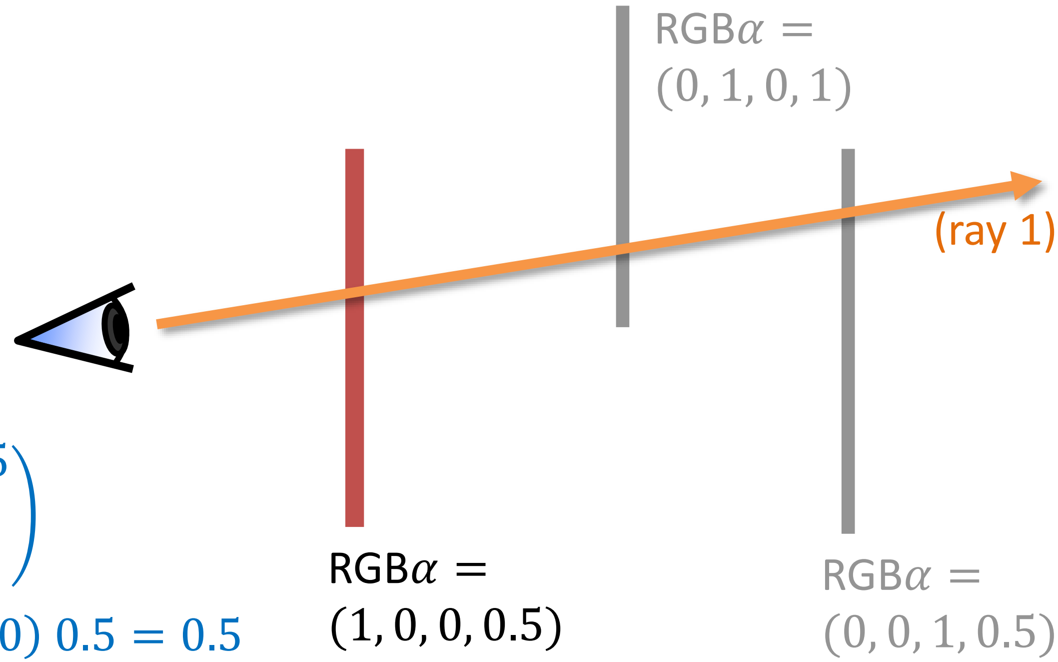
$\alpha = 1 \rightarrow$ completely opaque
$\alpha = 0 \rightarrow$ completely transparent

Ray 1: $C_{in} = (0,0,0)^T \quad \alpha_{in} = 0$

$$C_{out} = C_{in} + (1 - \alpha_{in})\, \alpha\, C$$

$$= \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} + (1 - 0)\, 0.5 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0.5 \\ 0 \\ 0 \end{pmatrix}$$

$$\alpha_{out} = \alpha_{in} + (1 - \alpha_{in})\, \alpha = 0 + (1 - 0)\, 0.5 = 0.5$$

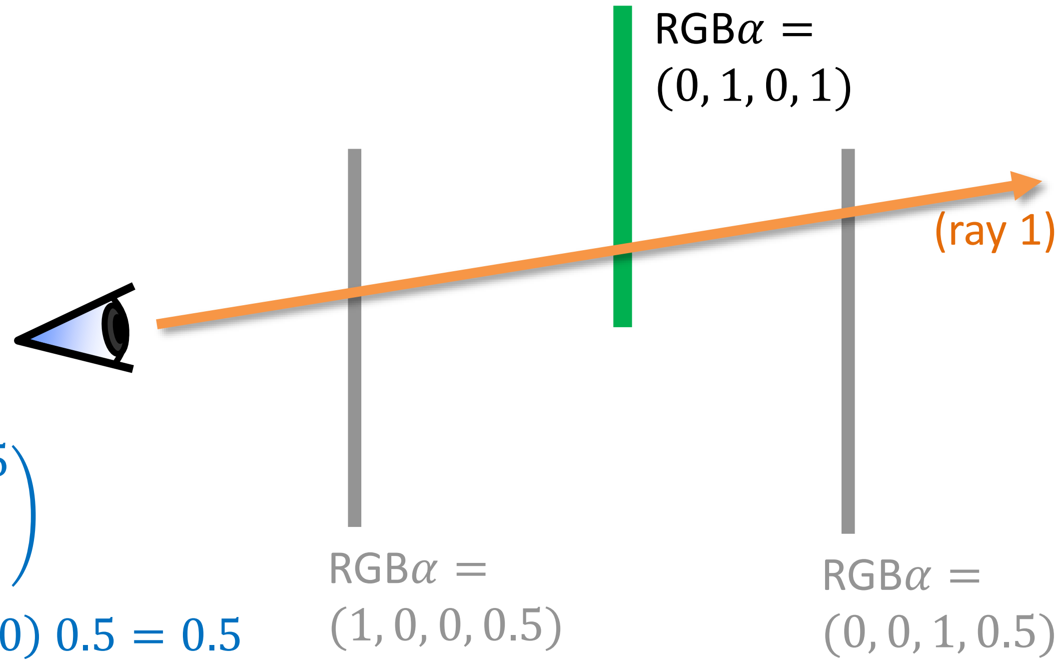


Ray casting

SIEMENS
Ingenuity for life

- Example

  $\alpha = 1 \rightarrow$ completely opaque
  $\alpha = 0 \rightarrow$ completely transparent



Ray 1: $C_{in} = (0, 0, 0)^T \quad \alpha_{in} = 0$

$$C_{out} = C_{in} + (1 - \alpha_{in})\, \alpha\, C$$

$$= \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} + (1-0)\, 0.5 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0.5 \\ 0 \\ 0 \end{pmatrix}$$

$$\alpha_{out} = \alpha_{in} + (1 - \alpha_{in})\, \alpha = 0 + (1-0)\, 0.5 = 0.5$$

$$C_{out} = \begin{pmatrix} 0.5 \\ 0 \\ 0 \end{pmatrix} + (1-0.5)\, 1 \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0.5 \\ 0.5 \\ 0 \end{pmatrix}$$

$$\alpha_{out} = 0.5 + (1-0.5)\, 1 = 1$$

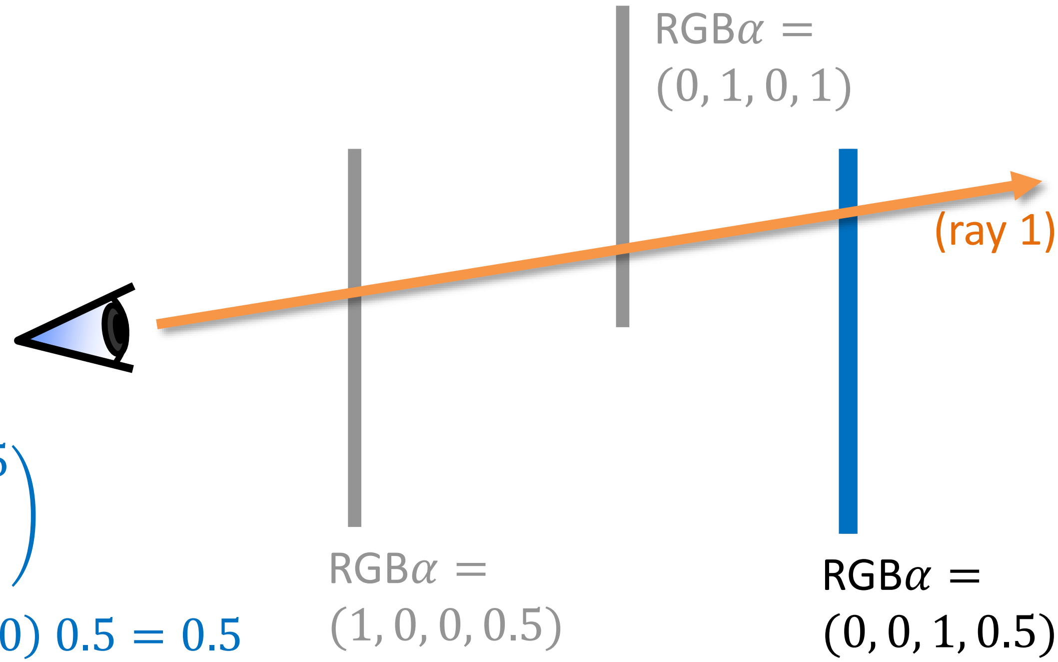
Early ray termination:

Stop calculation when $\alpha_{out} \approx 1$

Ray casting

- Example

$\alpha = 1 \rightarrow$ completely opaque
$\alpha = 0 \rightarrow$ completely transparent



Ray 1: $C_{in} = (0,0,0)^T \quad \alpha_{in} = 0$

$$C_{out} = C_{in} + (1-\alpha_{in})\,\alpha\,C$$

$$= \begin{pmatrix}0\\0\\0\end{pmatrix} + (1-0)\,0.5\begin{pmatrix}1\\0\\0\end{pmatrix} = \begin{pmatrix}0.5\\0\\0\end{pmatrix}$$

$$\alpha_{out} = \alpha_{in} + (1-\alpha_{in})\,\alpha = 0 + (1-0)\,0.5 = 0.5$$

$$C_{out} = \begin{pmatrix}0.5\\0\\0\end{pmatrix} + (1-0.5)\,1\begin{pmatrix}0\\1\\0\end{pmatrix} = \begin{pmatrix}0.5\\0.5\\0\end{pmatrix}$$

$$\alpha_{out} = 0.5 + (1-0.5)\,1 = 1$$

$$C_{out} = \begin{pmatrix}0.5\\0.5\\0\end{pmatrix} + (1-1)\,0.5\begin{pmatrix}0\\0\\1\end{pmatrix} = \begin{pmatrix}0.5\\0.5\\0\end{pmatrix}$$

$$\alpha_{out} = 1 + (1-1)\,0.5 = 1$$

Early ray termination:

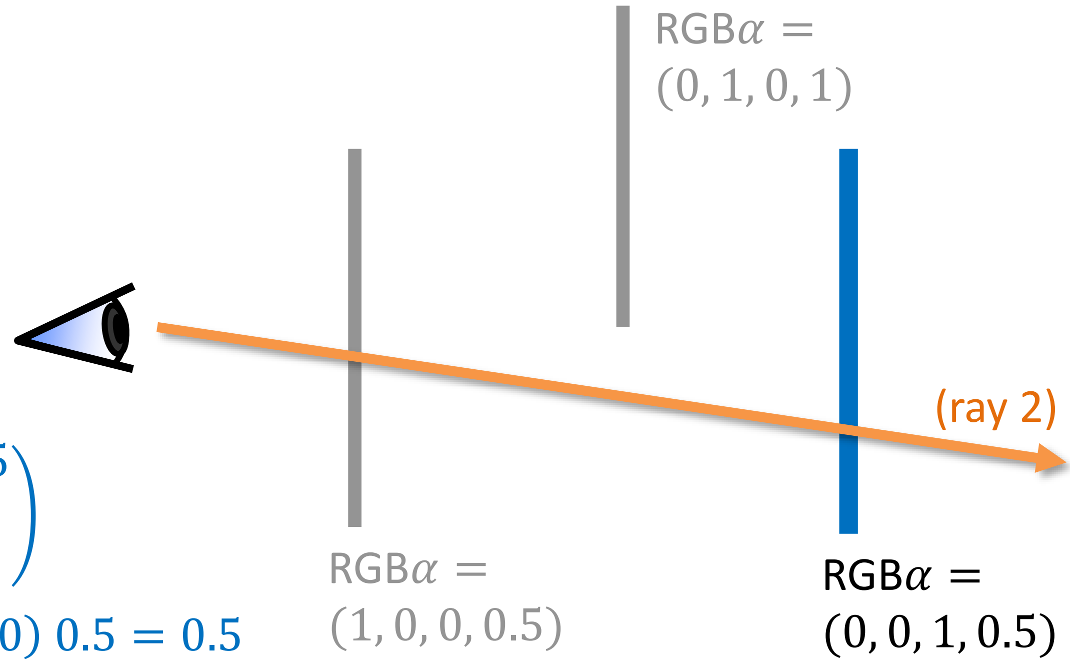
Stop calculation when $\alpha_{out} \approx 1$

Last object does not affect result since $\alpha_{in} = 1$

Ray casting

- Example

$\alpha = 1 \rightarrow$ completely opaque
$\alpha = 0 \rightarrow$ completely transparent



Ray 2: $C_{in} = (0,0,0)^{\mathrm{T}} \quad \alpha_{in} = 0$

$$C_{out} = C_{in} + (1-\alpha_{in})\,\alpha\, C$$

$$= \begin{pmatrix}0\\0\\0\end{pmatrix} + (1-0)\,0.5\begin{pmatrix}1\\0\\0\end{pmatrix} = \begin{pmatrix}0.5\\0\\0\end{pmatrix}$$

$$\alpha_{out} = \alpha_{in} + (1-\alpha_{in})\,\alpha = 0 + (1-0)\,0.5 = 0.5$$

$$C_{out} = \begin{pmatrix}0.5\\0\\0\end{pmatrix} + (1-0.5)\,0.5\begin{pmatrix}0\\0\\1\end{pmatrix} = \begin{pmatrix}0.5\\0\\0.25\end{pmatrix}$$

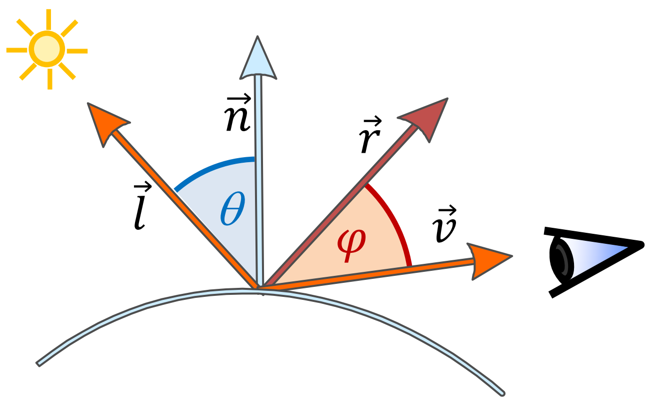
$$\alpha_{out} = 0.5 + (1-0.5)\,0.5 = 0.75$$

Lighting

- Evaluate Phong's illumination model based on...
  - position of current sample and light source
  - samples' color emission assigned by transfer function
  - sample's normal / gradient (e.g., central differences)

DVR without shading

Phong shading



Ray casting

SIEMENS
Ingenuity for life

- Gradient estimation
  - The gradient vector (normal) is the first-order derivative of the scalar field

$$\nabla f(\boldsymbol{x}) = \begin{pmatrix} \frac{\partial}{\partial x} f(\boldsymbol{x}) \\ \frac{\partial}{\partial y} f(\boldsymbol{x}) \\ \frac{\partial}{\partial z} f(\boldsymbol{x}) \end{pmatrix}$$

partial derivative in x-direction

partial derivative in y-direction

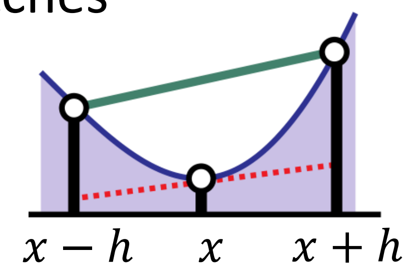
partial derivative in z-direction

  - We can estimate the gradient using finite differencing schemes (e.g., forward or central differences)

Ray casting

- Gradient estimation
  - Compute on-the-fly within fragment shader
    - Central differences require 6 additional texture fetches

$$\nabla f(x,y,z) \approx \frac{1}{2h}\begin{pmatrix} f(x+h,y,z) - f(x-h,y,z) \\ f(x,y+h,z) - f(x,y-h,z) \\ f(x,y,z+h) - f(x,y,z-h) \end{pmatrix}$$



  - Pre-compute and store together with voxel data
    - Pack into 3D RGBA texture for GPU-based rendering
    - One lookup to get interpolated gradient + value
    - Requires more memory → not applicable for large volume data

Voxel Data
- X Gradient
- Y Gradient
- Z Gradient
- Value

→

3D Texture
- R
- G
- B
- A

Ray casting

- Gradient estimation

Pre-computed,
8 bit per component

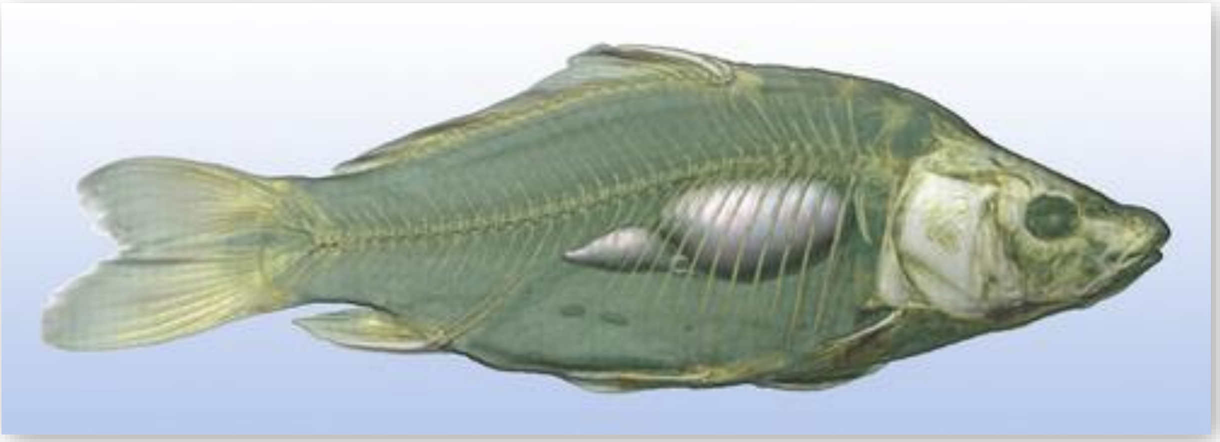
on-the-fly
computation

Ray casting

- Examples







Compositing Schemes

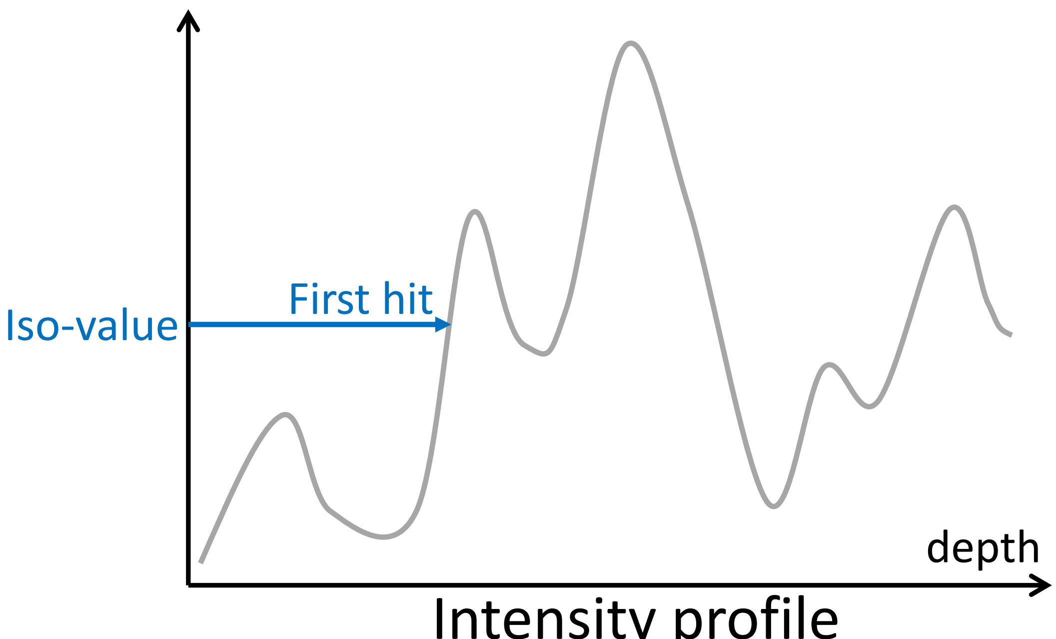
SIEMENS
Ingenuity for life

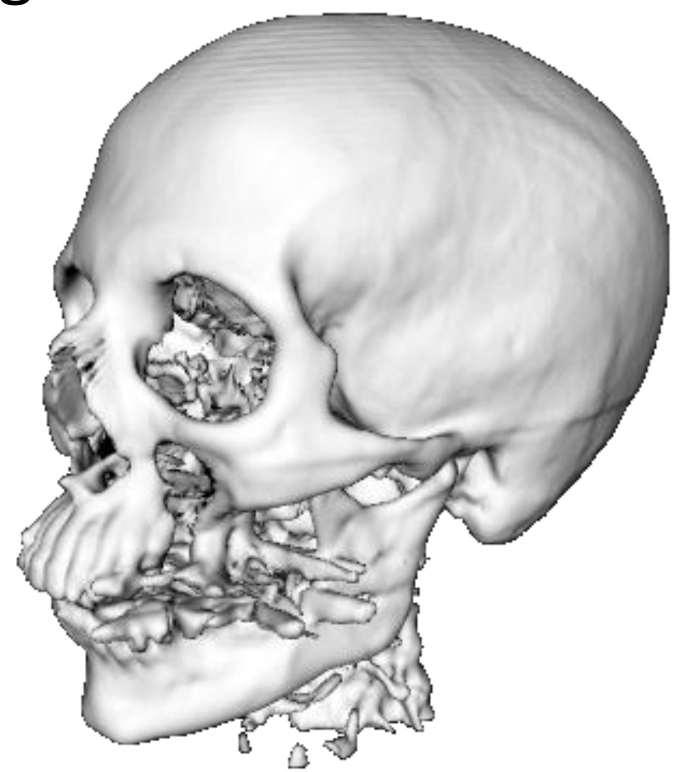
- Variations of compositing schemes
  - Surface rendering / first hit: stop ray traversal if an iso-surface is hit, and shade the surface points
  - Average: simply accumulate colors but do not account for opacity
  - Maximum: only takes the maximum color along the ray and displays it

Compositing Schemes

SIEMENS
Ingenuity for life

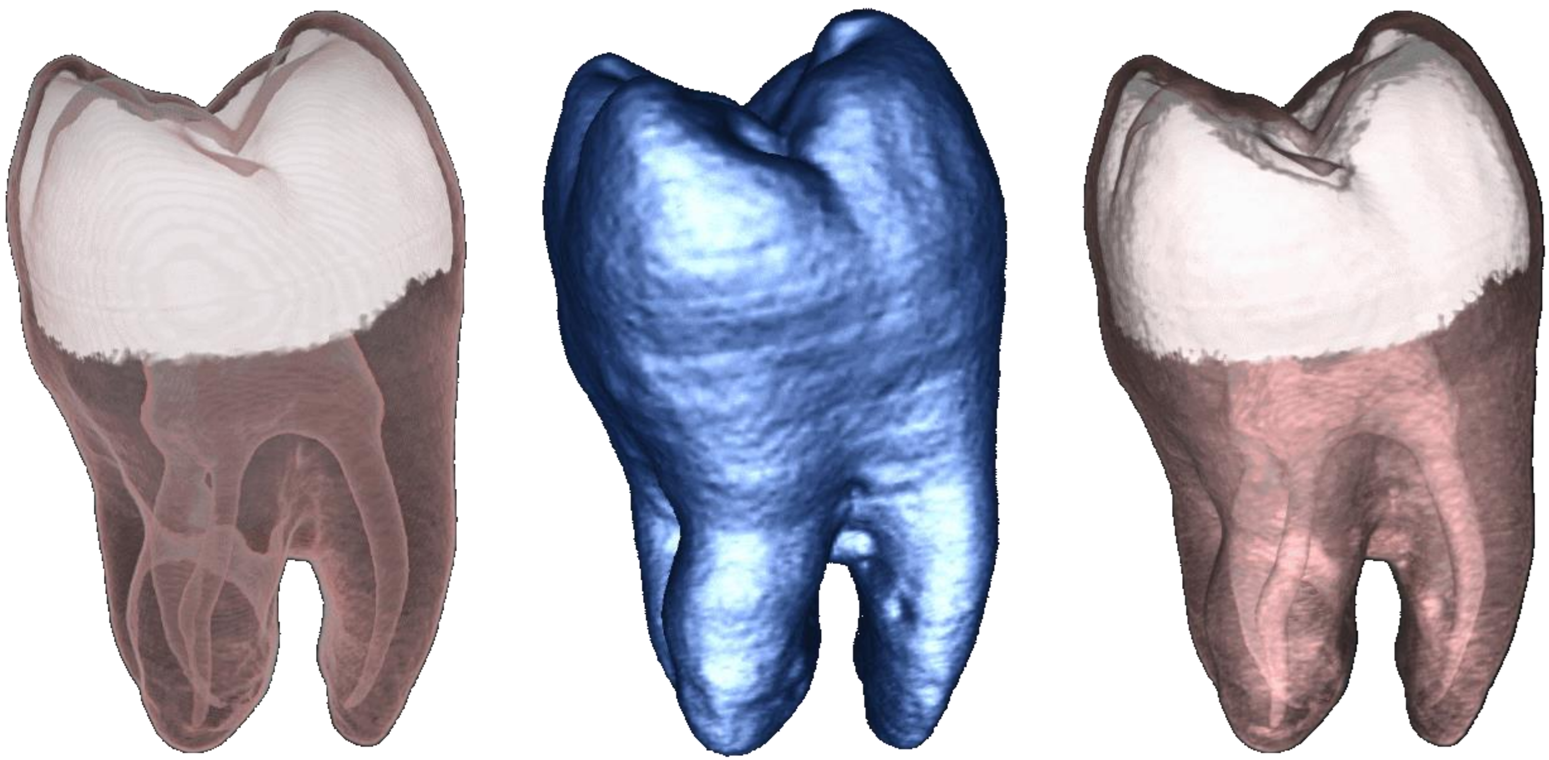
- Surface rendering / first hit
  - Stop ray traversal if an iso-surface is hit, and shade the surface points
  - Produces same result as marching cubes, but with higher accuracy





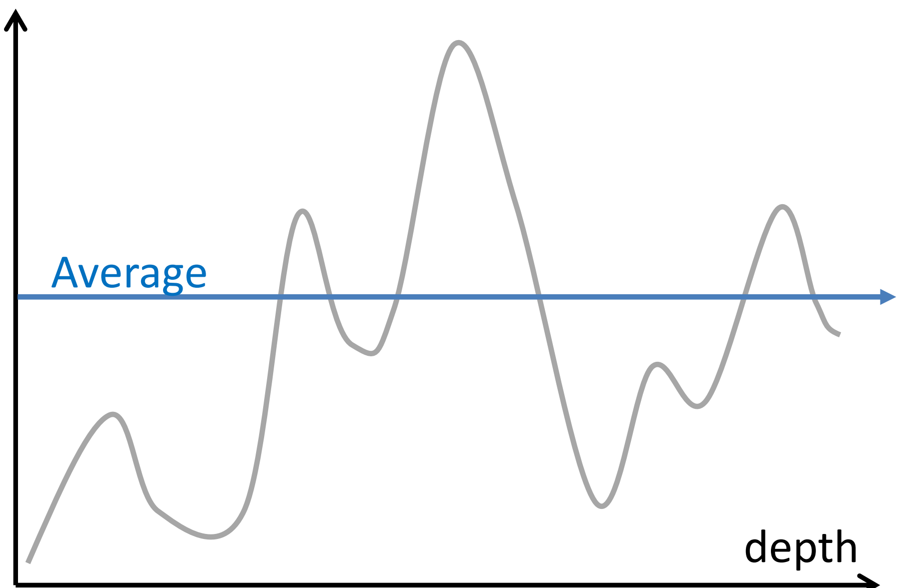
Compositing Schemes

- Compositing: First hit (middle image)



Compositing Schemes

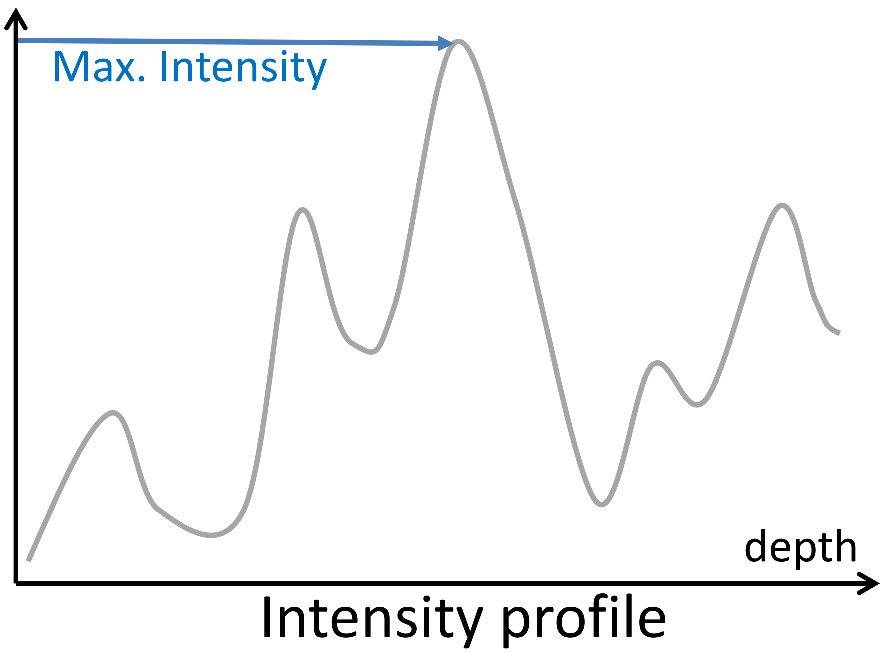
- Compositing: Average
  - Accumulate colors but do not account for opacity
  - Produces an X-ray image



Compositing Schemes

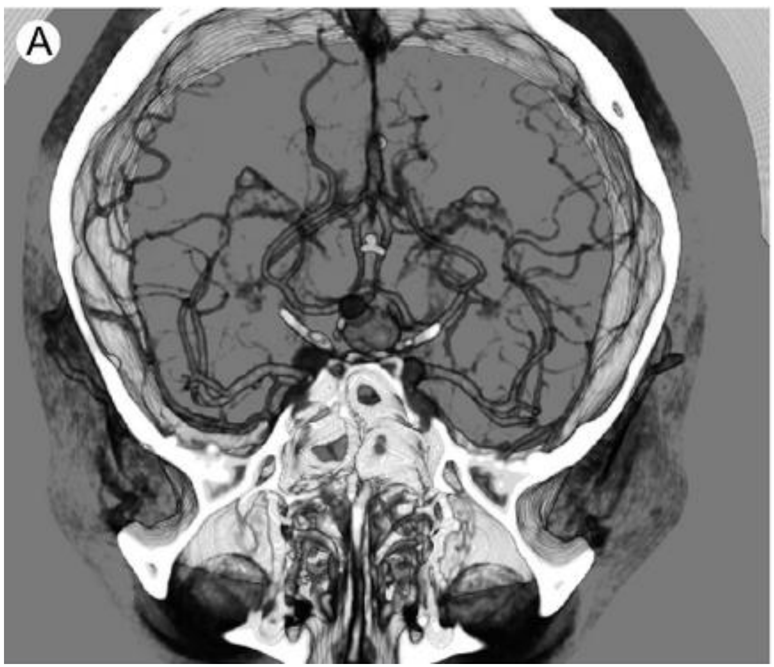
SIEMENS
Ingenuity for life

- Maximum Intensity Projection (MIP)
  - Only takes the maximum color along the ray and displays it

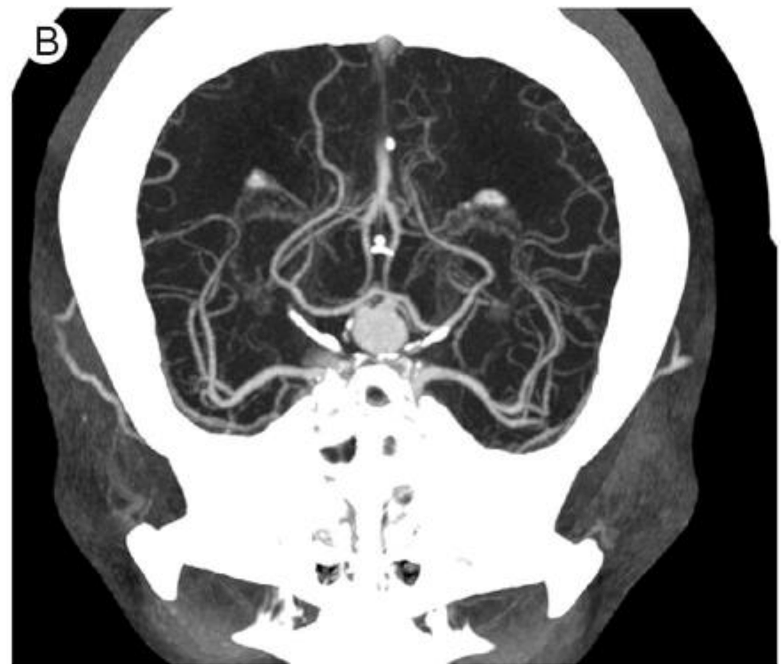


Compositing Schemes

- Maximum Intensity Projection (MIP)
  - Often used for magnetic resonance angiograms
  - Good to extract vessel structures



Direct volume rendering

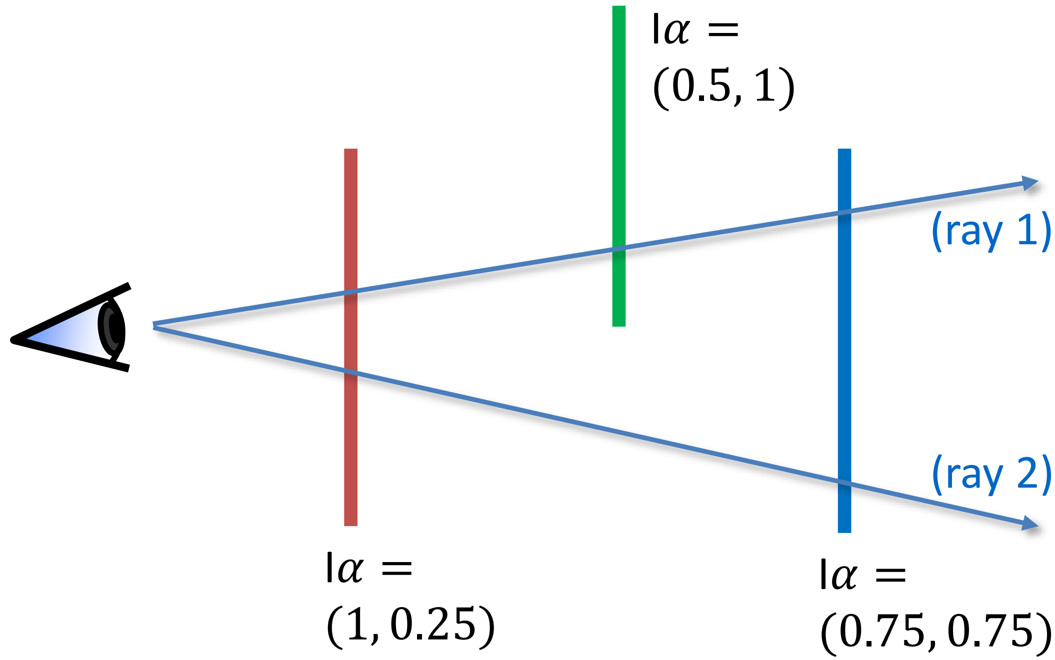


Maximum intensity projection

Ray casting

- Example

I… intensity
$\alpha = 1$ → completely opaque
$\alpha = 0$ → completely transparent



Ray 1 with average compositing:

$$I_{out} = \frac{1}{3}(1 + 0.5 + 0.75) = 0.75$$

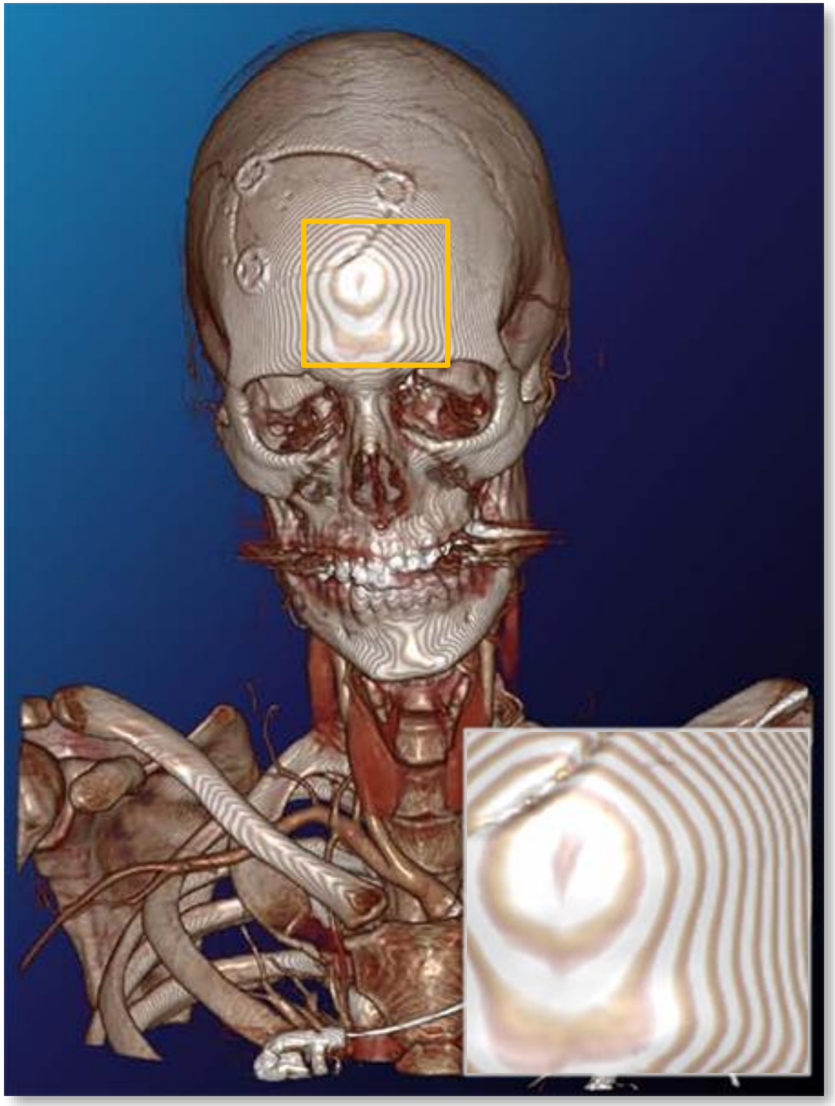
Ray 2 with maximum intensity projection:

$$I_{out} = \max(1, 0.75) = 1$$

Both average compositing and MIP do not account for opacity

Ray casting

- Sampling artifacts
  - To few samples along the ray
  - Solution: Increase sampling rate to Nyquist frequency
    → at least 2 samples per voxel

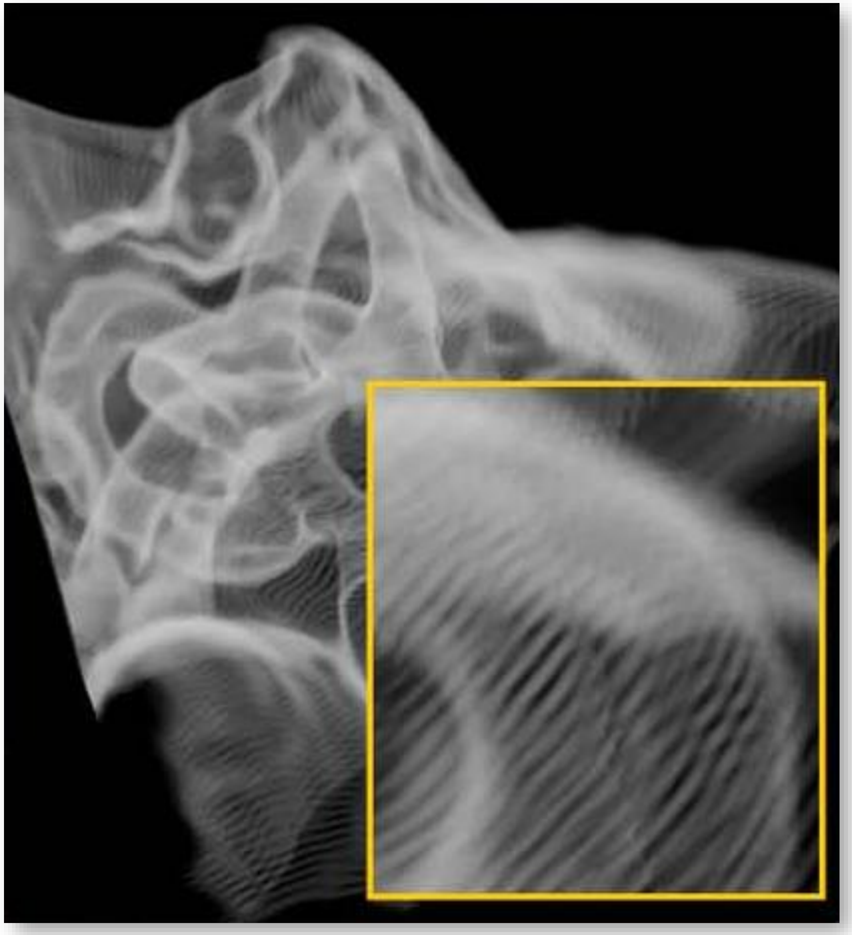


[Engel et al. 06]

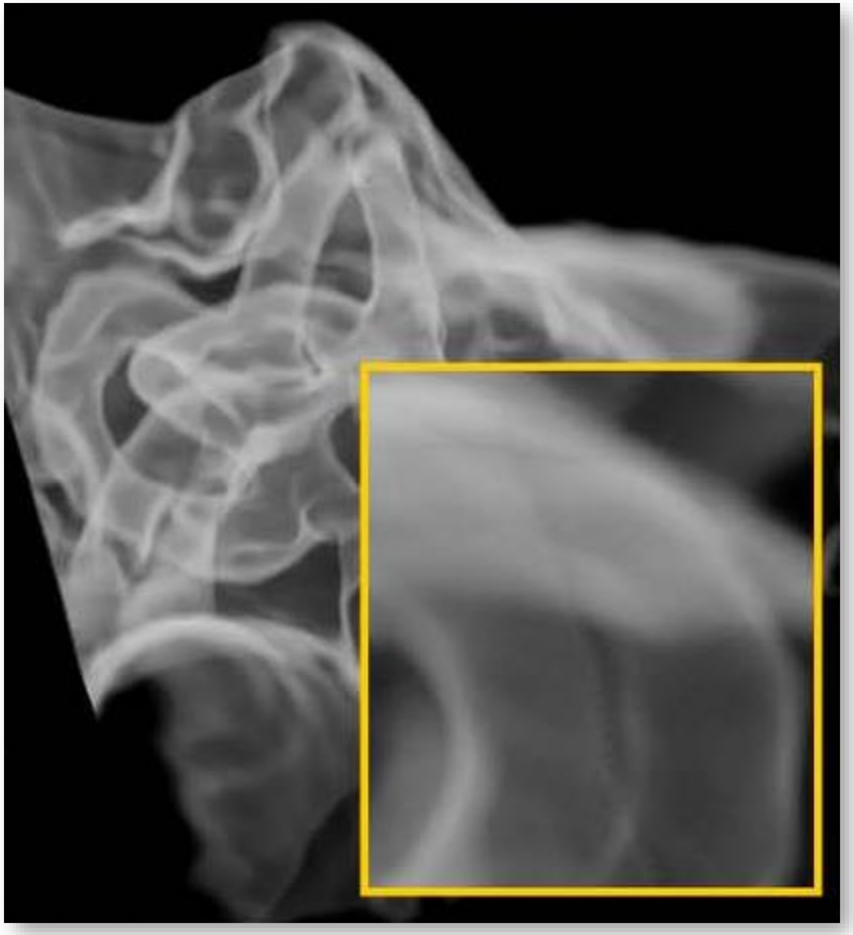
Ray casting

SIEMENS
Ingenuity for life

- Sampling artifacts



128 samples



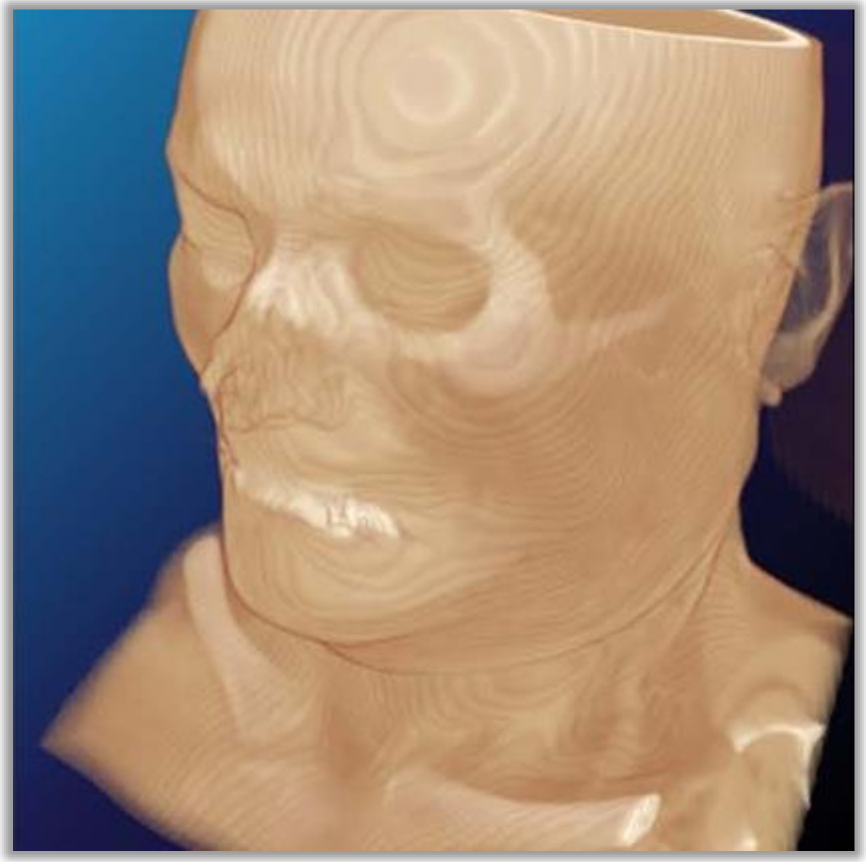
284 samples

[Engel et al. 06]

Ray casting

SIEMENS
Ingenuity for life

- Remove artifacts by stochastic jittering of ray-start position



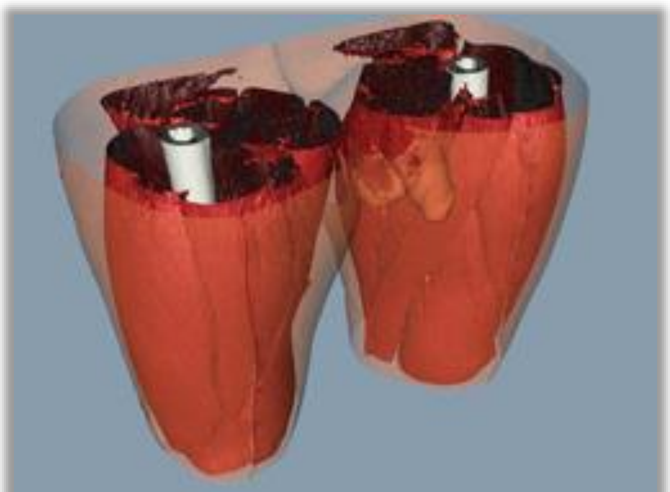
[Engel et al. 06]

Summary

SIEMENS
Ingenuity for life

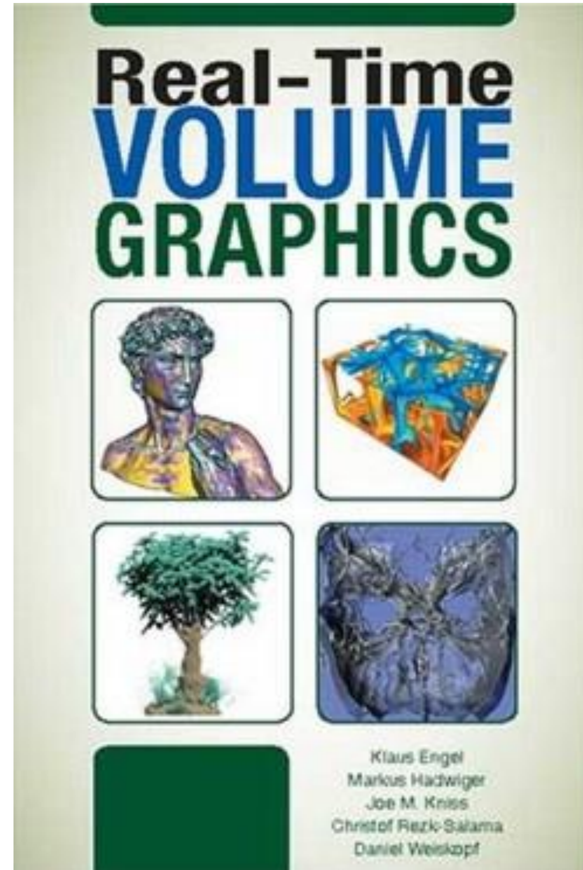
- Surface rendering
  - Indirect representation
  - Conveys surface impression
  - Hardware supported rendering (fast?!)
  - Iso-value-definition

- Direct volume rendering
  - Direct representation
  - Conveys volume impression
  - Often realized in software (slow?!)
  - Transfer function specification

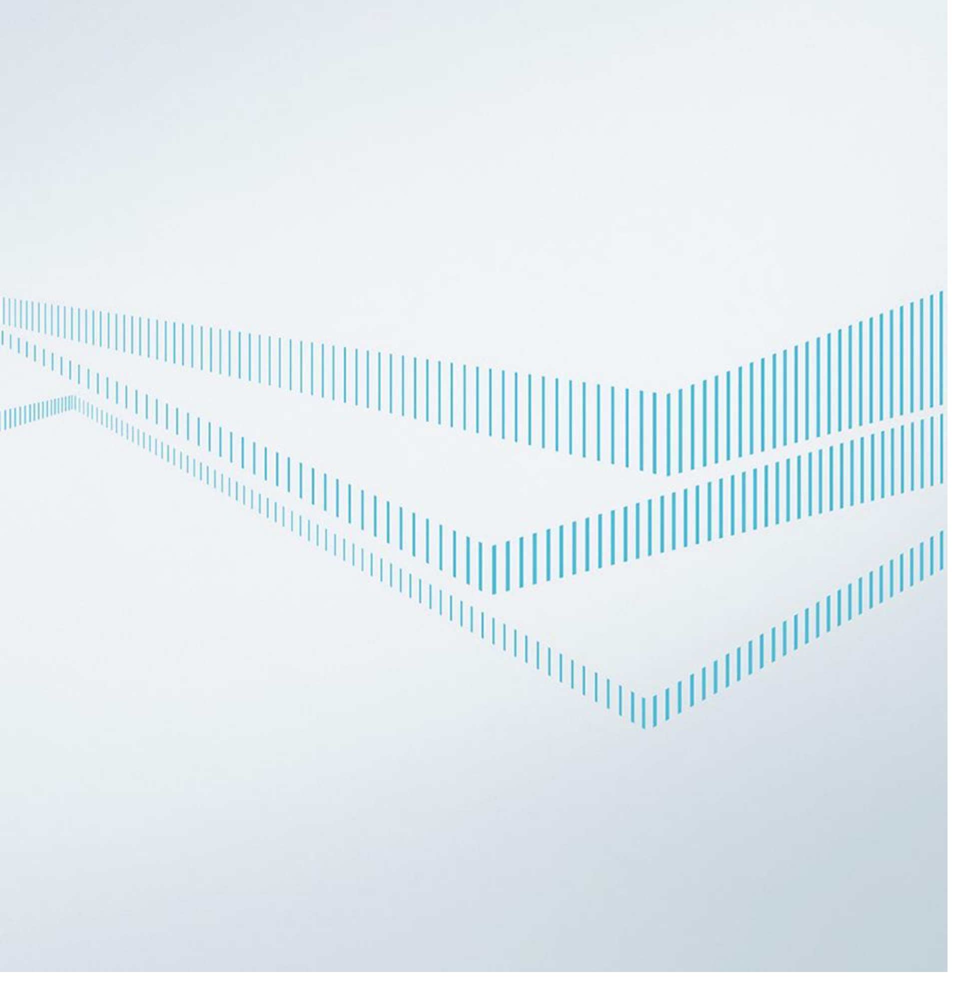


Further Resources

- Engel et al. 06:
  Real-time volume graphics
- Tutorial slides:
  www.real-time-volume-graphics.org



Contact information

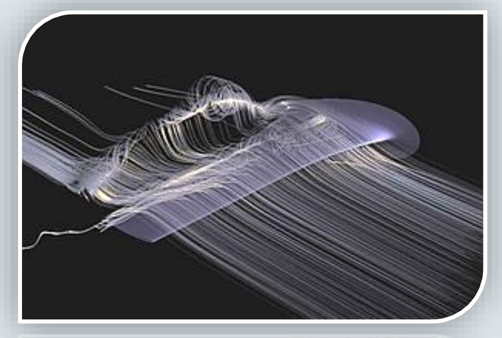
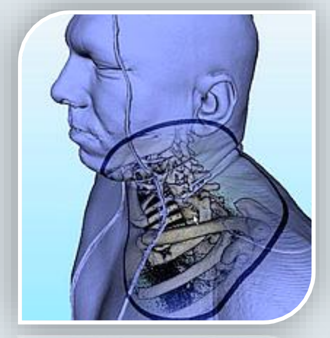


Dr. Johannes Kehrer

Siemens Technology
T RDA BAM IBI-DE
Otto-Hahn-Ring 6
81739 München, Deutschland

E-mail:
kehrer.johannes@siemens.com

Internet
siemens.com/innovation

Visual Data Analytics
Flow Visualization I

Dr. Johannes Kehrer

Disclaimer

These lecture slides and materials are for personal use by students and are intended for educational purposes only. Distribution or reproduction of the material is prohibited.

The views and opinions expressed by any individual during the course of these lecturers are their own and do not necessarily represent the views, opinions, or positions of Siemens or the Technical University of Munich.

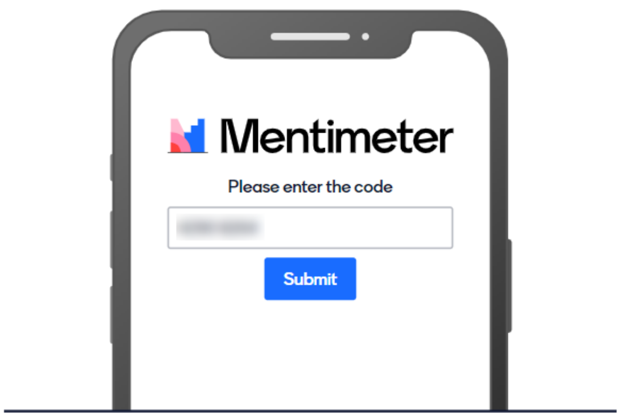
Although reasonable efforts have been made to ensure the information in these lecture slides is current and accurate, we do not assume any responsibility for the content of the posted material.

These slides also contain links to websites of third parties. As the content of these websites is not under our control, we cannot assume any liability for such external content.

Any comments/questions?

Go to

www.menti.com



Enter the code

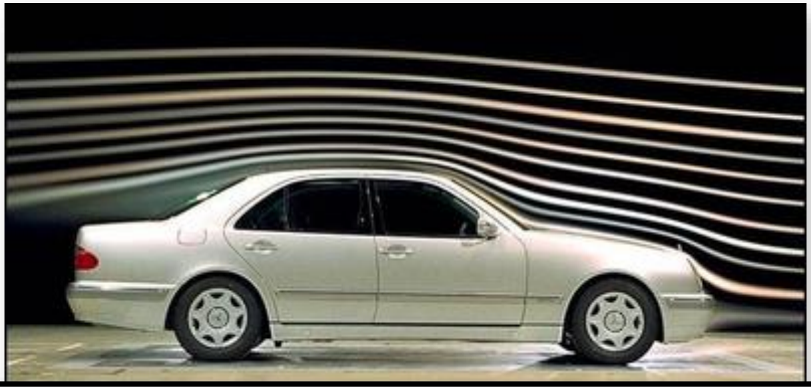
6248 9347

Or use QR code

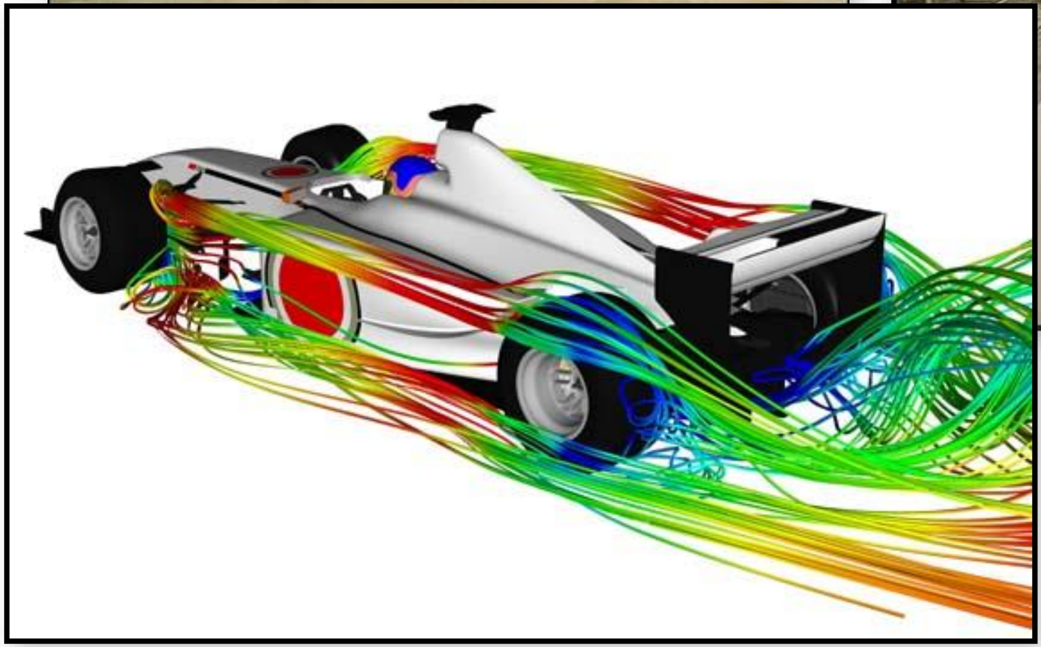
Flow visualization

SIEMENS
Ingenuity for life

Visualize vector-valued quantities like wind fields

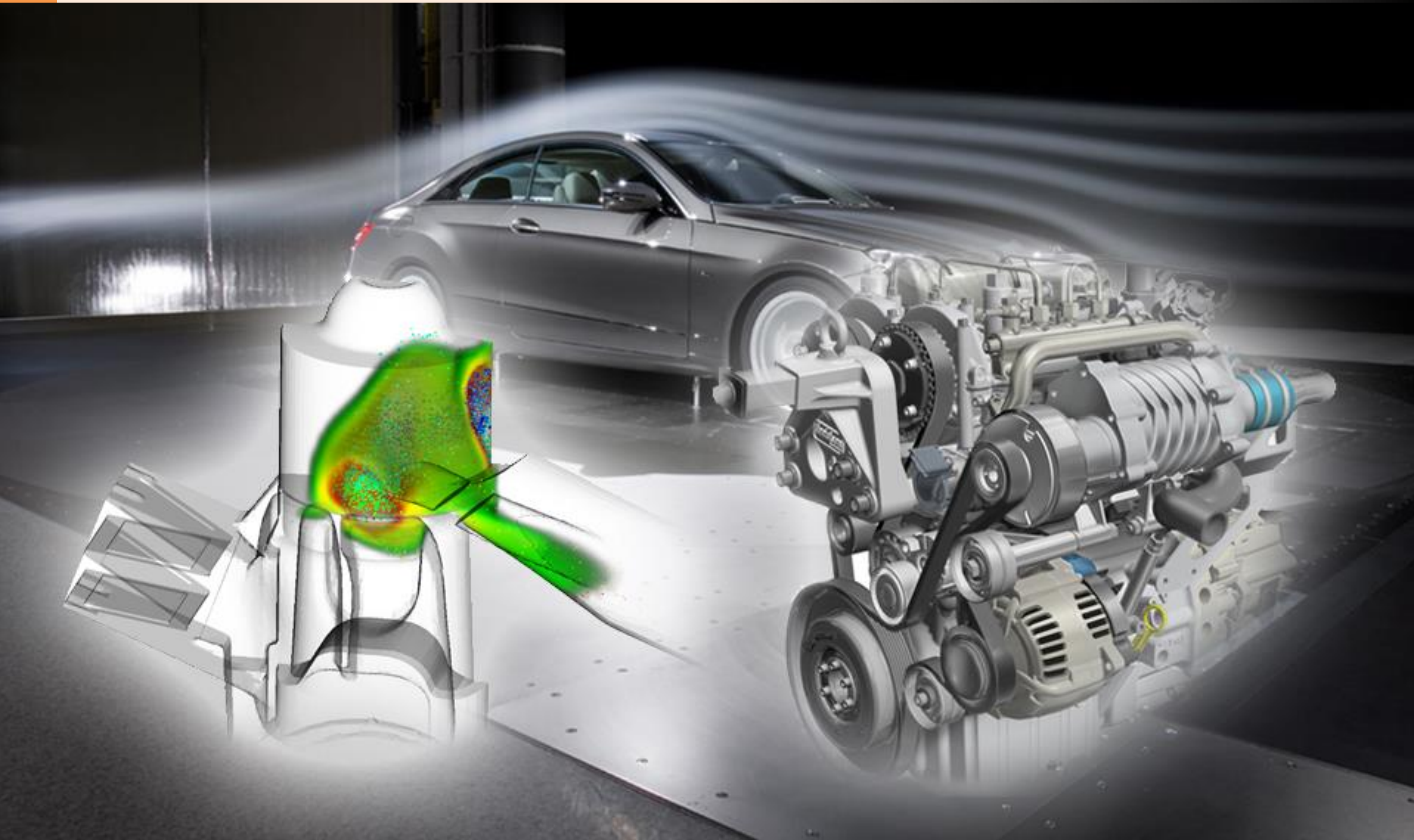


Real wind tunnel



Car design: virtual wind tunnel

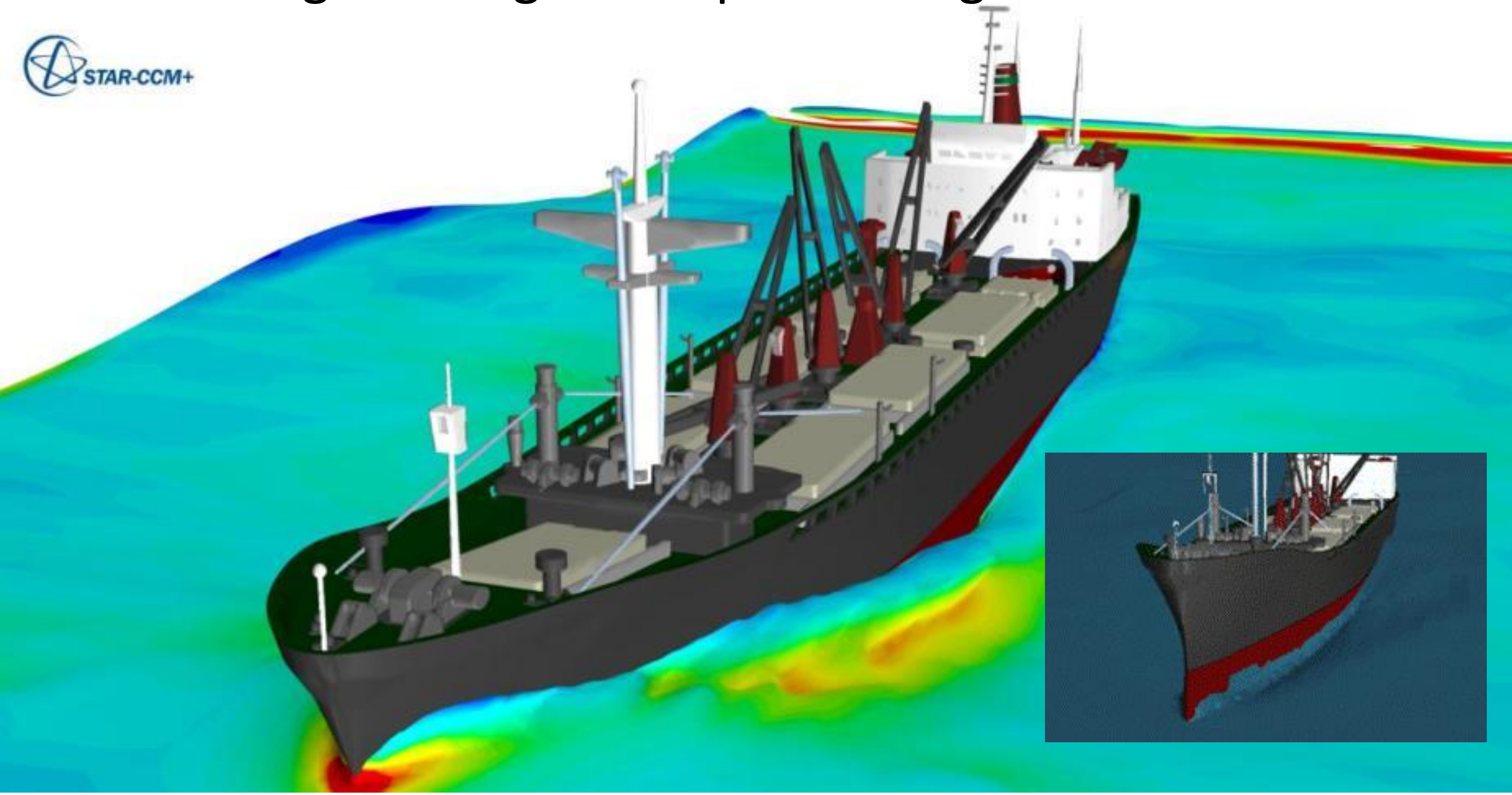
Automotive Engineering



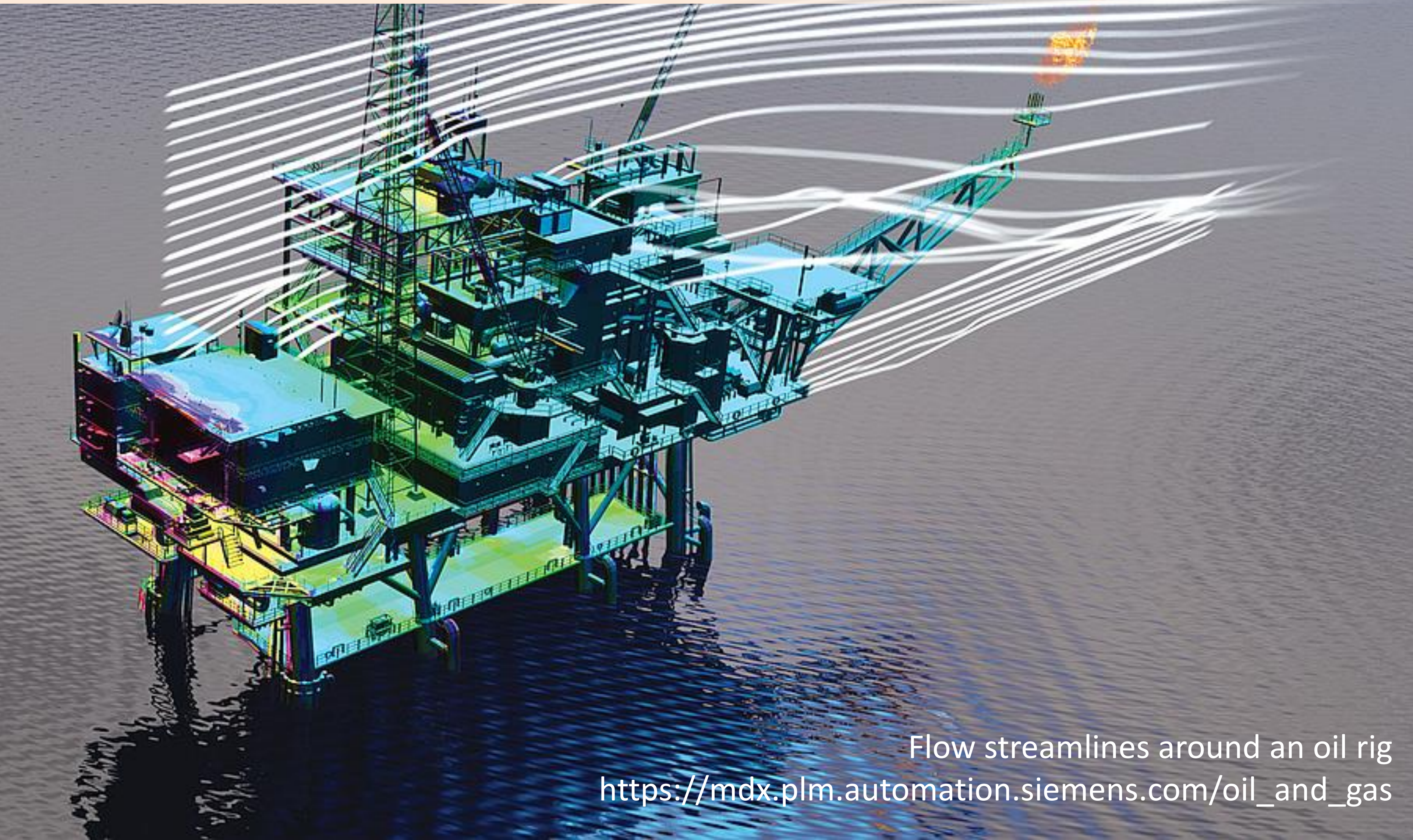
Ship design

SIEMENS
Ingenuity for life
CD-adapco
A Siemens Business

- Simulates forces acting on everything from engine design to ship hull design

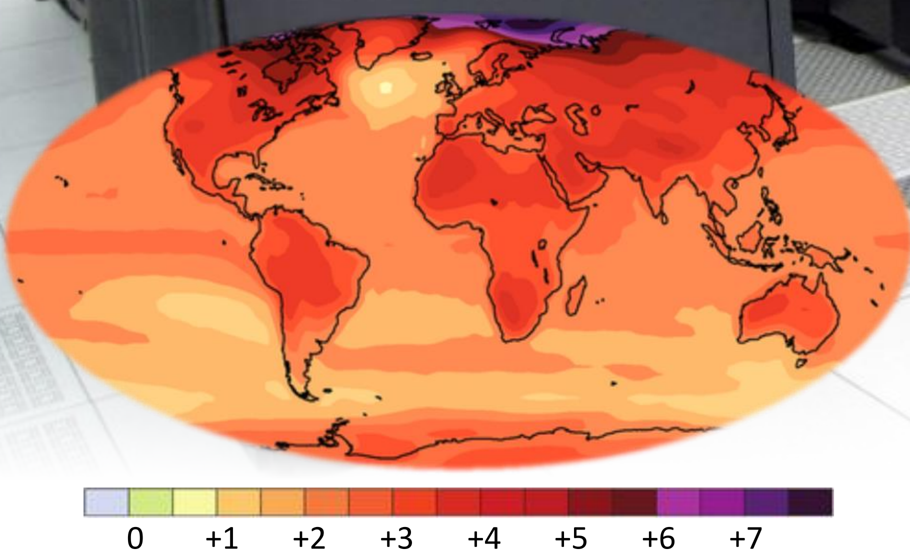


Oil & Gas



Flow streamlines around an oil rig
https://mdx.plm.automation.siemens.com/oil_and_gas

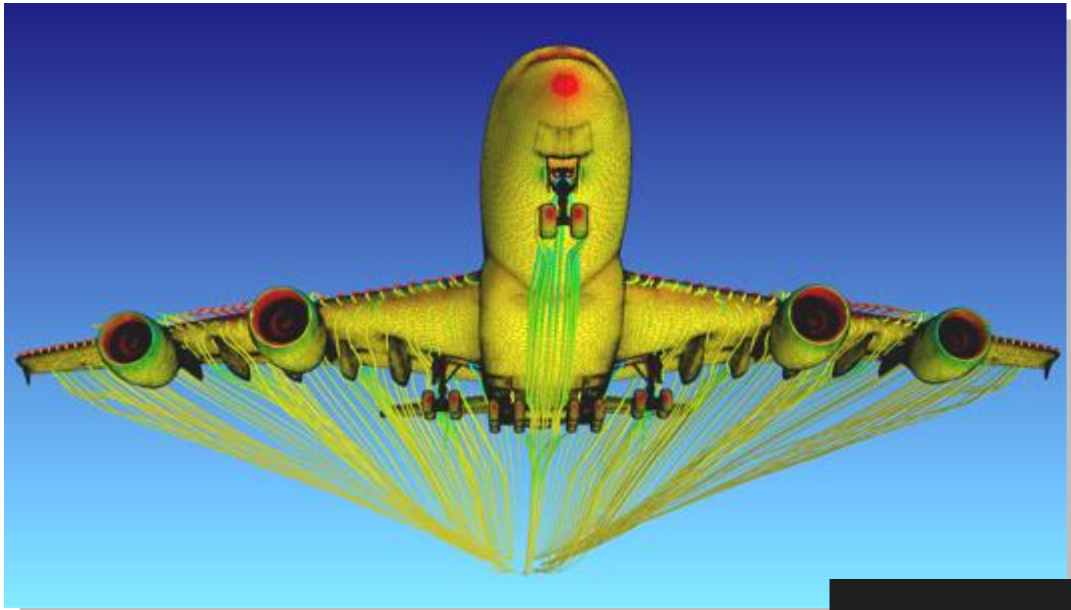
Weather & Climate Simulations

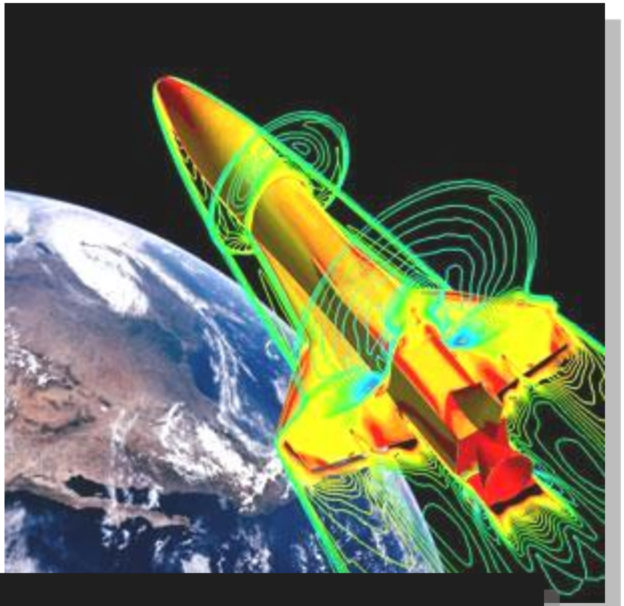


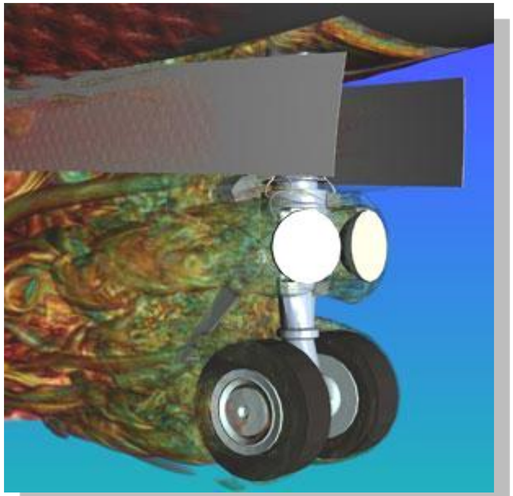
Changes in Temperature 2090-2099 (°C)

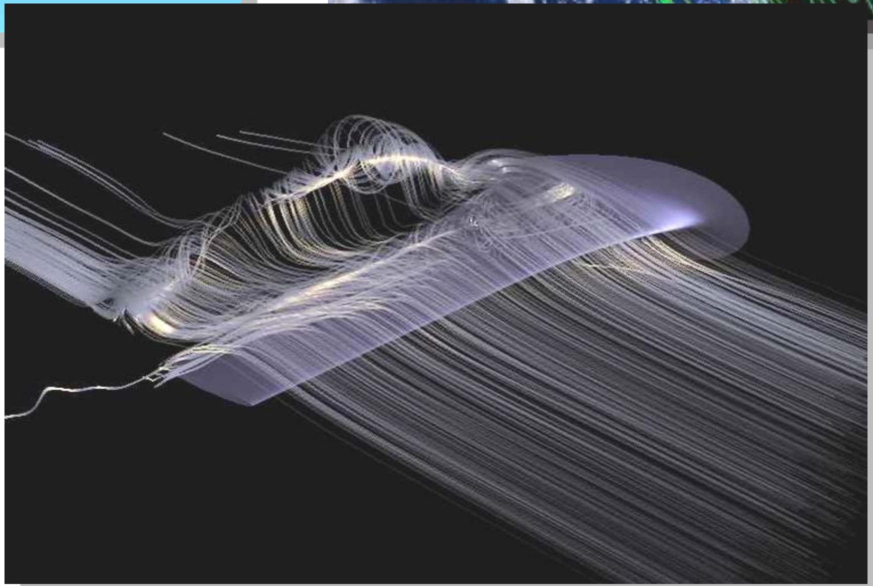
Aerospace design

SIEMENS
Ingenuity for life





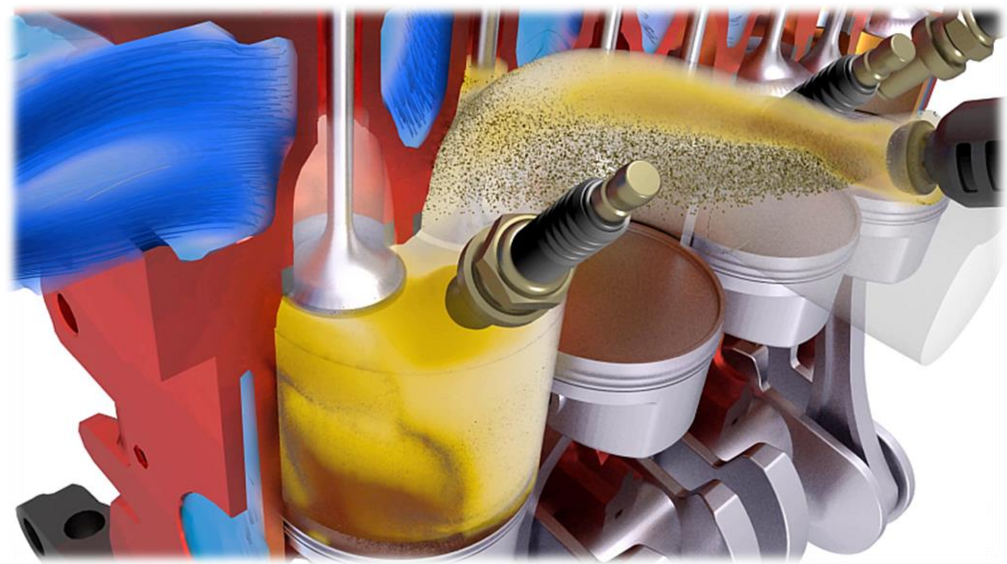




https://mdx.plm.automation.siemens.com/aerospace

Flow Visualization – Examples

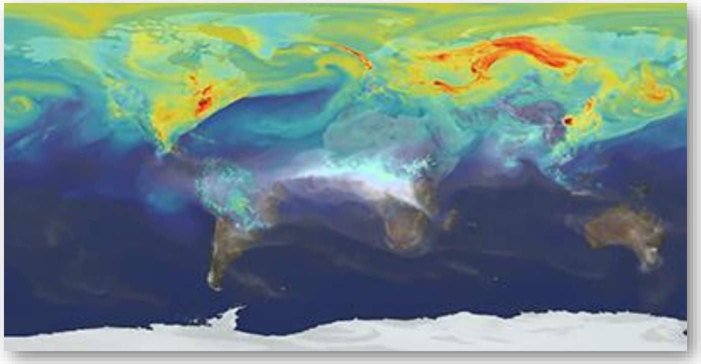
SIEMENS
Ingenuity for life



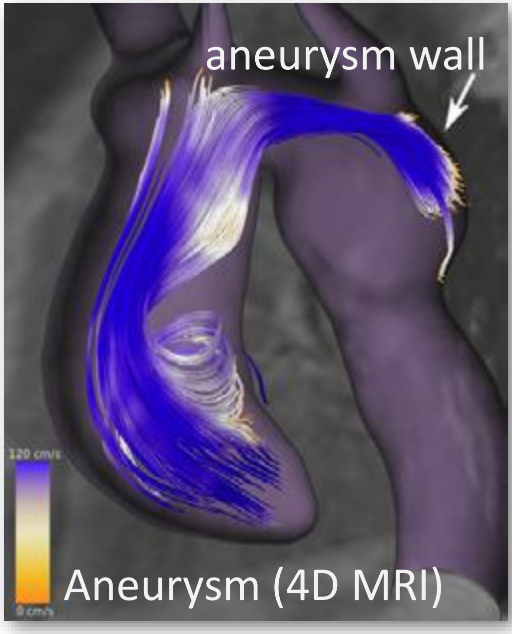
Simcenter STAR-CCM+
https://youtu.be/443kbDFPjUo

Flow visualization – data sources

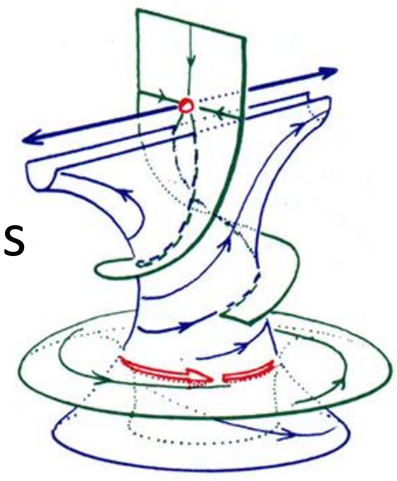
SIEMENS
Ingenuity for life

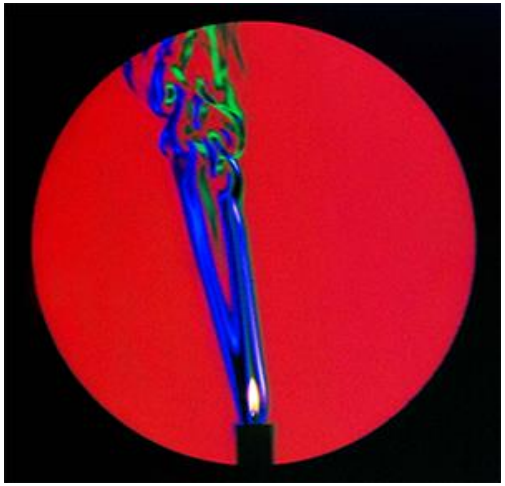


- Flow simulation
  - Design of ships, cars, airplanes, …
  - Weather simulations (e.g., atmospheric flow)
  - Medical blood flows
- Measurements
  - Wind tunnel
  - Schlieren imaging
- Modeling
  - Differential equations systems (dynamical systems)



[Born et al. 2012]





Color schlieren of burning candle

Vector field visualization

- Main application of flow visualization
  - Motion of fluids (gases, liquids)
  - Geometric boundary conditions
  - Velocity/flow field $\boldsymbol{v}(\boldsymbol{x}, t)$
  - Conservation of mass, energy, and momentum
  - Navier-Stokes equations
  - Computational fluid dynamics (CFD)

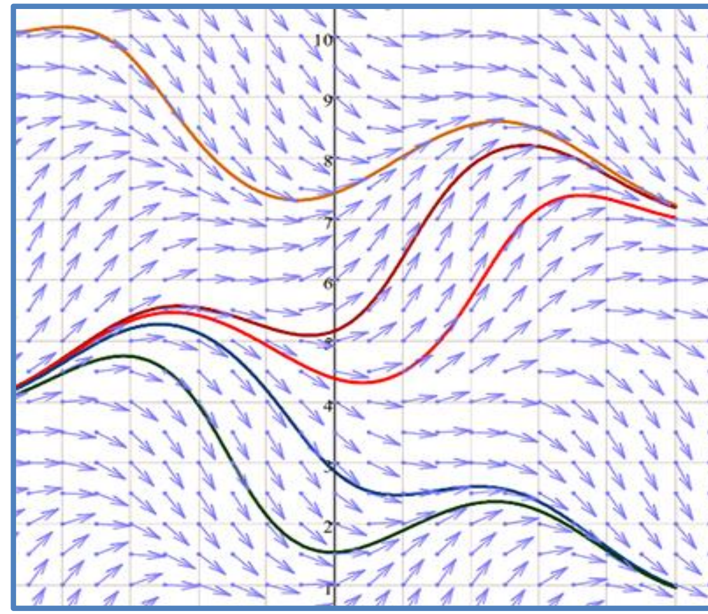
Vector field visualization

- Flow visualization – classification
  - Dimension (2D or 3D)
  - Time-dependency: steady vs. time-varying flows
  - Direct vs. indirect flow visualization

- In most cases, numerical methods required for flow visualization

Vector field visualization

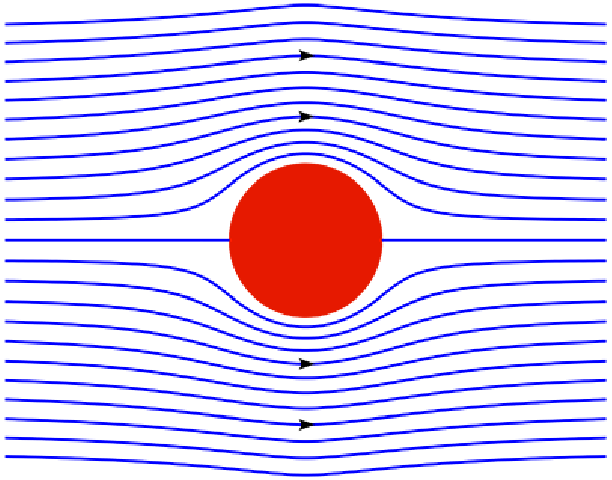
- Vector data representing direction & magnitude
  - Given by an n-tupel $(f_1, \ldots, f_n)$ with $f_k = f_k(x_1, \ldots, x_n)$, $n \geq 2$ and $1 \leq k \leq n$
  - Typically 2D ($n = k = 2$) or 3D ($n = k = 3$)



- Example
  - 2D vector field where every sample represents a 2D vector (u, v) with $u = f(x, y)$ and $v = g(x, y)$

Steady vs. time-dependent flows

SIEMENS
Ingenuity for life



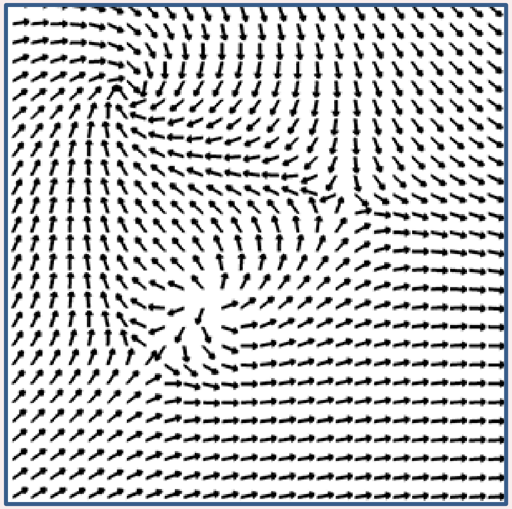
- Steady (time-independent) flow
  - Flow static over time
  - $\boldsymbol{v}(\boldsymbol{x}) : \mathbb{R}^n \rightarrow \mathbb{R}^n$, e.g., laminar flows
  - Simpler interrelationships

Turbulence from an airplane wing

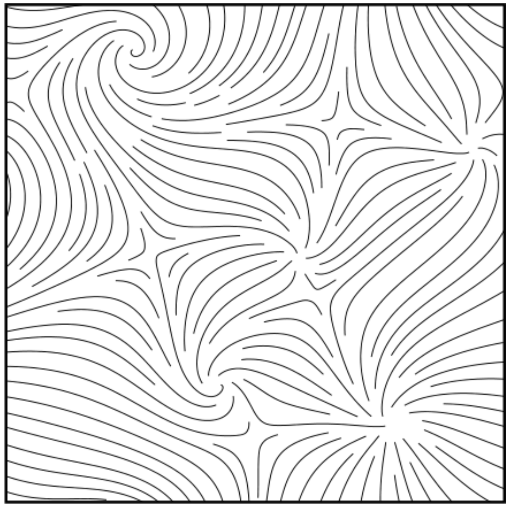
- Time-varying (unsteady) flow
  - Flow changes over time
  - $\boldsymbol{v}(\boldsymbol{x}, t) : \mathbb{R}^n \times \mathbb{R}^1 \rightarrow \mathbb{R}^n$, e.g., turbulent flows
  - More complex interrelationships

Flow visualization – Approaches

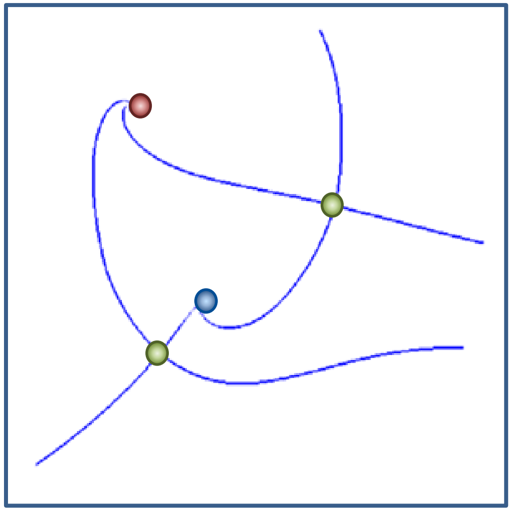
SIEMENS
Ingenuity for life



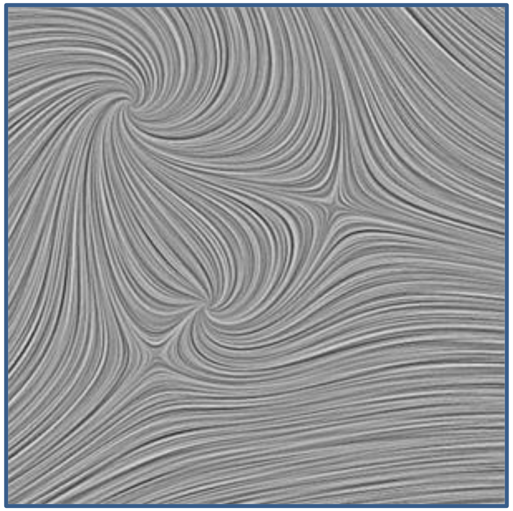
Direct flow visualization
(arrows, color coding, …)



Geometric flow visualization
(stream lines/surfaces, …)



Sparse (feature-based) vis.

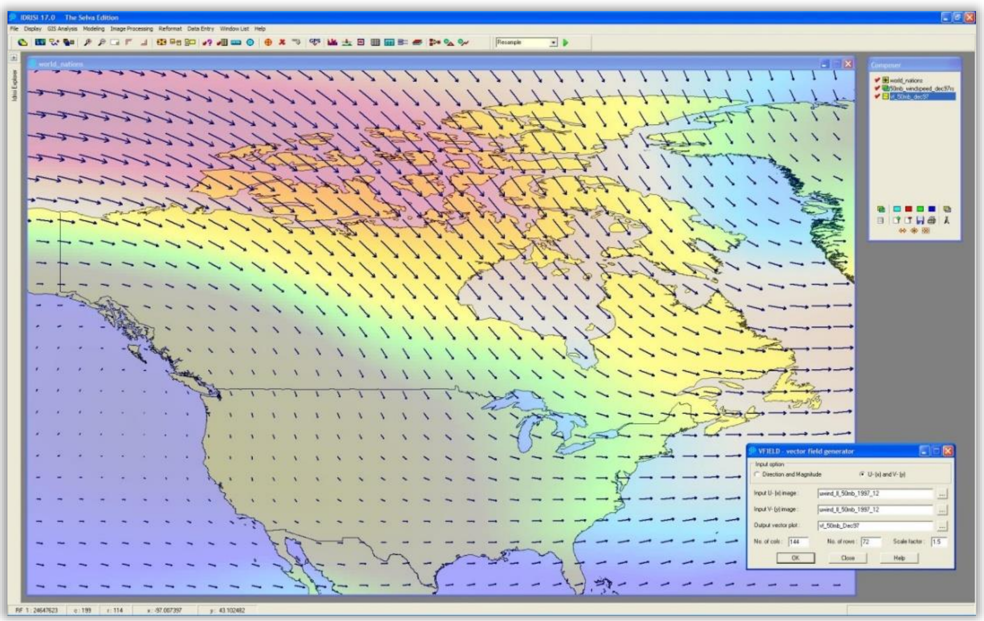


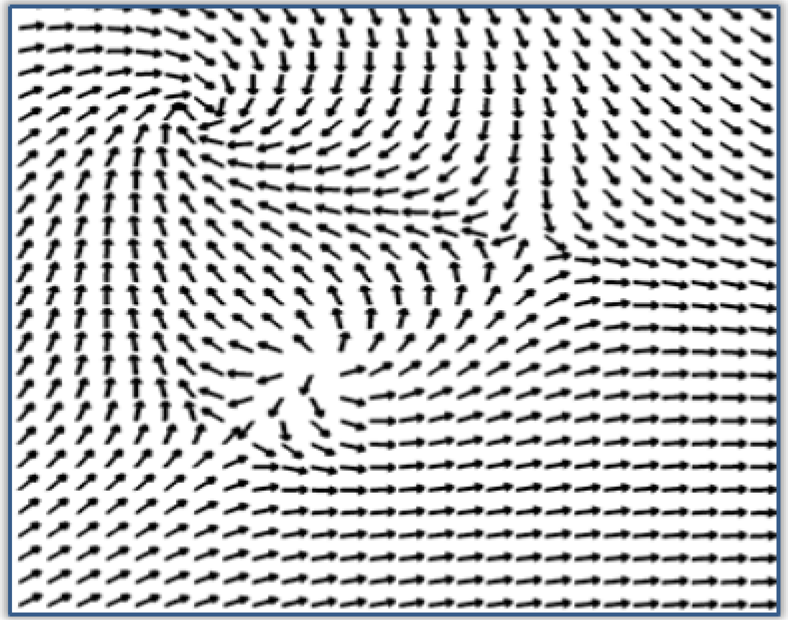
Dense (texture-based) vis.

Flow visualization – Approaches

SIEMENS
Ingenuity for life

- Direct flow visualization
  - Color coding, arrow plots, glyphs
  - Gives overview on current flow state
  - Visualization of vectors





Vector and scalar functions

SIEMENS
Ingenuity for life

- Scalar function $\rho(\boldsymbol{x}, t)$
  - Gradient

$$\nabla\rho(\boldsymbol{x}, t) = \begin{pmatrix} \frac{\partial}{\partial x}\rho(\boldsymbol{x}, t) \\ \frac{\partial}{\partial y}\rho(\boldsymbol{x}, t) \\ \frac{\partial}{\partial z}\rho(\boldsymbol{x}, t) \end{pmatrix}$$

$\boldsymbol{x}$ … position
t … time
∇… vector operator

Vector of partial derivatives

- Gradient takes a scalar field and gives us a vector field
- Gradient points into direction of max. change of $\rho(\boldsymbol{x}, t)$
- Gradient magnitude gives the slope along this direction

Vector and scalar functions

SIEMENS
Ingenuity for life

- Vector function $\boldsymbol{v}(\boldsymbol{x}, t)$
  - Jacobian matrix ("Gradient tensor")

$$\mathbf{J} = \nabla \boldsymbol{v}(\boldsymbol{x}, t) = \begin{pmatrix} \frac{\partial}{\partial x}\boldsymbol{v}_x & \frac{\partial}{\partial y}\boldsymbol{v}_x & \frac{\partial}{\partial z}\boldsymbol{v}_x \\ \frac{\partial}{\partial x}\boldsymbol{v}_y & \frac{\partial}{\partial y}\boldsymbol{v}_y & \frac{\partial}{\partial z}\boldsymbol{v}_y \\ \frac{\partial}{\partial x}\boldsymbol{v}_z & \frac{\partial}{\partial y}\boldsymbol{v}_z & \frac{\partial}{\partial z}\boldsymbol{v}_z \end{pmatrix}$$

Matrix of first-order partial derivatives of vector field $\boldsymbol{v}(\boldsymbol{x}, t)$

Vector and scalar functions

SIEMENS
Ingenuity for life

- Vector function $\boldsymbol{v}(\boldsymbol{x}, t)$
  - Jacobian matrix ("Gradient tensor")

$$\mathbf{J} = \nabla \boldsymbol{v}(\boldsymbol{x}, t) = \begin{pmatrix} \frac{\partial}{\partial x}\boldsymbol{v}_x & \frac{\partial}{\partial y}\boldsymbol{v}_x & \frac{\partial}{\partial z}\boldsymbol{v}_x \\ \frac{\partial}{\partial x}\boldsymbol{v}_y & \frac{\partial}{\partial y}\boldsymbol{v}_y & \frac{\partial}{\partial z}\boldsymbol{v}_y \\ \frac{\partial}{\partial x}\boldsymbol{v}_z & \frac{\partial}{\partial y}\boldsymbol{v}_z & \frac{\partial}{\partial z}\boldsymbol{v}_z \end{pmatrix}$$

Matrix of first-order partial derivatives of vector field $\boldsymbol{v}(\boldsymbol{x}, t)$

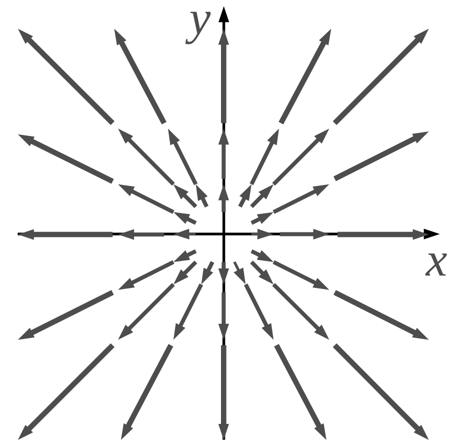
  - Divergence (dot product of vector operator)

$$div\, \boldsymbol{v}(\boldsymbol{x}, t) = \nabla \cdot \boldsymbol{v}(\boldsymbol{x}, t) = \frac{\partial}{\partial x}\boldsymbol{v}_x(\boldsymbol{x}, t) + \frac{\partial}{\partial y}\boldsymbol{v}_y(\boldsymbol{x}, t) + \frac{\partial}{\partial z}\boldsymbol{v}_z(\boldsymbol{x}, t)$$

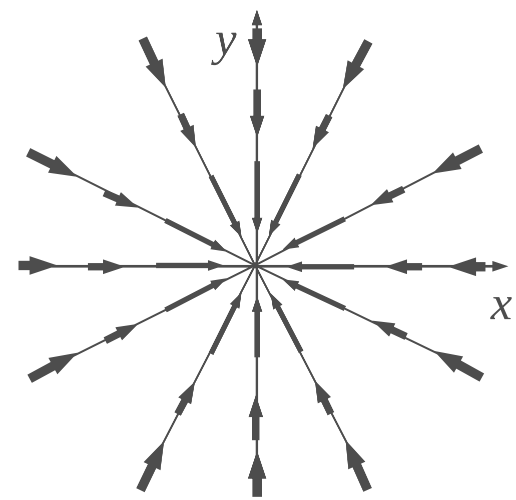
Outflow per unit volume (rate at which density exits a region)

Vector and scalar functions

- Properties of divergence
  - Describes flow into/out of a region
  - $div\ \boldsymbol{v}$ is a scalar field
  - $div\ \boldsymbol{v}(\boldsymbol{x}_0) > 0$: $\boldsymbol{v}$ has a source in $\boldsymbol{x}_0$
  - $div\ \boldsymbol{v}(\boldsymbol{x}_0) < 0$: $\boldsymbol{v}$ has a sink in $\boldsymbol{x}_0$
  - $div\ \boldsymbol{v}(\boldsymbol{x}_0) = 0$: $\boldsymbol{v}$ is source-free in $\boldsymbol{x}_0$



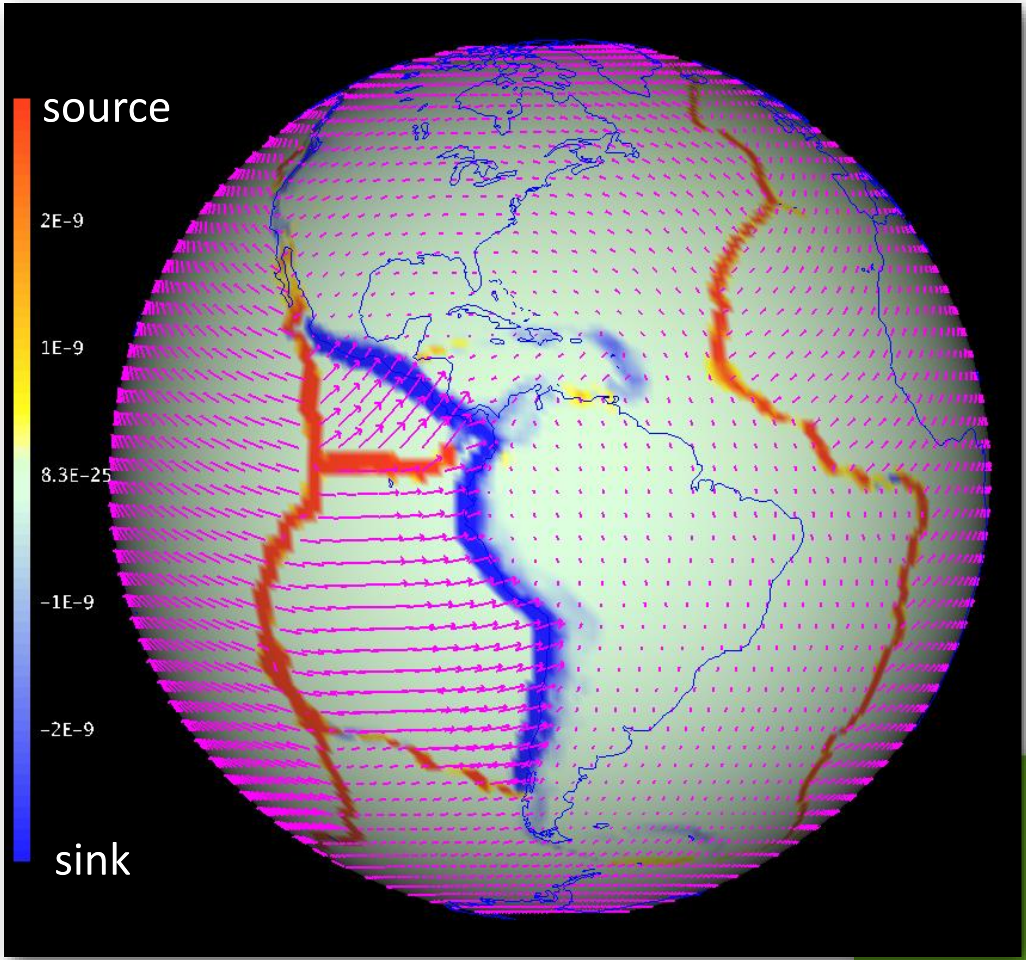
source



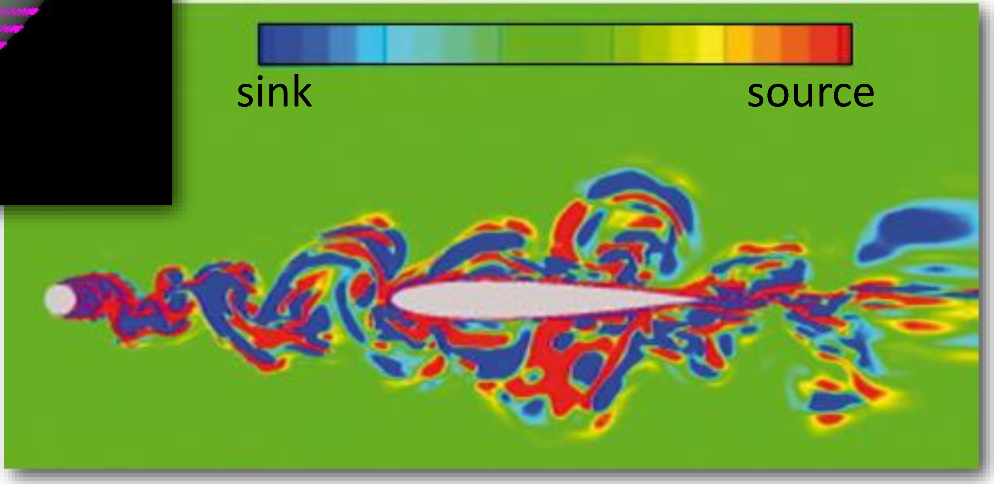
sink

Vector and scalar functions

Color coding of divergence



Tectonic plate motion



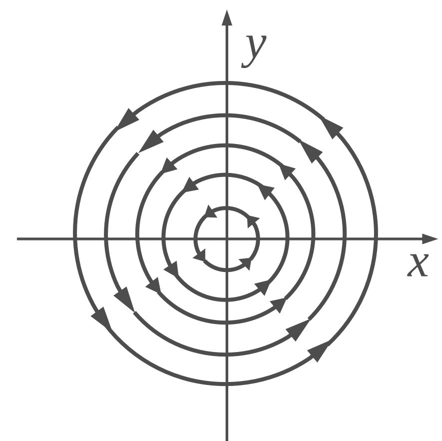
Vector and scalar functions

SIEMENS
Ingenuity for life

- Curl / vorticity

$$curl\, \boldsymbol{v}(\boldsymbol{x},t) = \nabla \times \boldsymbol{v}(\boldsymbol{x},t) = \begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} \end{pmatrix} \times \begin{pmatrix} \boldsymbol{v}_x \\ \boldsymbol{v}_y \\ \boldsymbol{v}_z \end{pmatrix} = \begin{pmatrix} \frac{\partial}{\partial y}\boldsymbol{v}_z - \frac{\partial}{\partial z}\boldsymbol{v}_y \\ \frac{\partial}{\partial z}\boldsymbol{v}_x - \frac{\partial}{\partial x}\boldsymbol{v}_z \\ \frac{\partial}{\partial x}\boldsymbol{v}_y - \frac{\partial}{\partial y}\boldsymbol{v}_x \end{pmatrix}$$

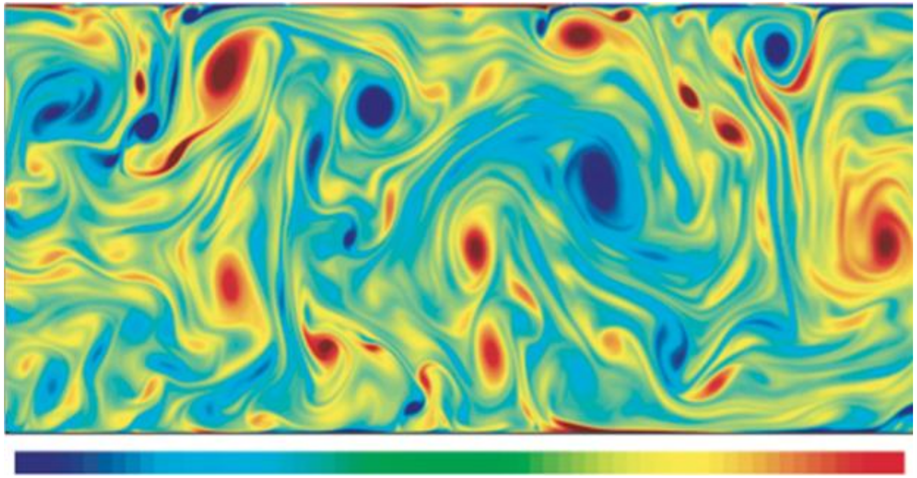
  - Curl is a vector-valued quantity
  - Describes vortex characteristics in flow
  - A measure of how fast the flow rotates (magnitude of the curl), and
  - … around which axis it rotates (direction of the curl)



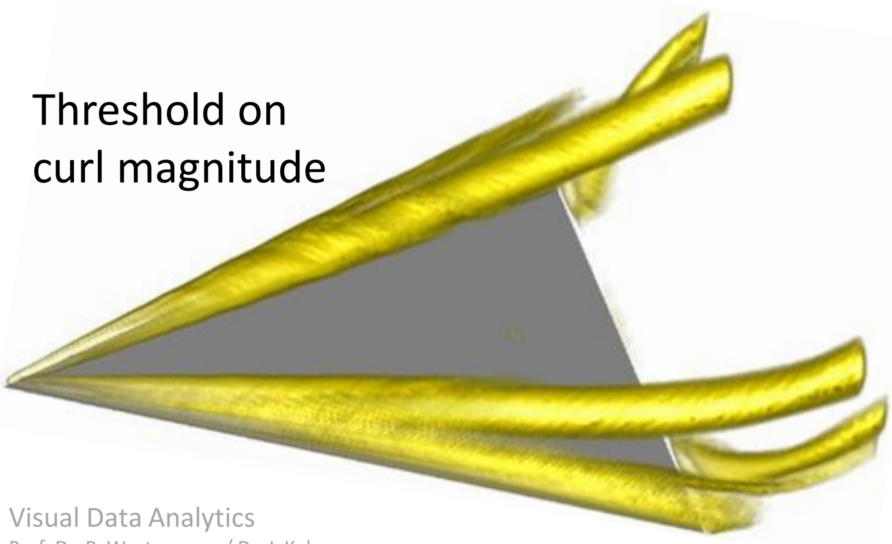
Vector and scalar functions

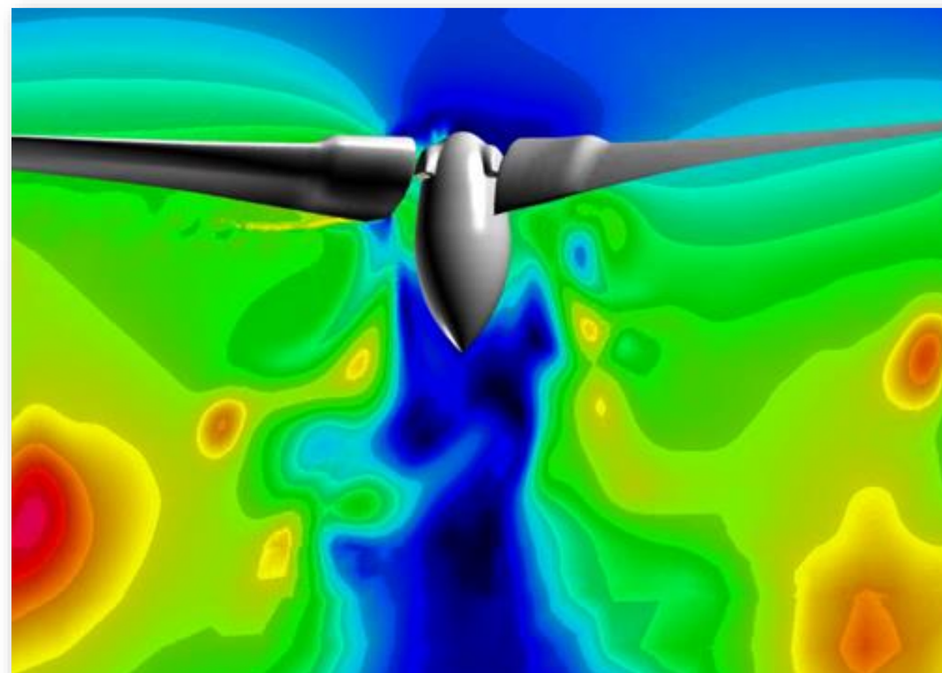
SIEMENS
Ingenuity for life

- Color coding of curl direction/magnitude



Threshold on
curl magnitude



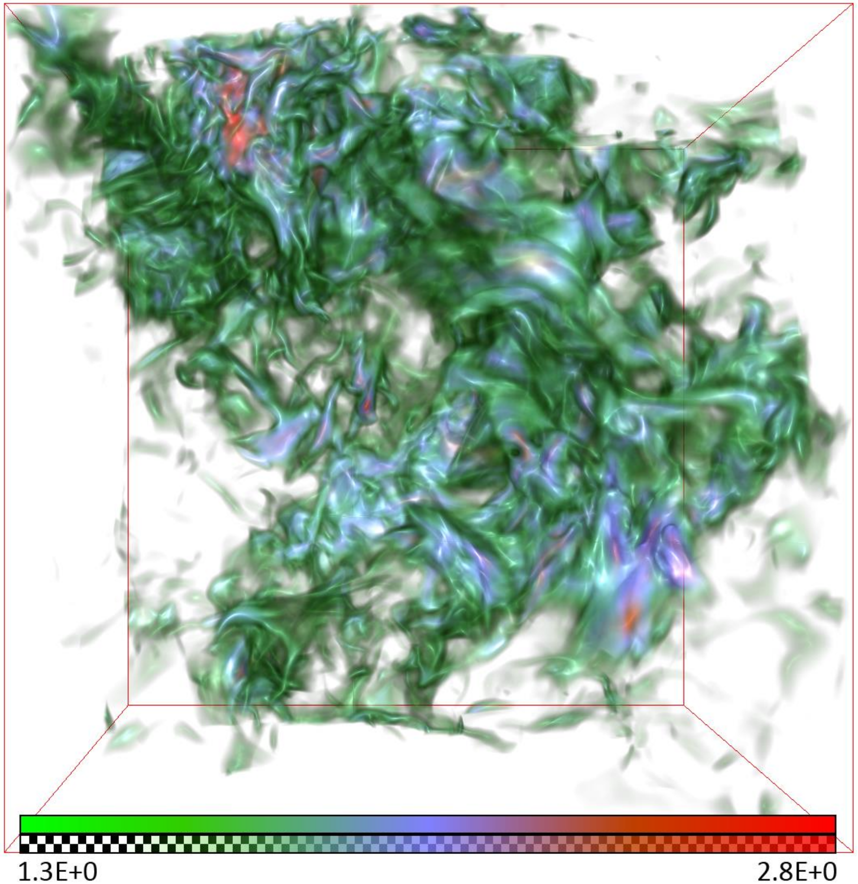


Curl magnitude

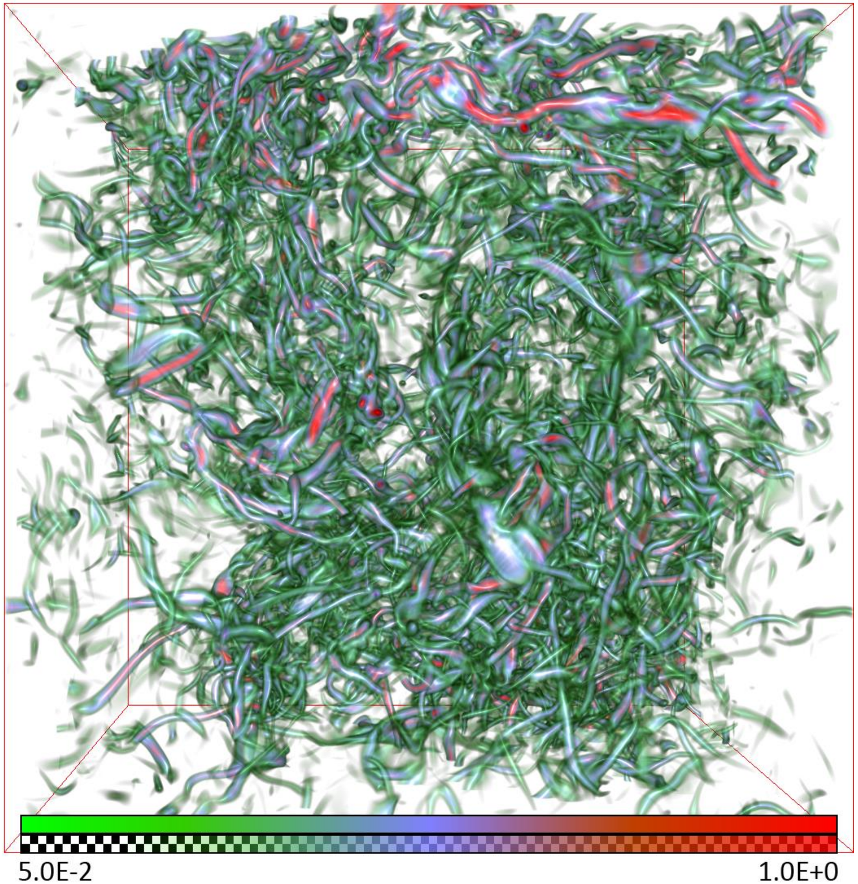
Vector and scalar functions

SIEMENS
Ingenuity for life

- Direct volume rendering (DVR) of turbulent flow



DVR of velocity magnitude

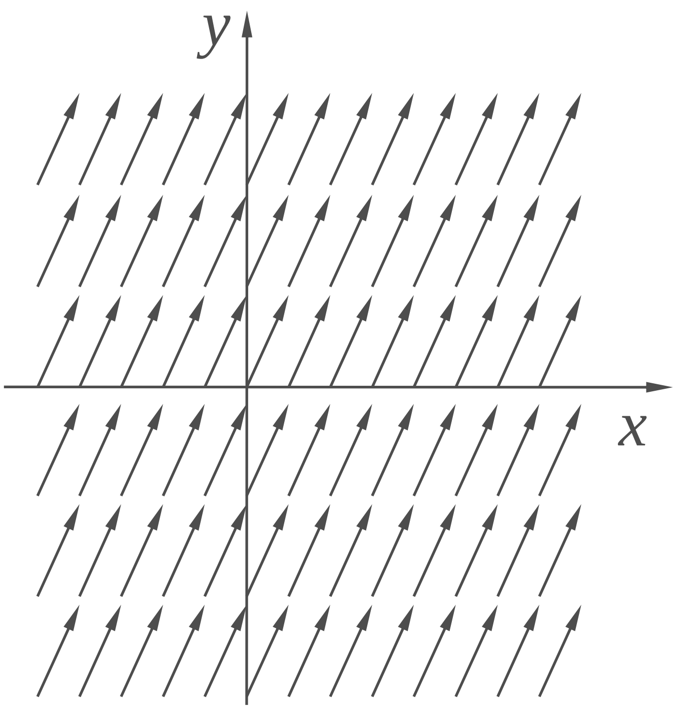


DVR of curl magnitude

Vector and scalar functions

SIEMENS
Ingenuity for life

- Example



A constant 2D vector function

$$H_1(x,y,z) = \begin{pmatrix} 1 \\ 2 \\ 0 \end{pmatrix}$$

$$J_{H_1}(x,y,z) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

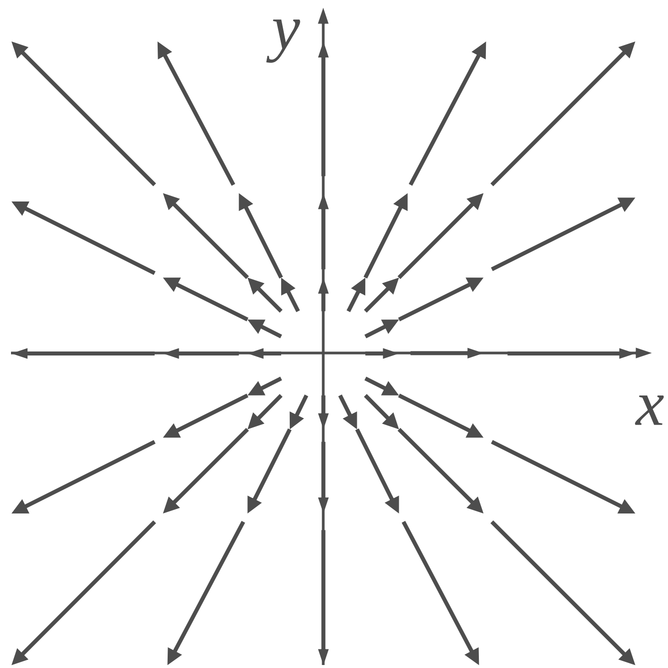
No velocity change in any direction

$$div(H_1)(x,y,z) = 0$$

$$curl(H_1)(x,y,z) = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

Vector and scalar functions

- Example



The 2D identity vector function

$$\text{Identity}: H_2(x, y, z) = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

Velocity increases with distance

$$J_{H_2}(x, y, z) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

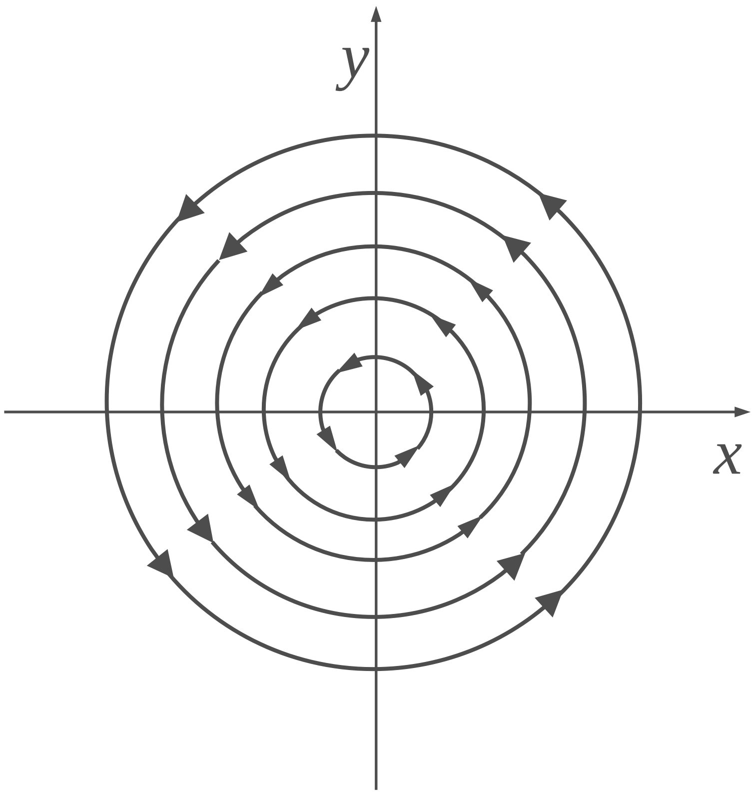
$$div(H_2)(x, y, z) = 3$$

$$curl(H_2)(x, y, z) = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

Vector and scalar functions

SIEMENS
Ingenuity for life

- Example



A 2D curl field

$$\text{Curl field}: H_3(x,y,z) = \begin{pmatrix} -y \\ x \\ 0 \end{pmatrix}$$

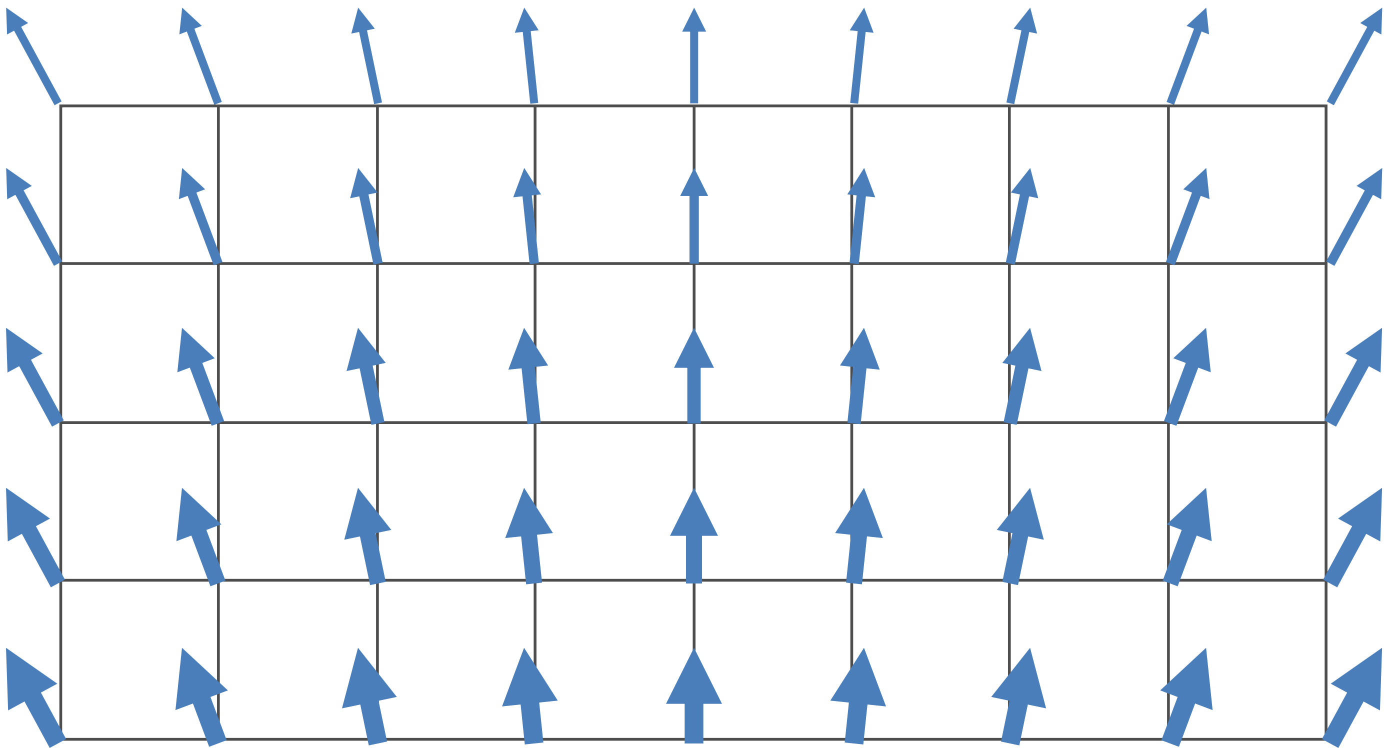
$$J_{H_3}(x,y,z) = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$div(H_3)(x,y,z) = 0$$

$$curl(H_3)(x,y,z) = \begin{pmatrix} 0 \\ 0 \\ 2 \end{pmatrix}$$

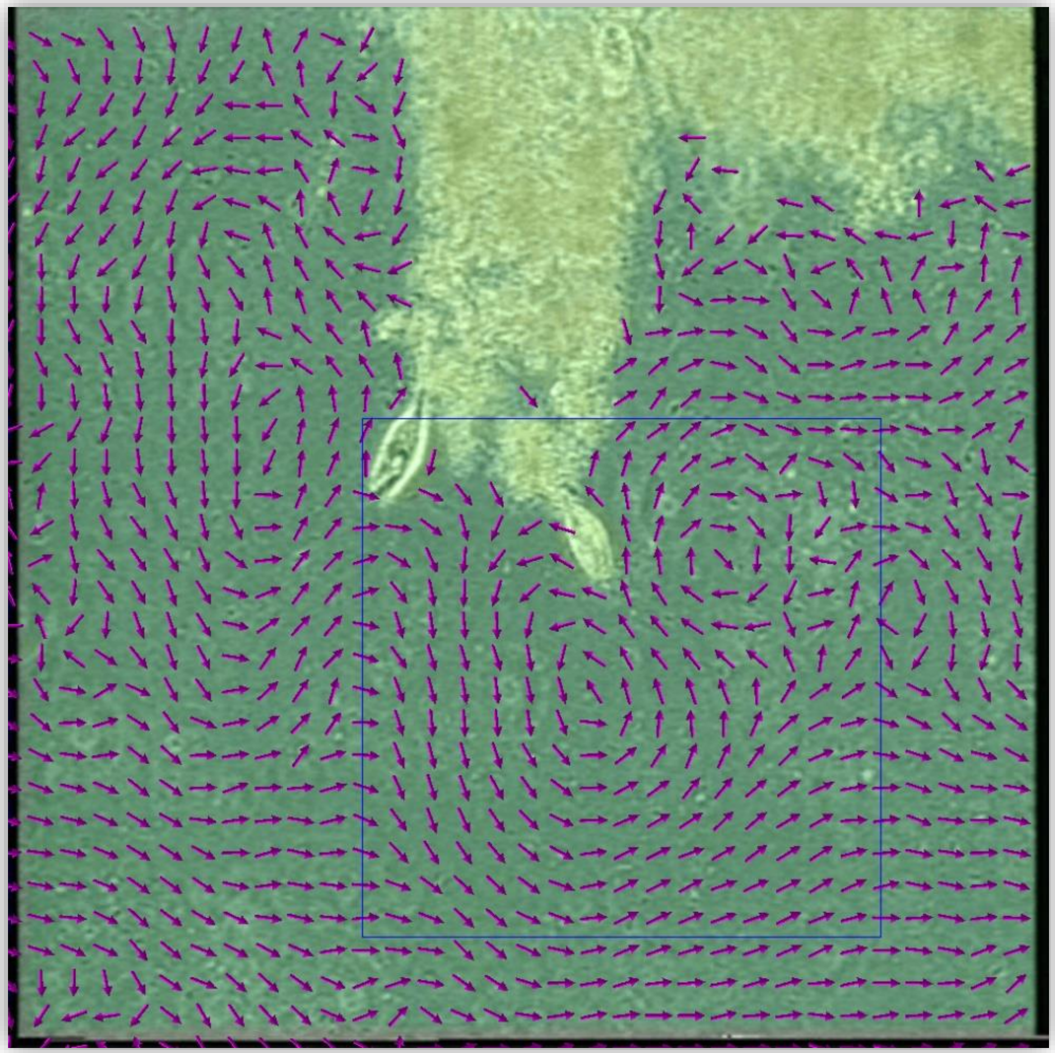
Flow visualization

- Glyphs
  - Visualize local features of the vector field
  - Map vector or curl to arrow glyphs



Flow visualization with arrows

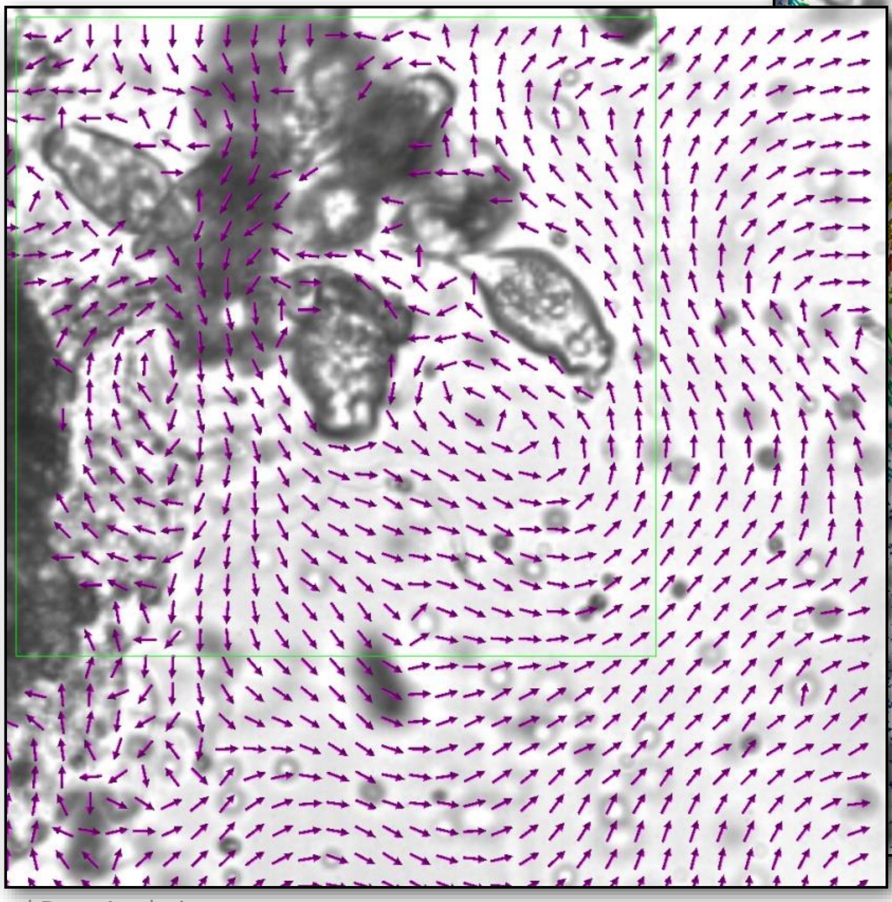
- Vector per grid point pointing into the flow direction

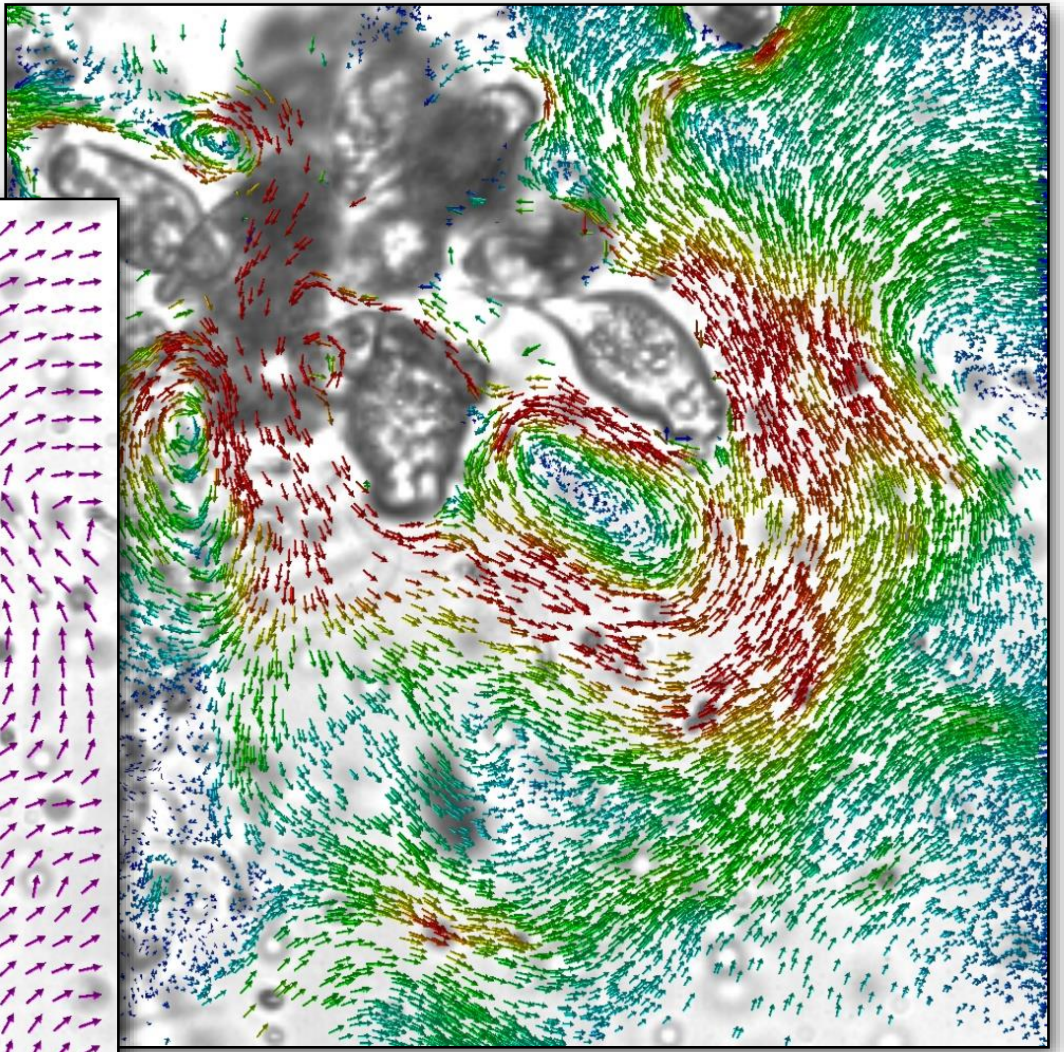


Flow visualization with arrows

SIEMENS
Ingenuity for life

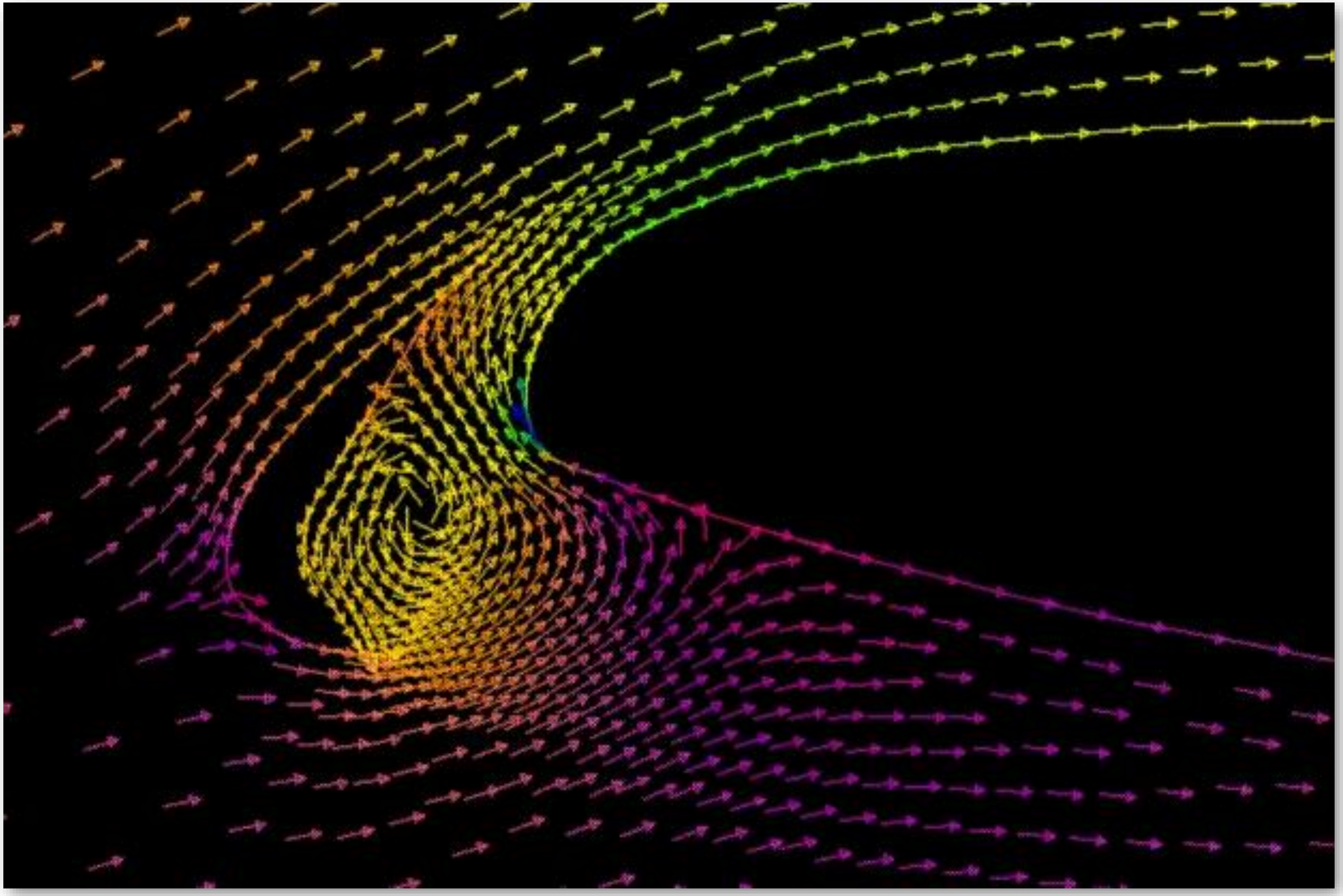
- Use arrow length and/or color to highlight special regions





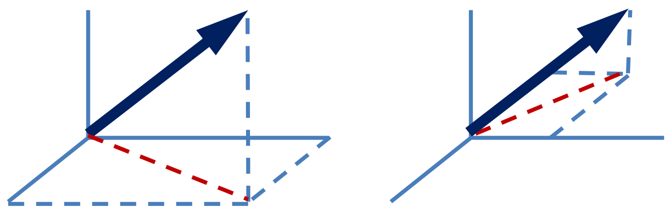
Flow visualization with arrows

SIEMENS
Ingenuity for life



Arrows in 3D

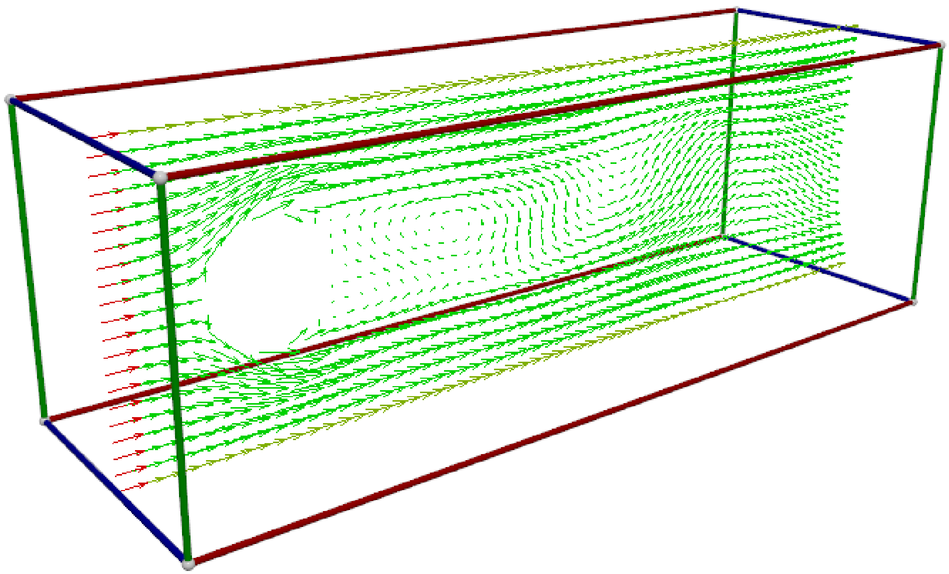
- Advantages
  - Simple
  - 3D effects
- Disadvantages
  - Ambiguity
  - Difficult spatial perception (1D-objects in 3D)
  - Inherent occlusion effects
  - Poor results if magnitude of velocity varies significantly and changes rapidly

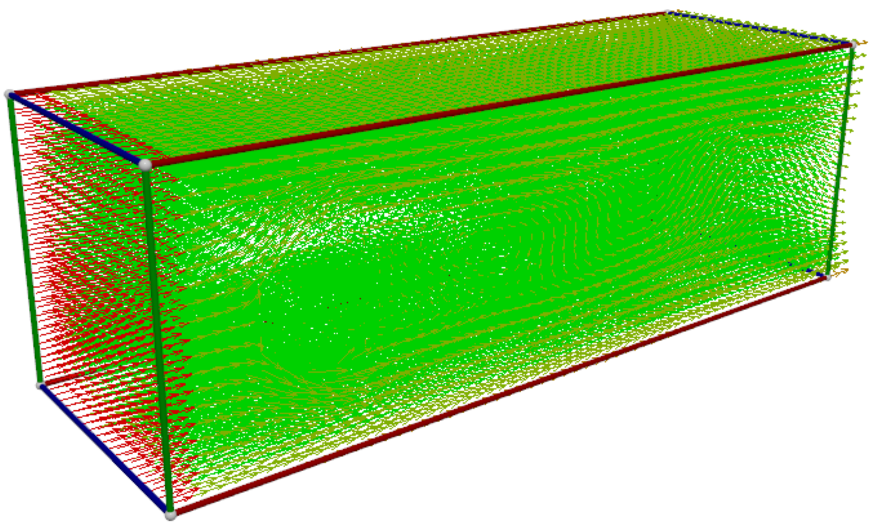


Use 3D arrows of constant length and color code magnitude

Arrows in 3D

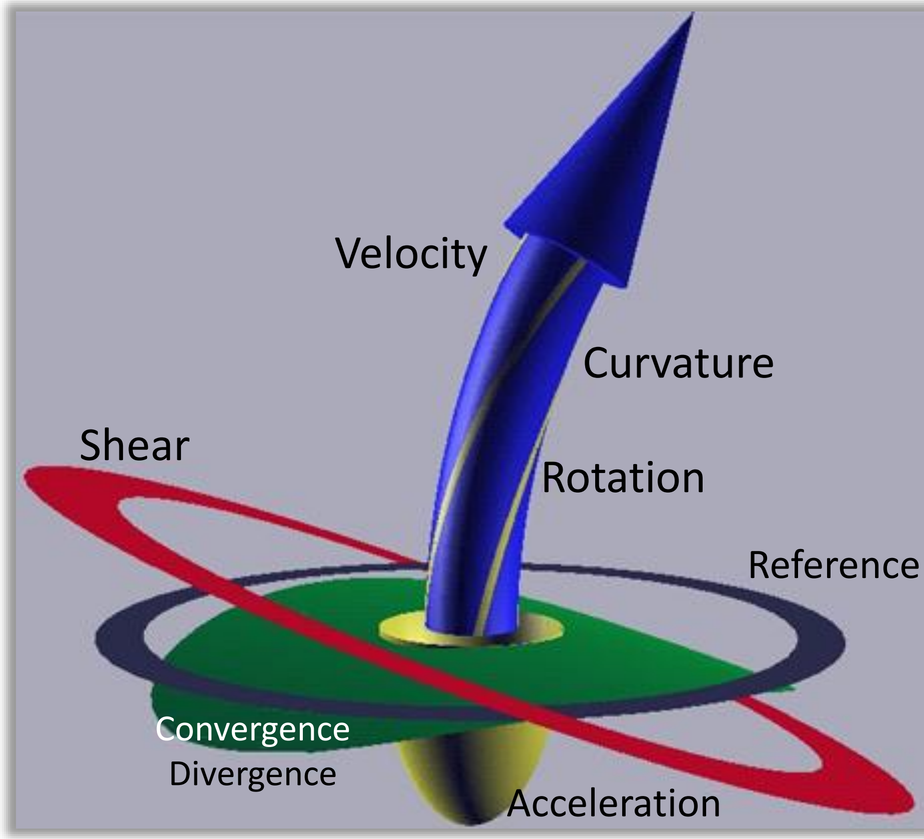
- Compromise
  - Arrows only in slices





Flow visualization with glyphs

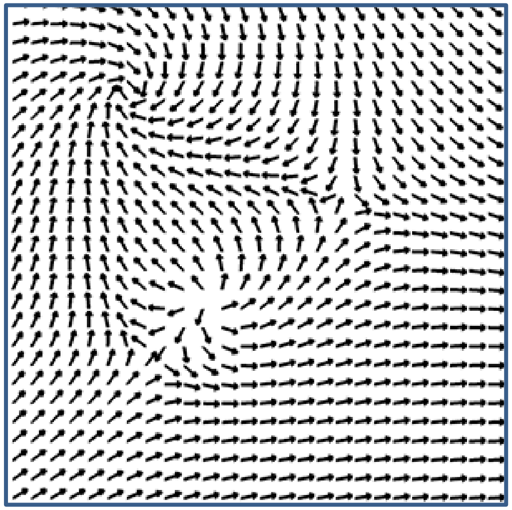
- Glyphs
  - Can visualize more features of vector field



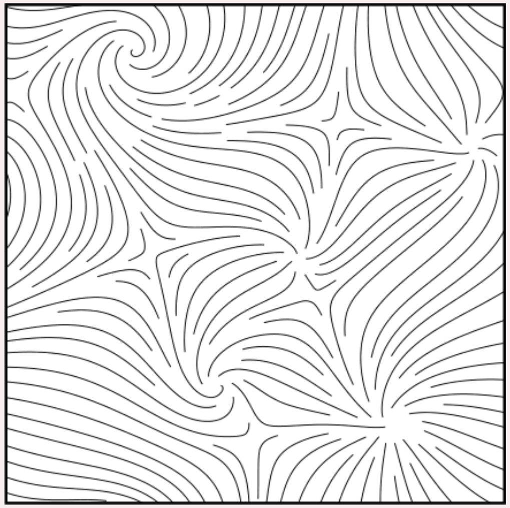
[de Leeuw and van Wijk 93]

Flow visualization – Approaches

SIEMENS
Ingenuity for life

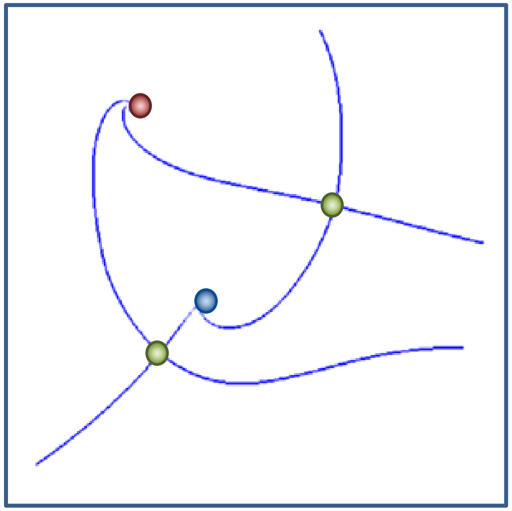


Direct flow visualization (arrows, color coding, …)

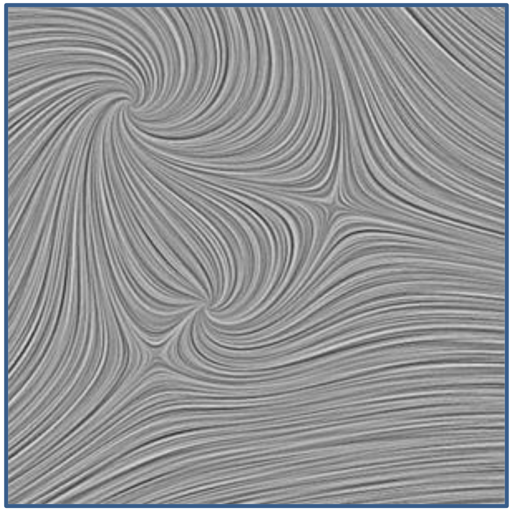


Geometric flow visualization

- Use intermediate representation (vector-field integration over time)
- Visualization of temporal evolution
- Stream lines, path lines, streak lines



Sparse (feature-based) vis.

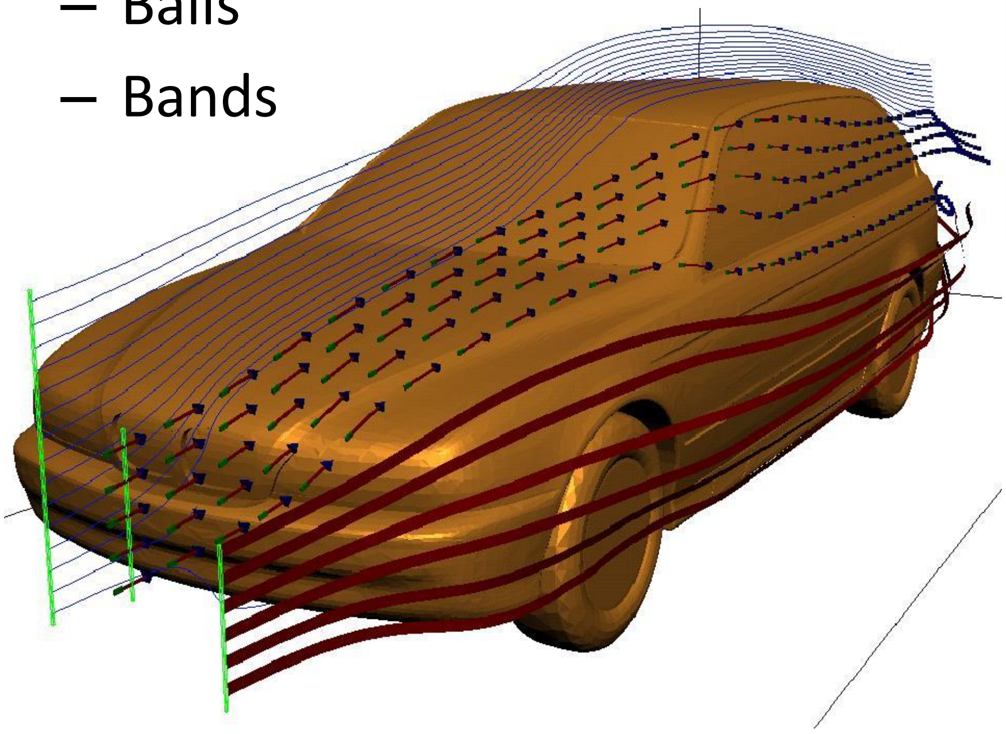


Dense (texture-based) vis.

Characteristic Lines

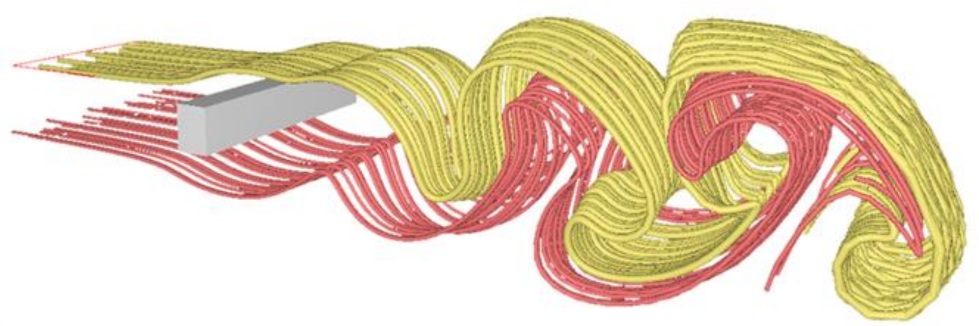
SIEMENS
Ingenuity for life

- Basic idea: trace particles along characteristic trajectories
- Map trajectories to
  - Particles
  - Lines
  - Balls
  - Bands



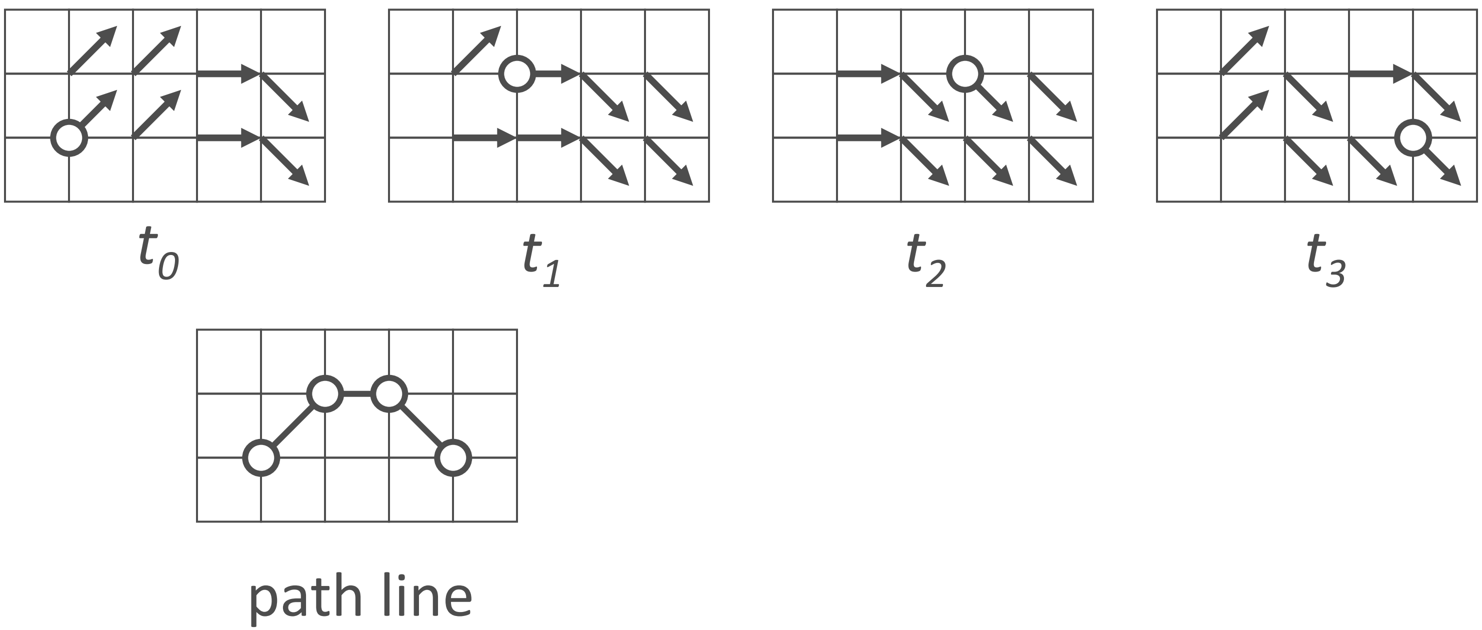
Characteristic Lines

- Types of characteristic lines
  - Stream lines: trajectories of massless particles in a "frozen" (steady) vector field
  - Path lines: trajectories of massless particles in (unsteady/time-varying) flow
  - Streak lines: trace of dye that is continuously released into (unsteady/time-varying) flow at a fixed position



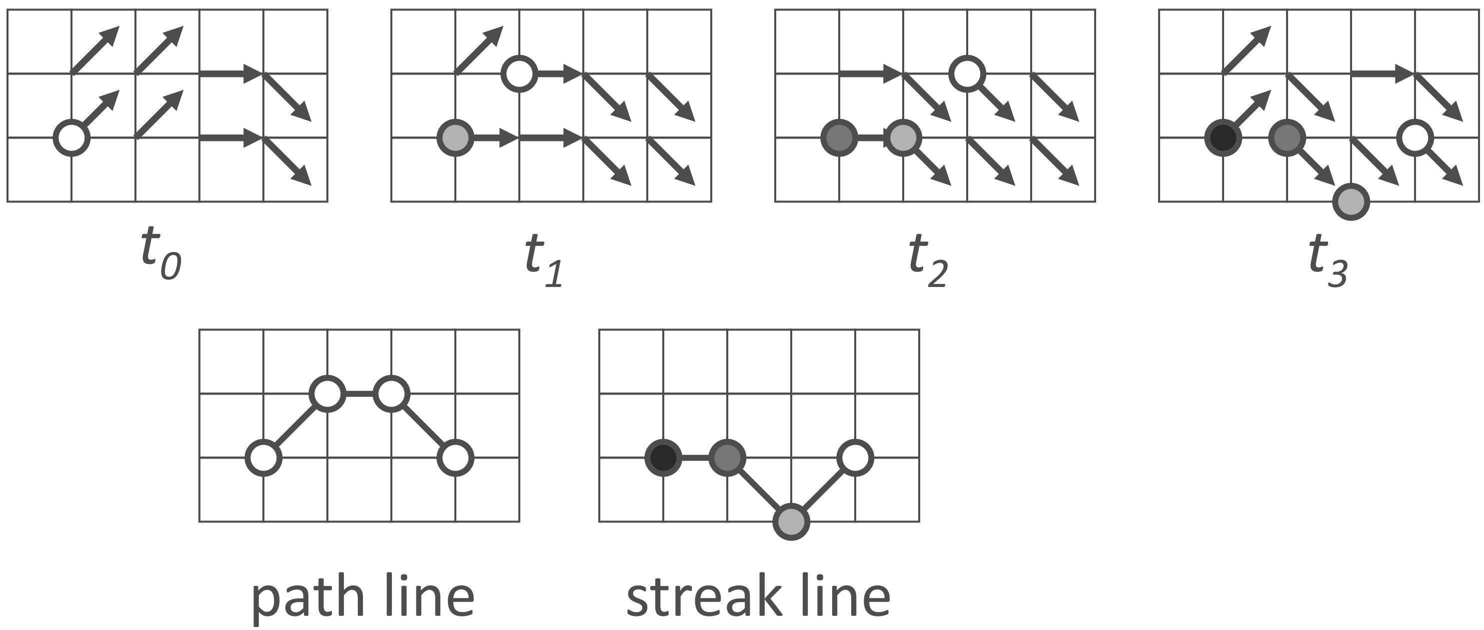
Characteristic Lines

- Path lines
  - Follow one particle through time and space



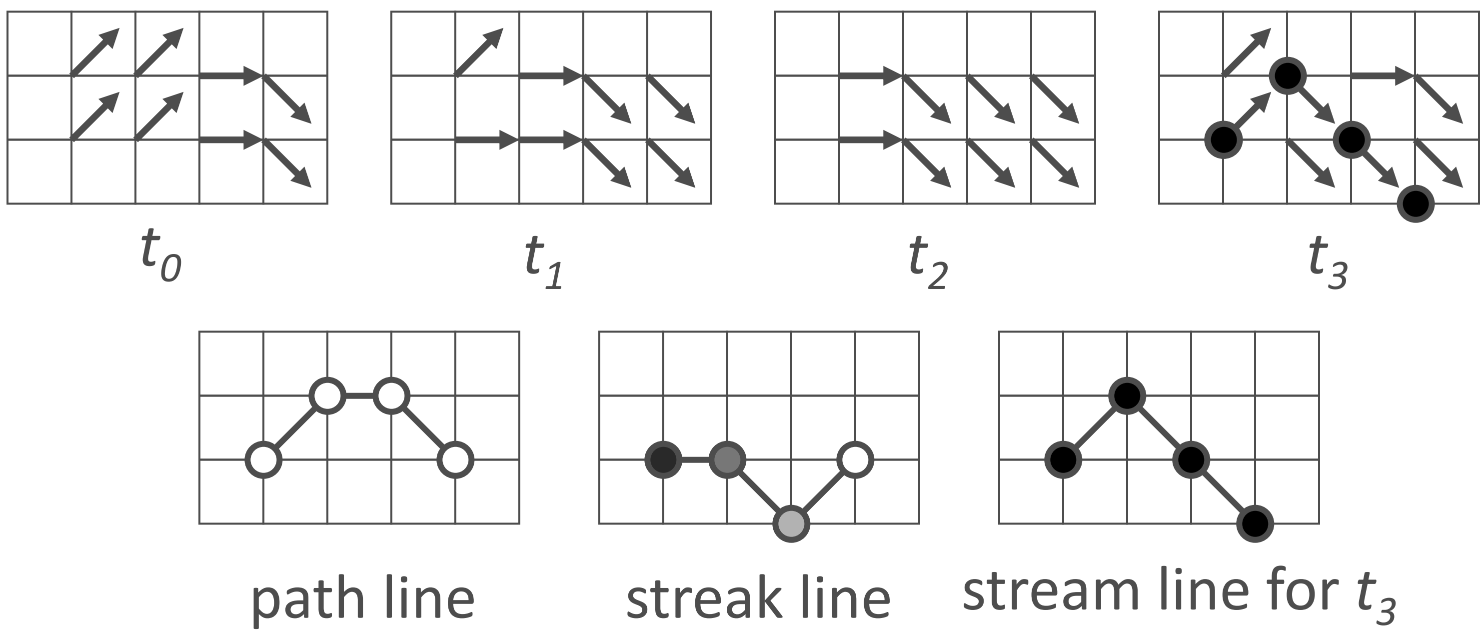
Characteristic Lines

- Streak lines
  - Connect all particles that started at the same seed point
    - A new particle is continuously injected at the same seed point
    - All existing particles are advected & connected (from youngest to oldest)



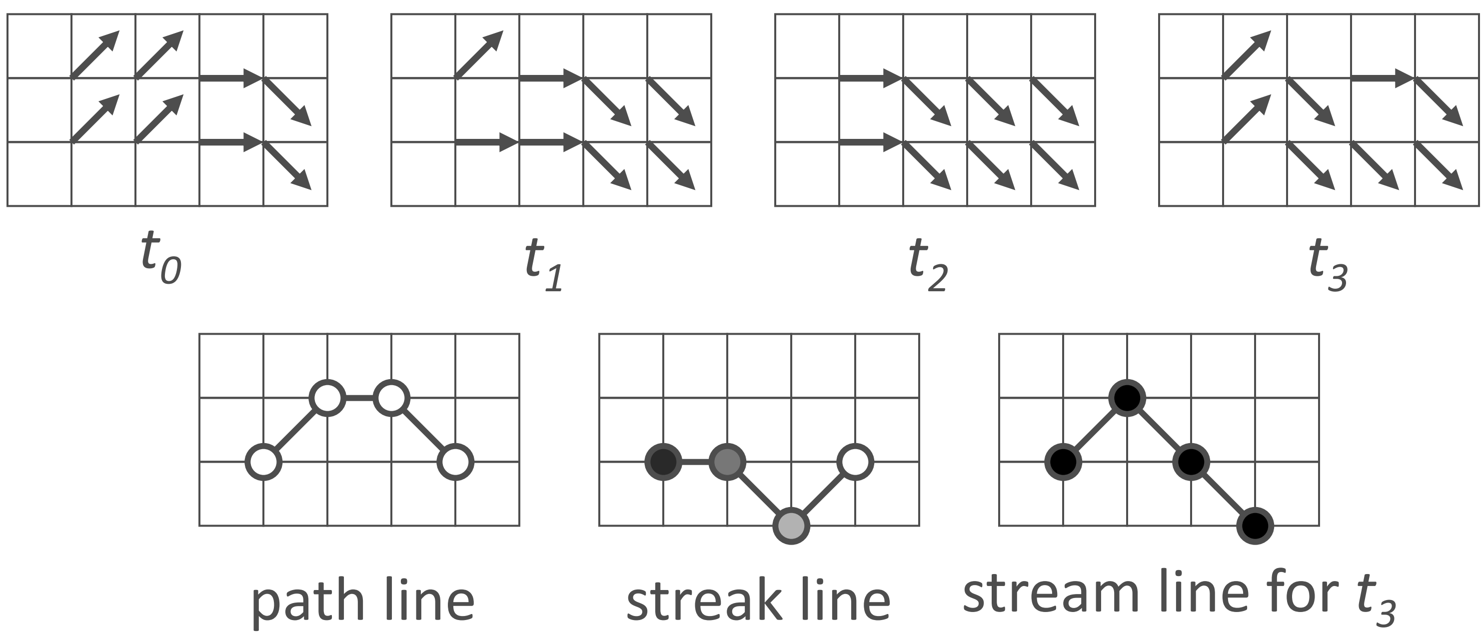
Characteristic Lines

- Stream lines
  - Trajectories of massless particles at one time step



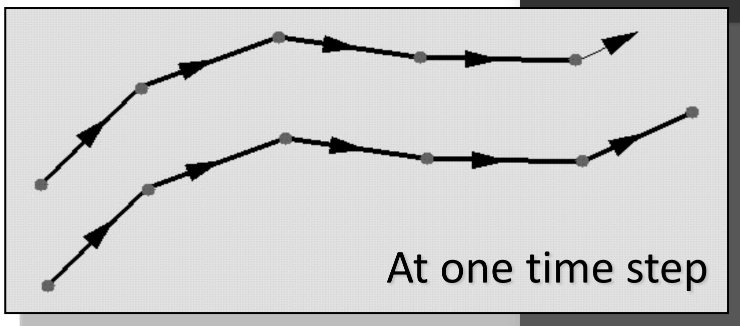
Characteristic Lines

- Comparison of path lines, streak lines, and stream lines
  - Identical for steady flows

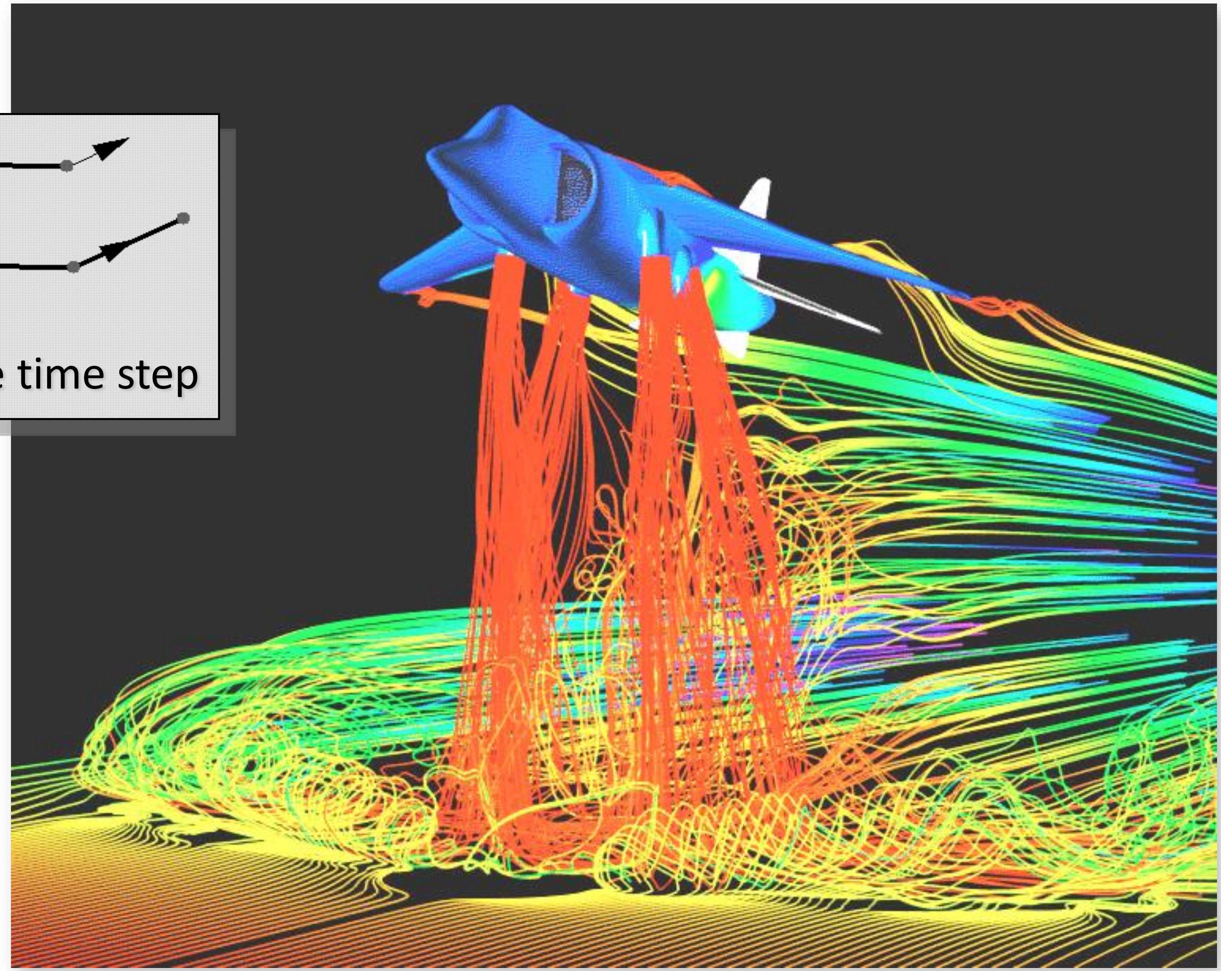


Mapping Based on Particle Tracing

- Stream lines

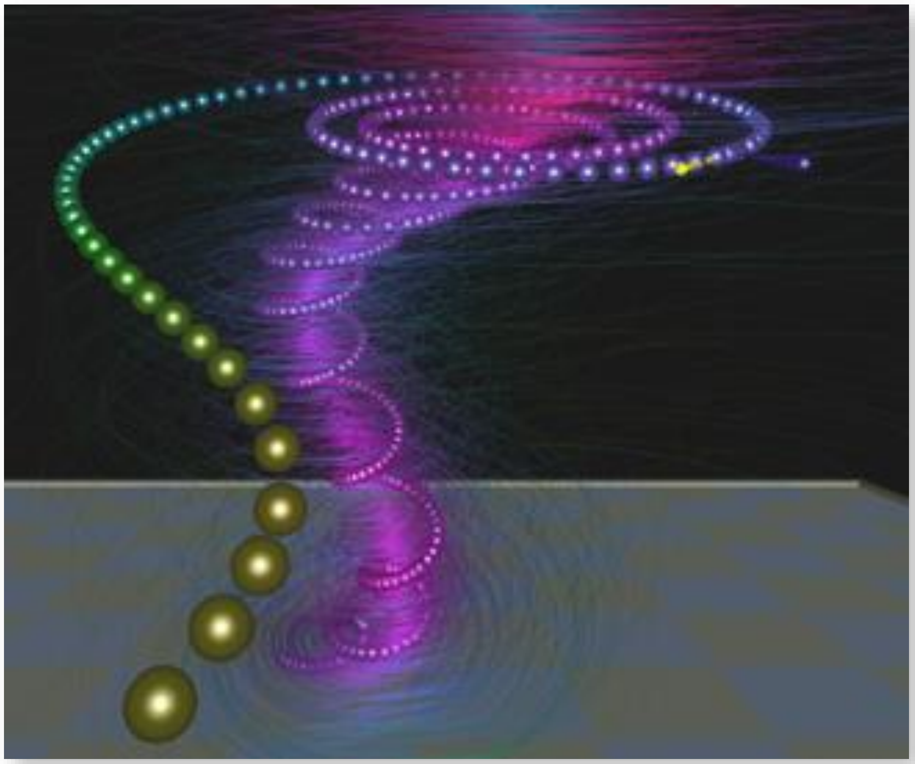


Simple colored
stream lines

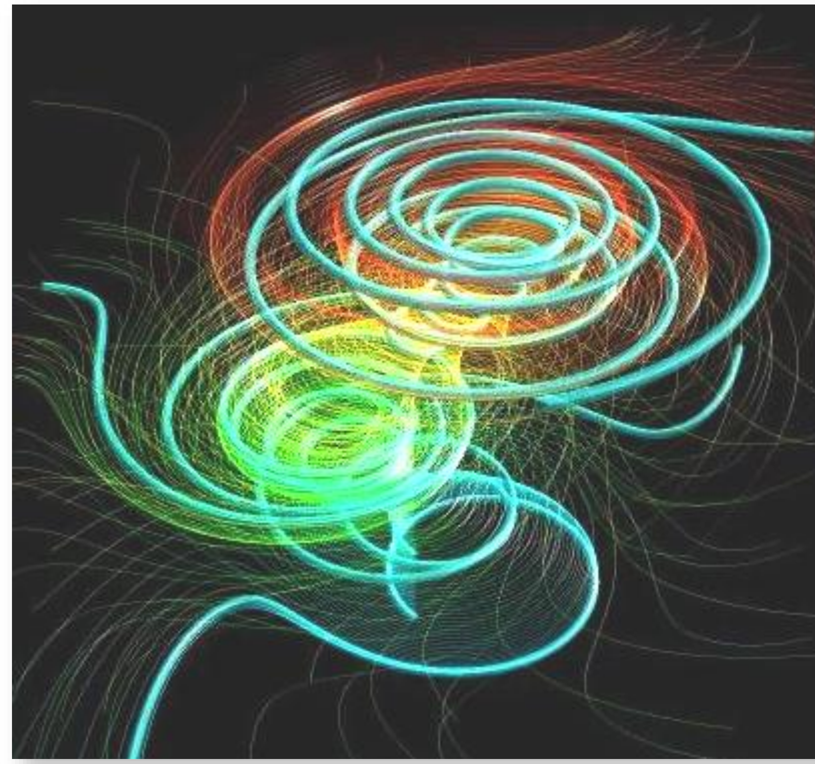


Mapping Based on Particle Tracing

- Stream lines



Stream balls & illuminated stream lines



Stream tubes

Particle tracing, bir madde parçacığının bir akışkan içinde nasıl hareket ettiğini takip etmek için kullanılan bir yöntemdir. Bu yöntem, bir parçacığın başlangıç koordinatlarını belirleyerek, parçacığın türevlerini hesaplayarak ve bu türevleri zamana göre çözerek parçacığın yolunu tahmin etmeye çalışır. Bu yöntem, hareket eden parçacıkların gelecekteki hareketlerini tahmin etmek için kullanılabilir.

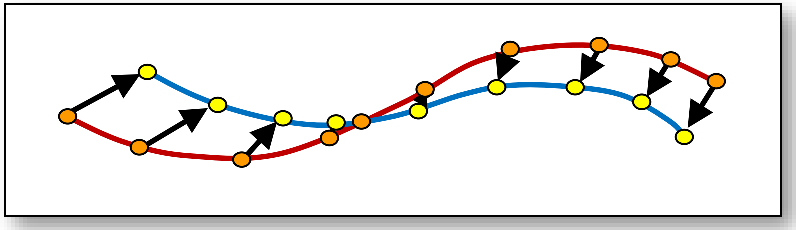
Streamlines, bir akışkan içinde hava veya su gibi maddelerin hareketini göstermek için kullanılan bir yöntemdir. Streamlines, bir akışkanın hızının yönünü gösterir ve bir akışkanın hareketini anlamaya yardımcı olur. Streamlines, bir akışkanın hareketi hakkında bilgi verirken, particle tracing ise bir madde parçacığının hareketi hakkında bilgi verir.

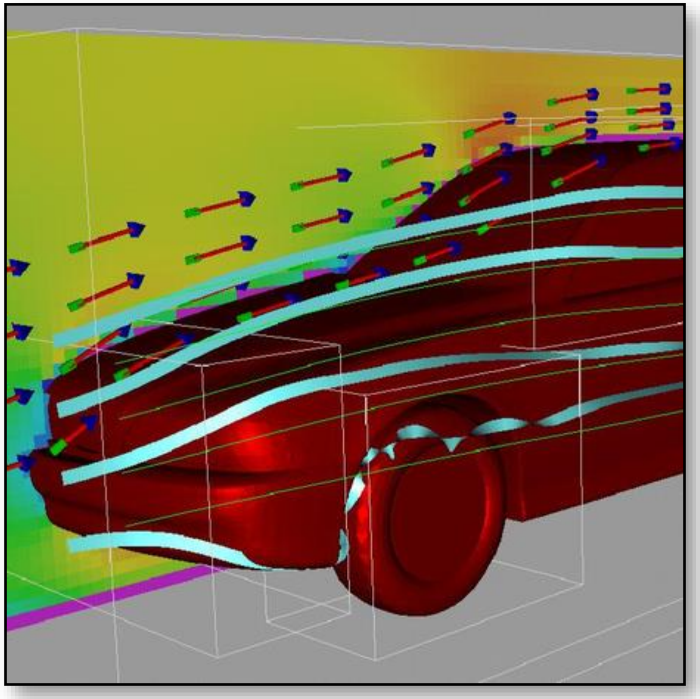
Farklı olarak, streamlines akışkanın hareketini gösterirken, particle tracing ise bir parçacığın hareketini takip eder. Ayrıca, particle tracing yalnızca bir parçacığın hareketini takip ederken, streamlines tüm akışkanın hareketini gösterir.

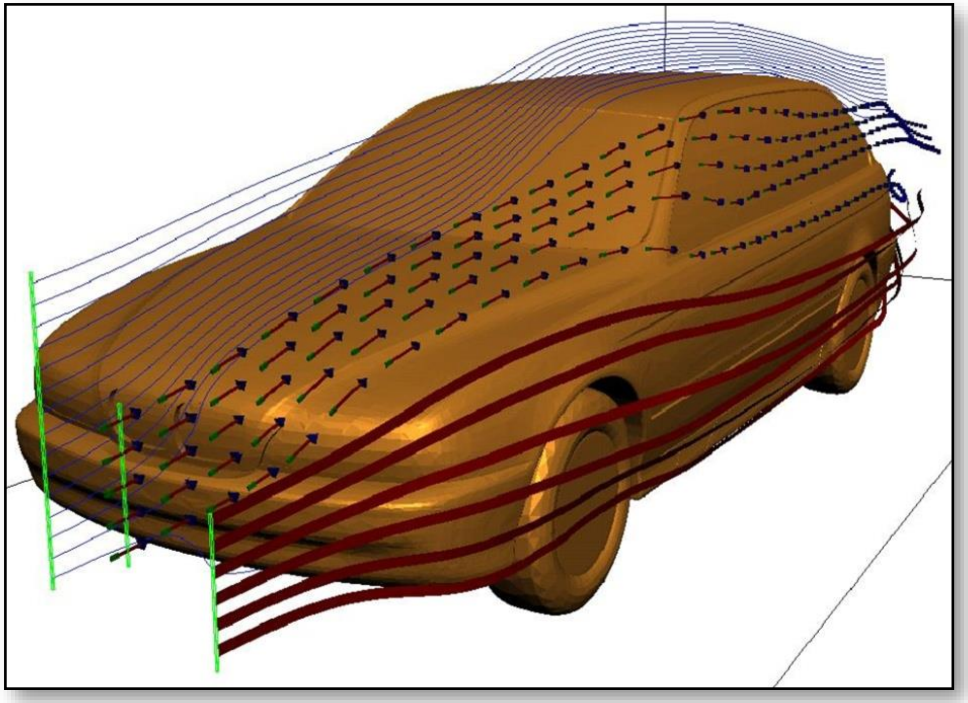
Mapping Based on Particle Tracing

SIEMENS
Ingenuity for life

- Stream ribbons (flow oriented)
  - We liked to see places where the flow twists (vortices)
  - Trace two close-by particles
    (keep distance constant)
  - Or rotate band according to curl



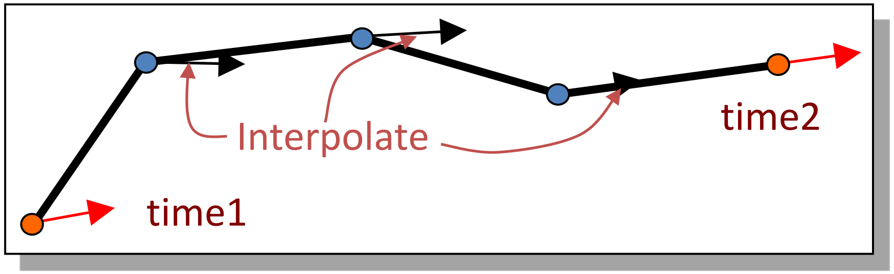


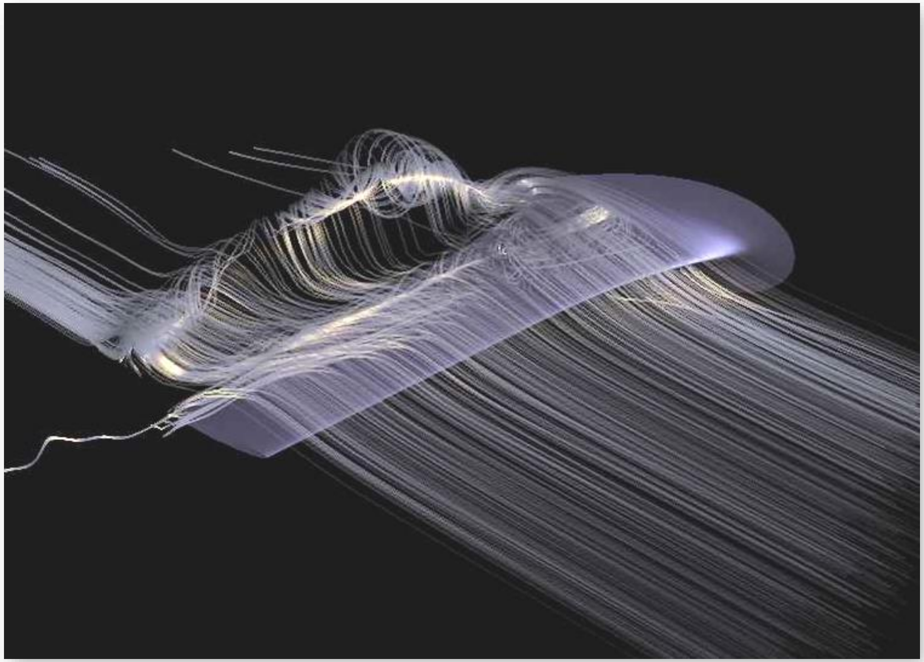


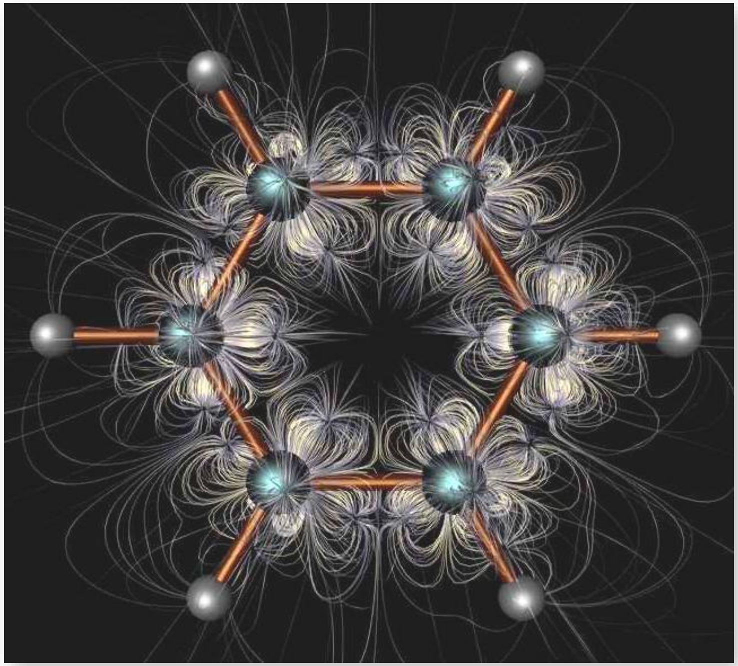
Mapping Based on Particle Tracing

SIEMENS
Ingenuity for life

- Path lines
  - Follow one particle through time and space
  - Illuminated 3D curves improves 3D perception

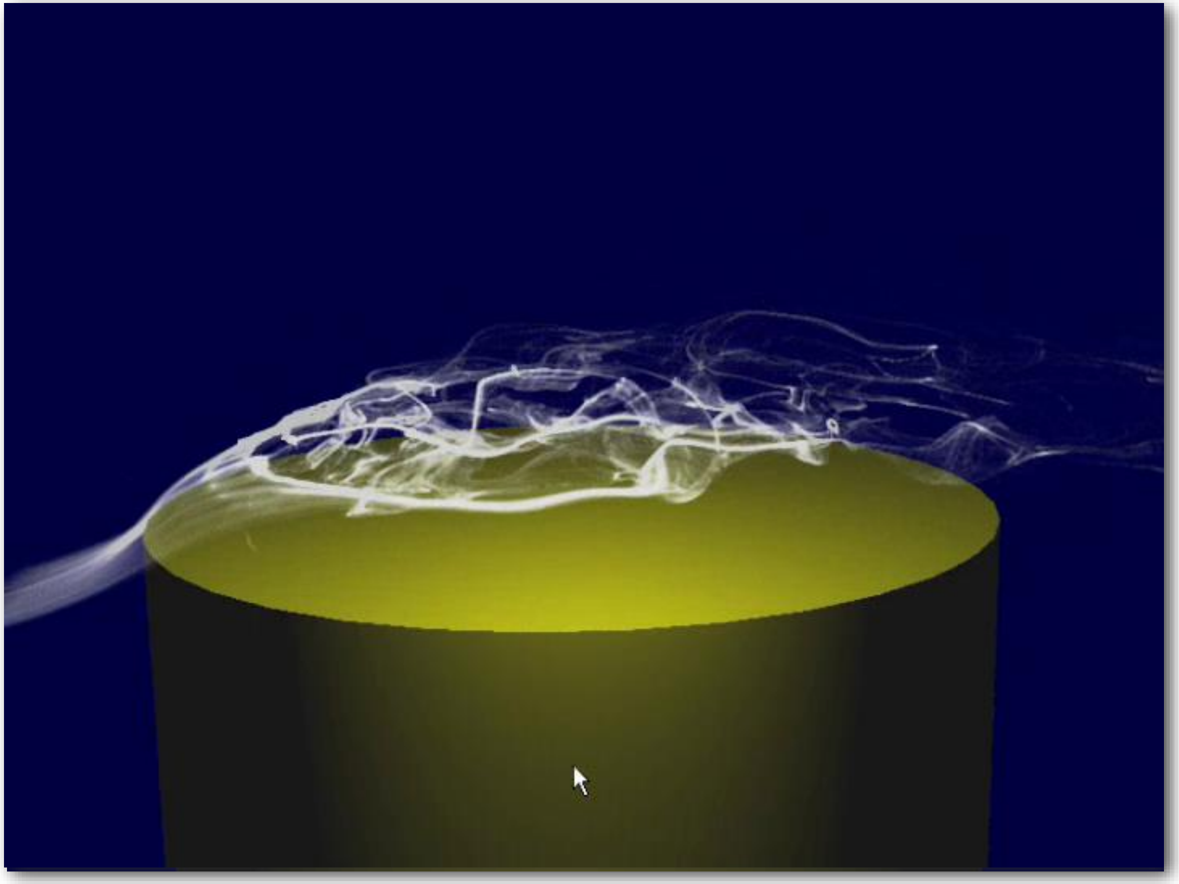






Mapping Based on Particle Tracing

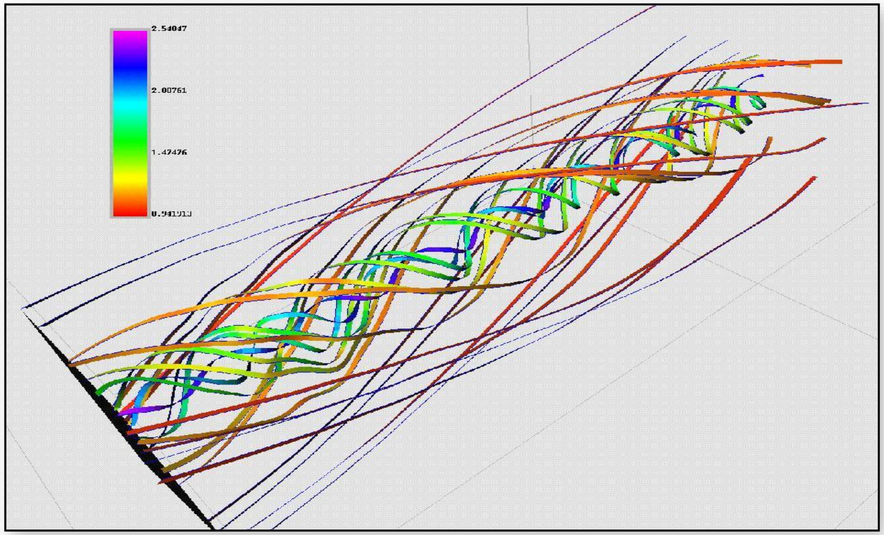
- Particle visualization in unsteady flow

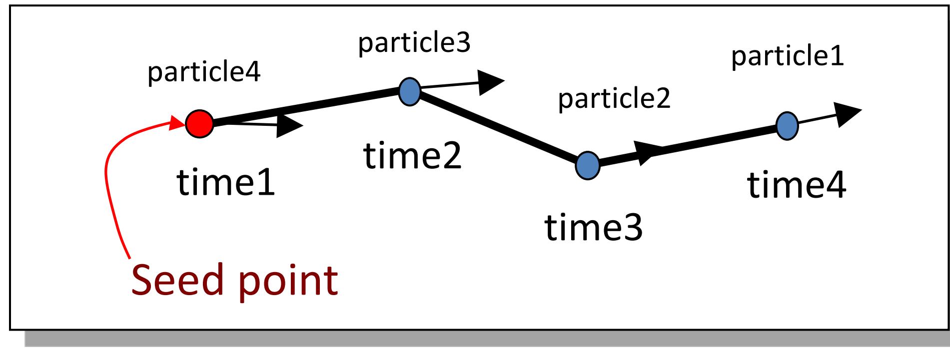


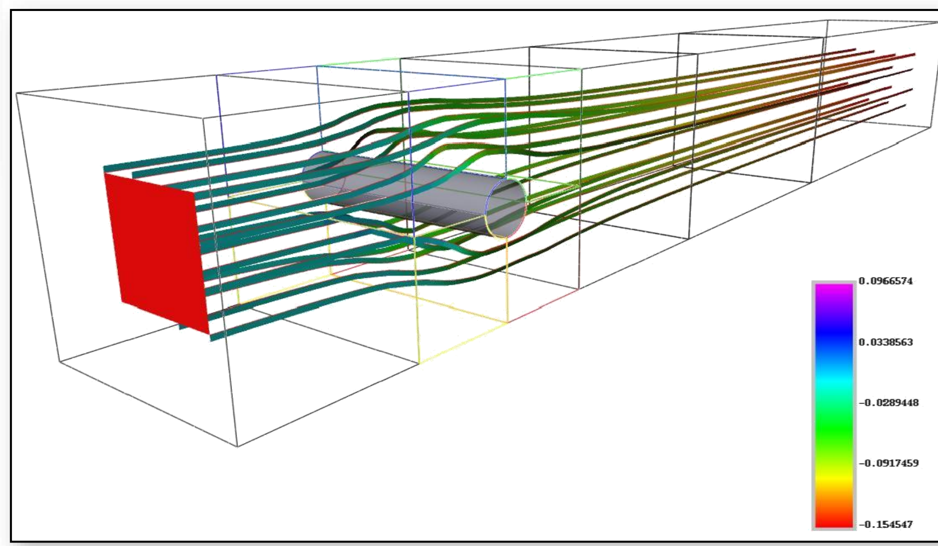
Mapping Based on Particle Tracing

SIEMENS
Ingenuity for life

- Streak lines
  - A new particle is continuously injected at the same seed point
  - Existing particles are advected & connected (from youngest to oldest)



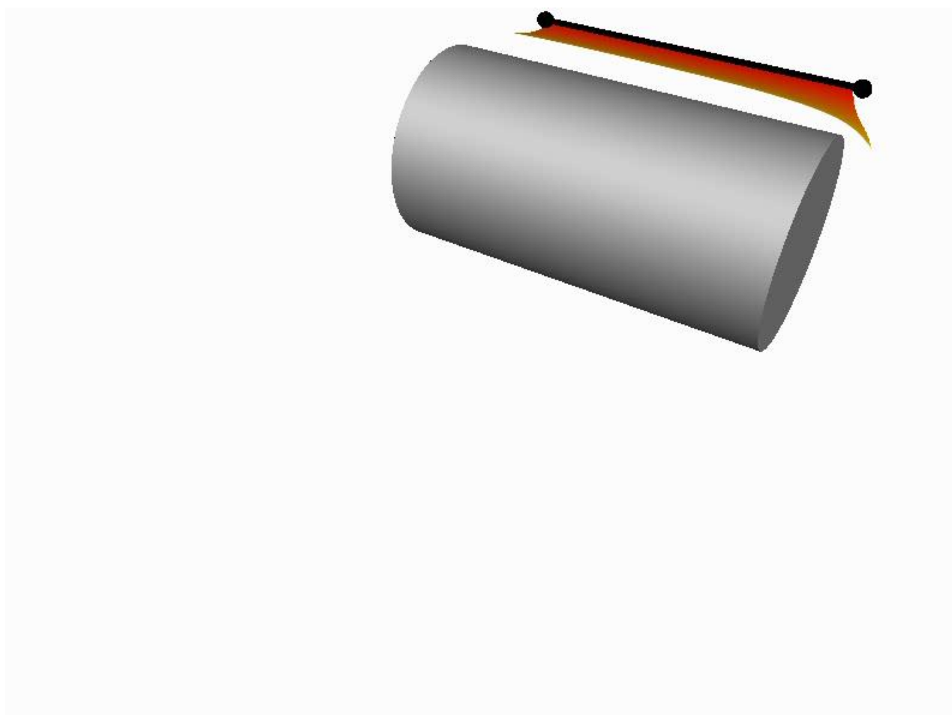




Mapping Based on Particle Tracing

SIEMENS Ingenuity for life

- Streak surface
  - Simultaneously release particles along a seeding structure (line) and connect all them to form a surface

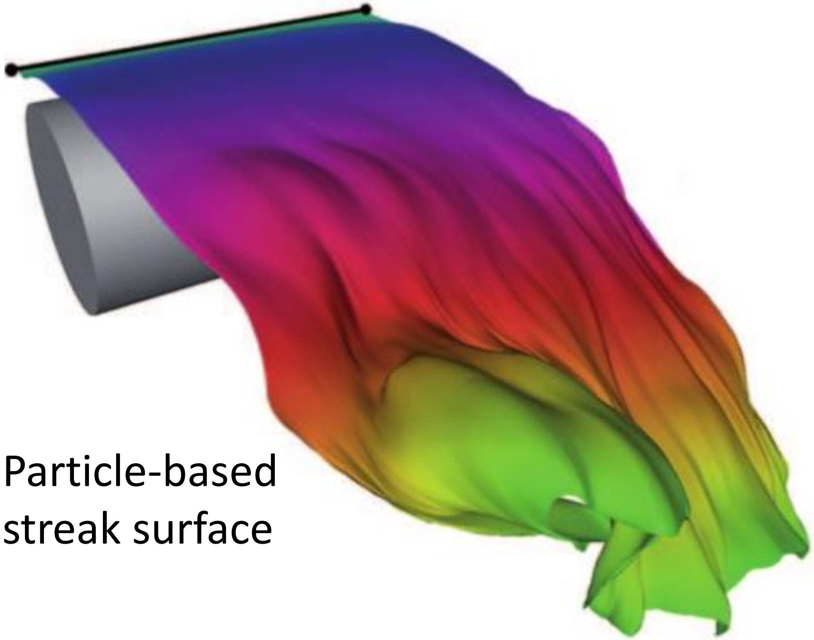


[Bürger et al. 09]

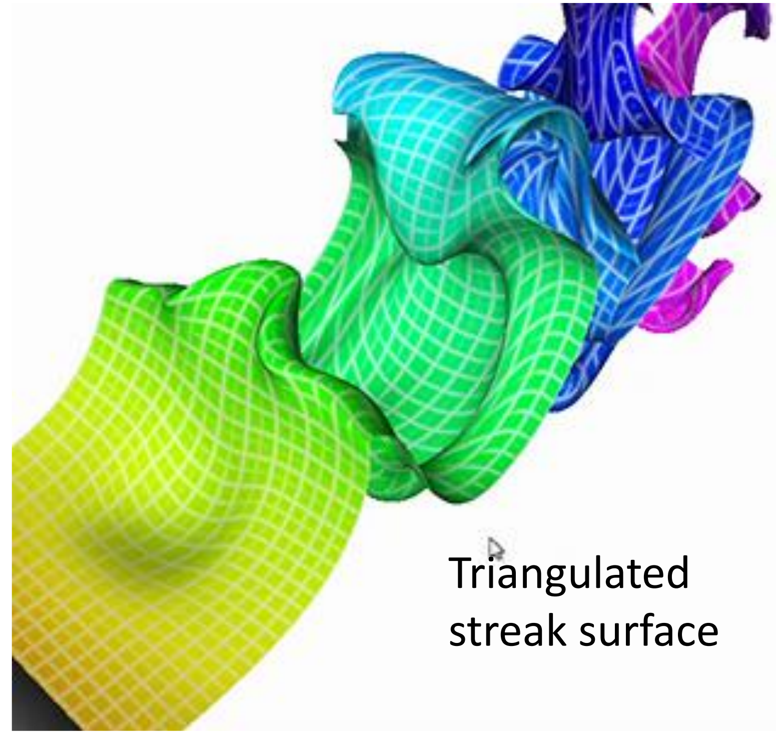
Mapping Based on Particle Tracing

SIEMENS
Ingenuity for life

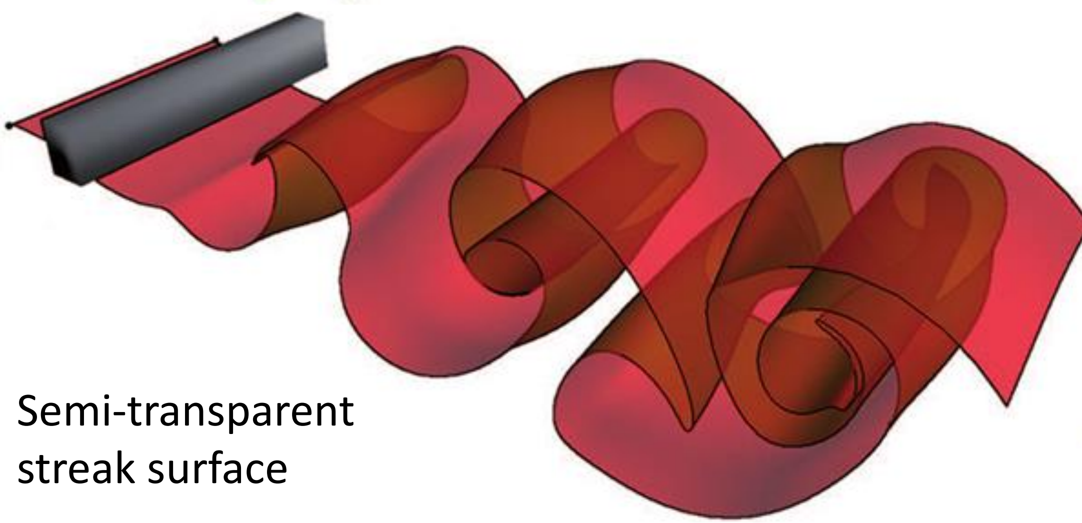
- Streak surface



Particle-based
streak surface



Triangulated
streak surface



Semi-transparent
streak surface

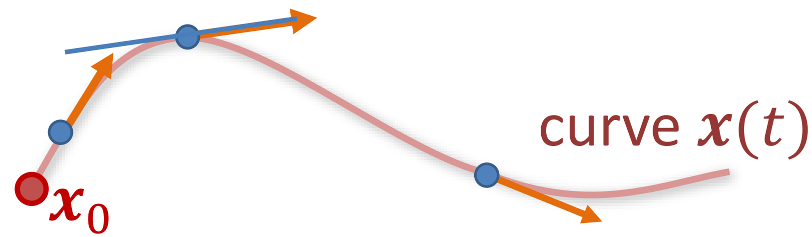
[Bürger et al. 09]

Characteristic Lines

SIEMENS
Ingenuity for life

- Characteristic lines are tangential to the flow
  - Means that line tangent (1st derivative) = vector field direction

$$\frac{\partial \boldsymbol{x}(t)}{\partial t} = \boldsymbol{v}(\boldsymbol{x}(t), t)$$



  - Characteristic lines do not intersect!
  - They are solutions to the initial value problem of an ordinary differential equation (ODE)

$$\boldsymbol{x}(0) = \boldsymbol{x}_0\,, \qquad \frac{\partial \boldsymbol{x}(t)}{\partial t} = \boldsymbol{v}(\boldsymbol{x}(t), t)$$

Initial value
(seed point $\boldsymbol{x}_0$)

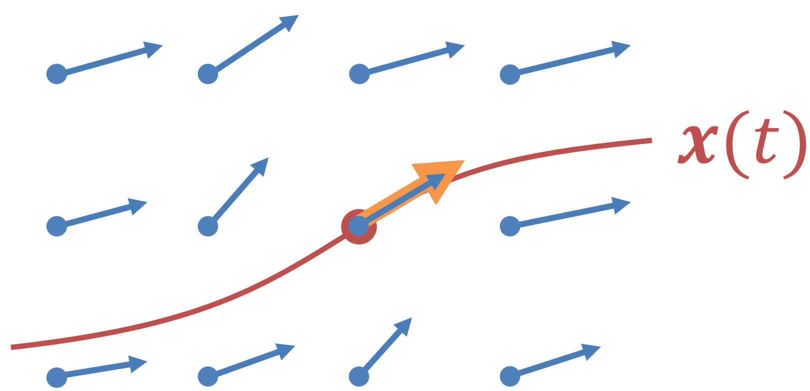
Ordinary differential equation (ODE)

Numerical Integration of ODEs

SIEMENS
Ingenuity for life

- Particle tracing problem – how to solve the differential equation:

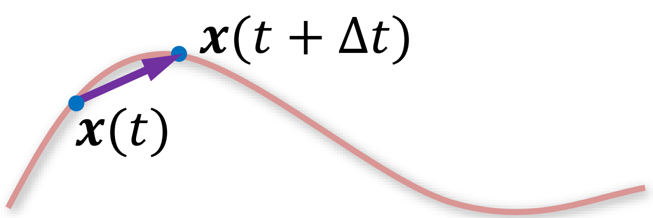
$$\boldsymbol{x}(0) = \boldsymbol{x}_0 \,, \qquad \dot{\boldsymbol{x}}(t) = \frac{\partial \boldsymbol{x}(t)}{\partial t} = \boldsymbol{v}(\boldsymbol{x}(t), t)$$



- What kind of numerical solver?

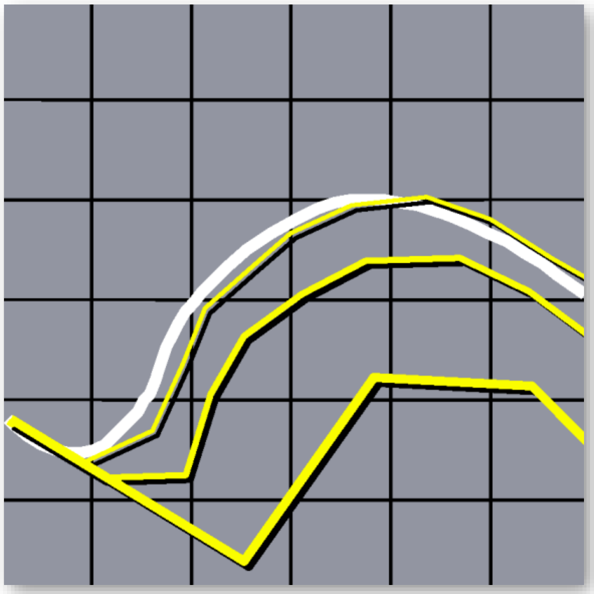
Numerical Integration of ODEs

- Rewrite ODE in generic form
- Initial value problem for: $\dot{\boldsymbol{x}}(t) = \boldsymbol{v}(\boldsymbol{x}(t), t)$
- Simple approach: Euler method



$$\frac{\boldsymbol{x}(t + \Delta t) - \boldsymbol{x}(t)}{\Delta t} \approx \boldsymbol{v}(\boldsymbol{x}(t), t)$$

$$\Rightarrow \boldsymbol{x}(t + \Delta t) \approx \boldsymbol{x}(t) + \Delta t \cdot \boldsymbol{v}(\boldsymbol{x}(t), t)$$

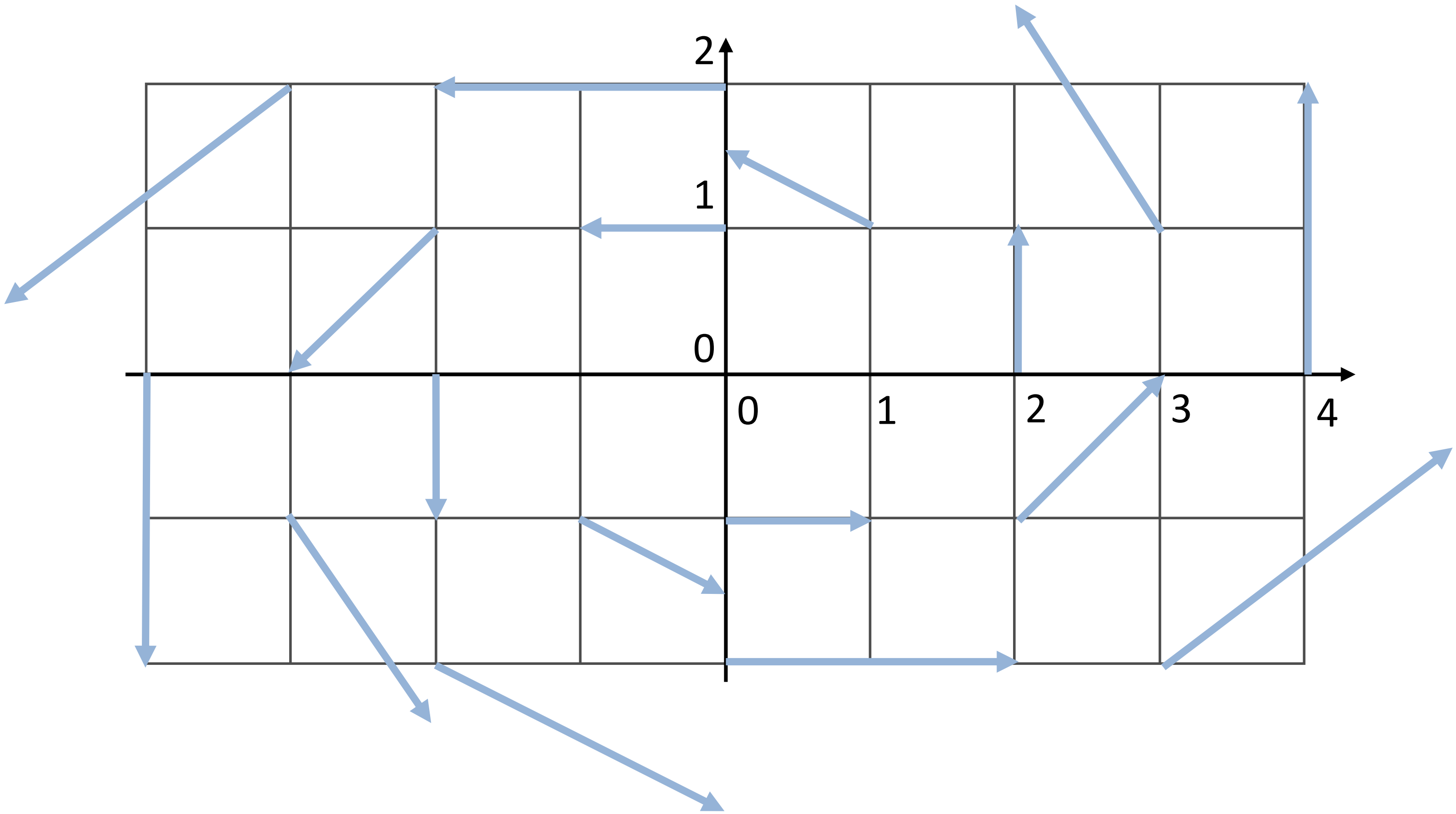


  – First order method
  – Higher accuracy with smaller step size

Numerical Integration of ODEs

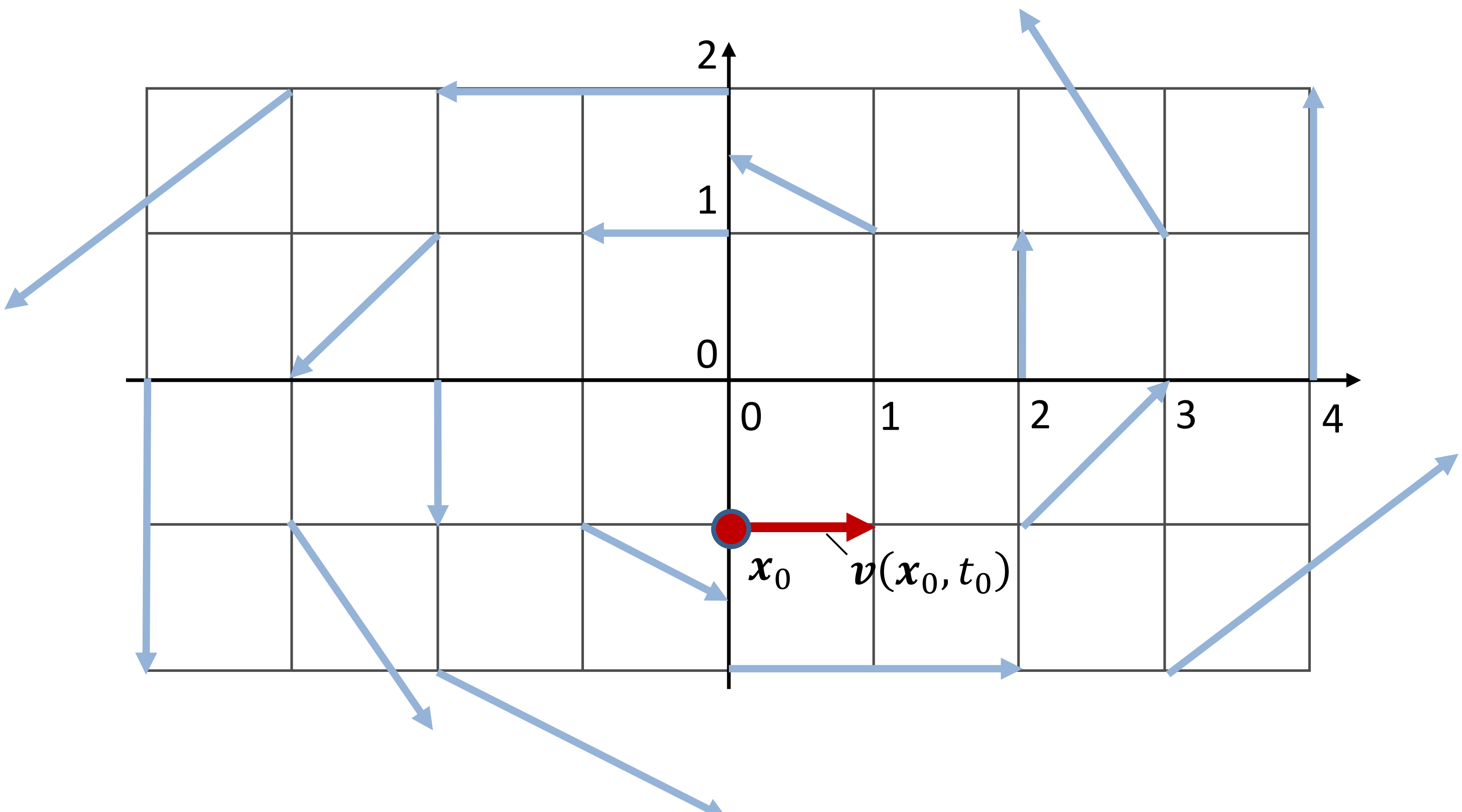
SIEMENS
Ingenuity for life

- Example: 2D vector field $\boldsymbol{v}(x, y, t) = (-y, {}^{x}/_{2})$
  - Sample arrows shown
  - True solution are ellipses



Numerical Integration of ODEs

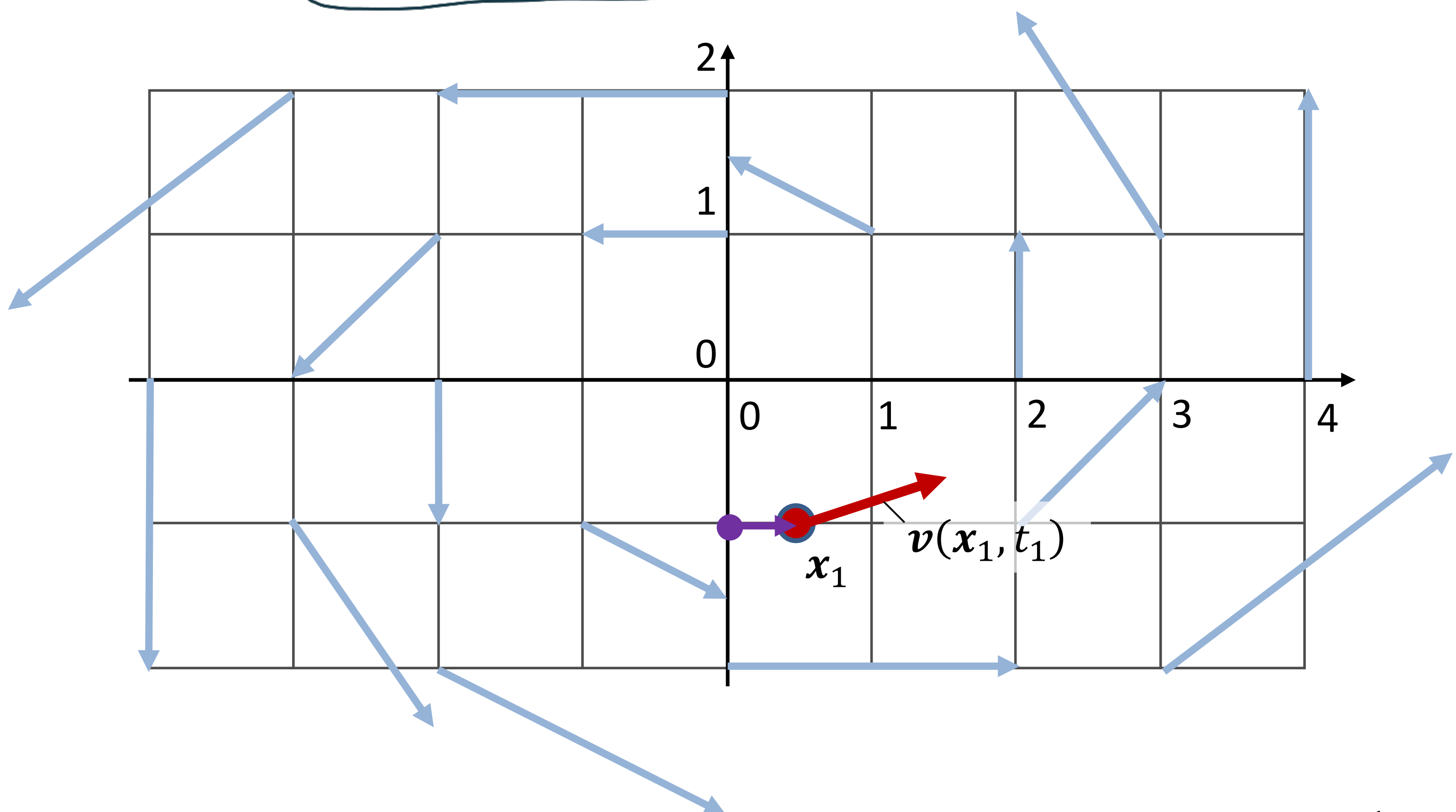
- Example: Euler Integration $(\Delta t = 0.5)$
  - Seed point: $\boldsymbol{x}_0 = \begin{pmatrix} 0 \\ -1 \end{pmatrix}$, current vector $\boldsymbol{v}(\boldsymbol{x}_0, t_0) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$



$$\boldsymbol{v}(x, y, t) = \left(-y, {}^{x}/_{2}\right)$$

Numerical Integration of ODEs

- Example: Euler Integration $(\Delta t = 0.5)$
  - New point: $\boxed{\boldsymbol{x}_1 = \boldsymbol{x}_0 + \Delta t \cdot \boldsymbol{v}(\boldsymbol{x}_0, t_0)} = \binom{0.5}{-1}$, $\quad \boldsymbol{v}(\boldsymbol{x}_1, t_1) = \binom{1}{0.25}$

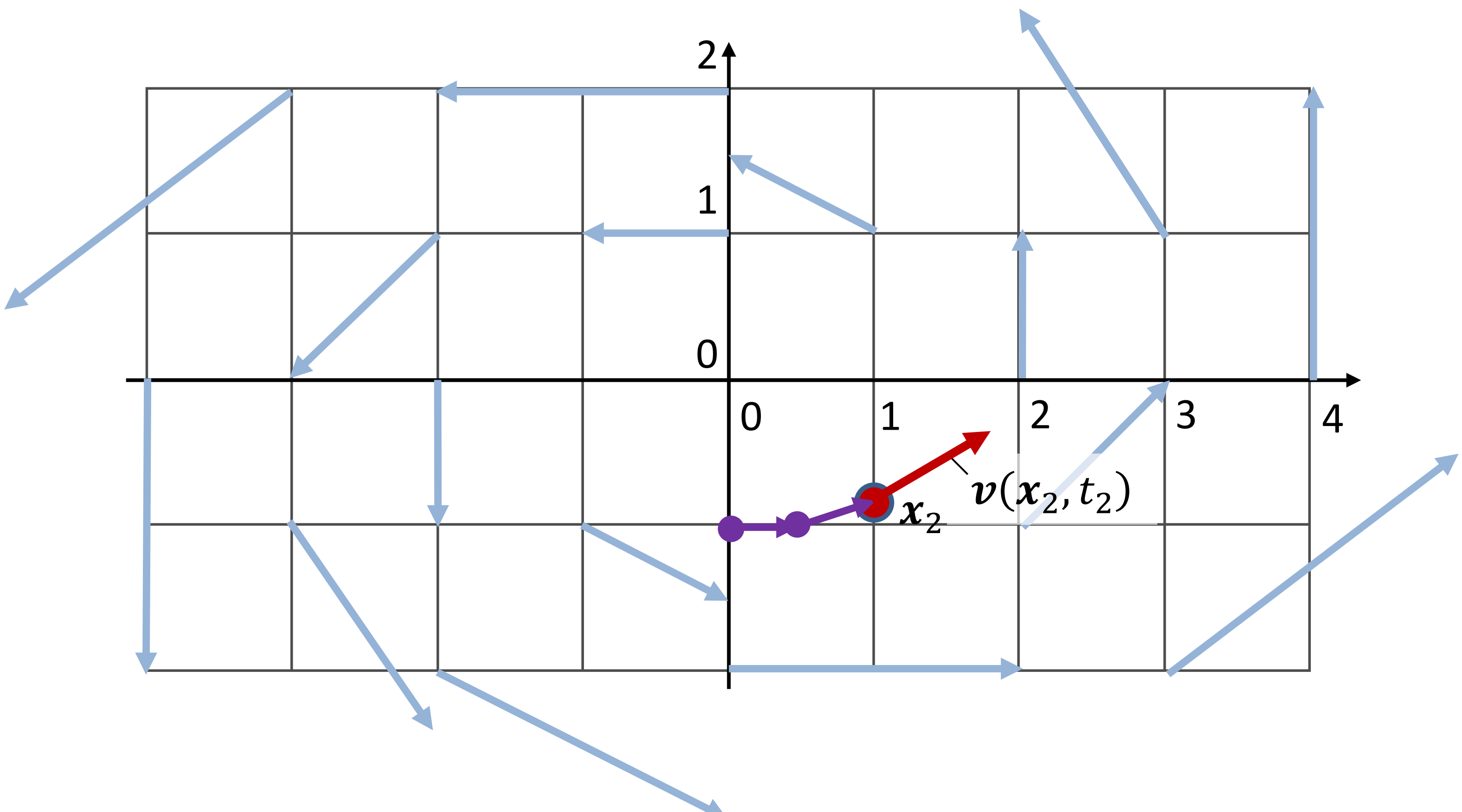


$$\boldsymbol{v}(x, y, t) = (-y, {}^{x}/_{2})$$

Numerical Integration of ODEs

SIEMENS
Ingenuity for life

- Example: Euler Integration $(\Delta t = 0.5)$
  - New point: $\boldsymbol{x}_2 = \boldsymbol{x}_1 + \Delta t \cdot \boldsymbol{v}(\boldsymbol{x}_1, t_1) \approx \begin{pmatrix} 1 \\ -0.88 \end{pmatrix}$, $\boldsymbol{v}(\boldsymbol{x}_2, t_2) \approx \begin{pmatrix} 0.88 \\ 0.5 \end{pmatrix}$

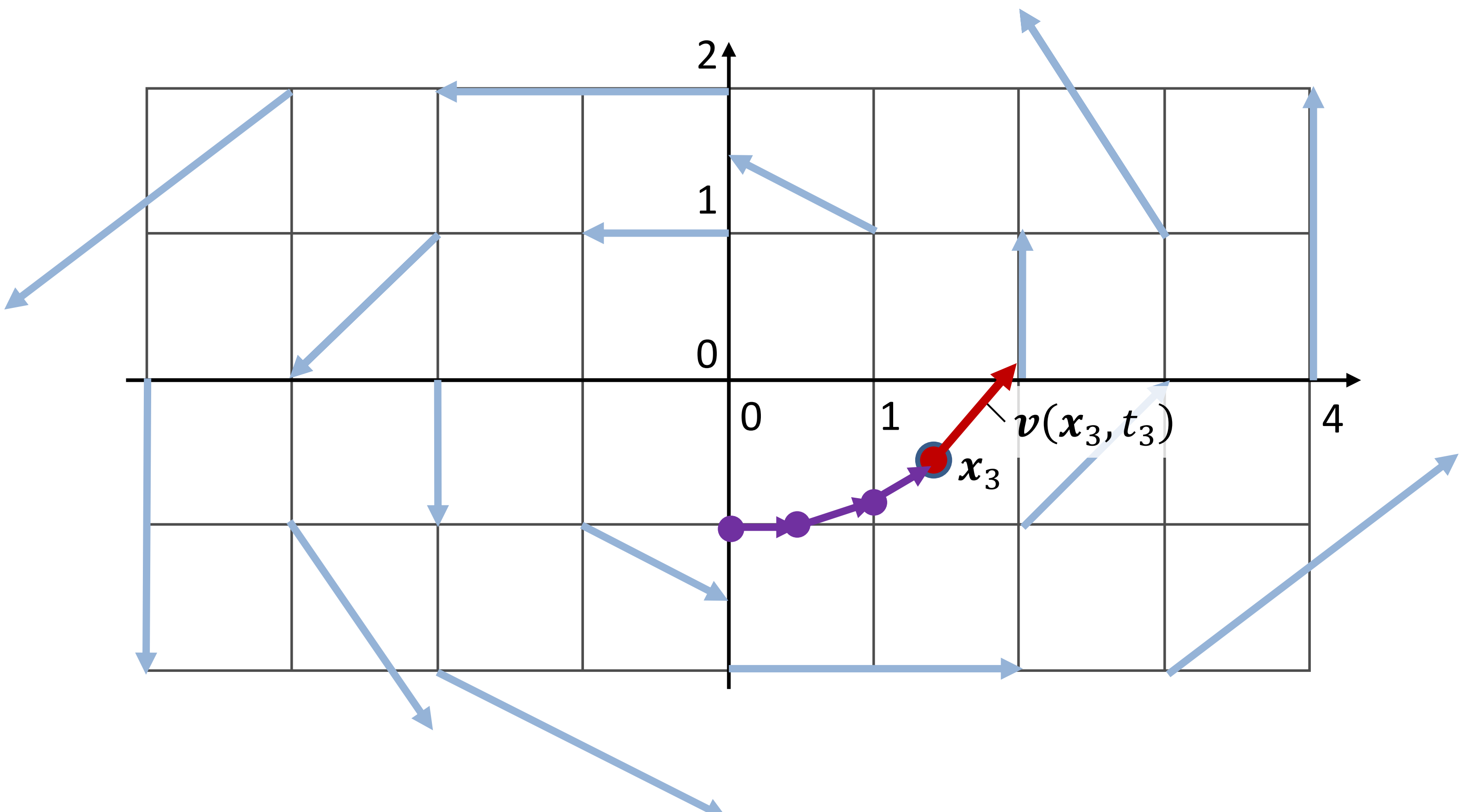


$\boldsymbol{v}(x, y, t) = (-y, {}^{x}/_{2})$

Numerical Integration of ODEs

SIEMENS
Ingenuity for life

- Example: Euler Integration $(\Delta t = 0.5)$
  - New point: $\boldsymbol{x}_3 = \boldsymbol{x}_2 + \Delta t \cdot \boldsymbol{v}(\boldsymbol{x}_2, t_2) \approx \begin{pmatrix} 1.44 \\ -0.63 \end{pmatrix}, \quad \boldsymbol{v}(\boldsymbol{x}_3, t_3) \approx \begin{pmatrix} 0.63 \\ 0.72 \end{pmatrix}$

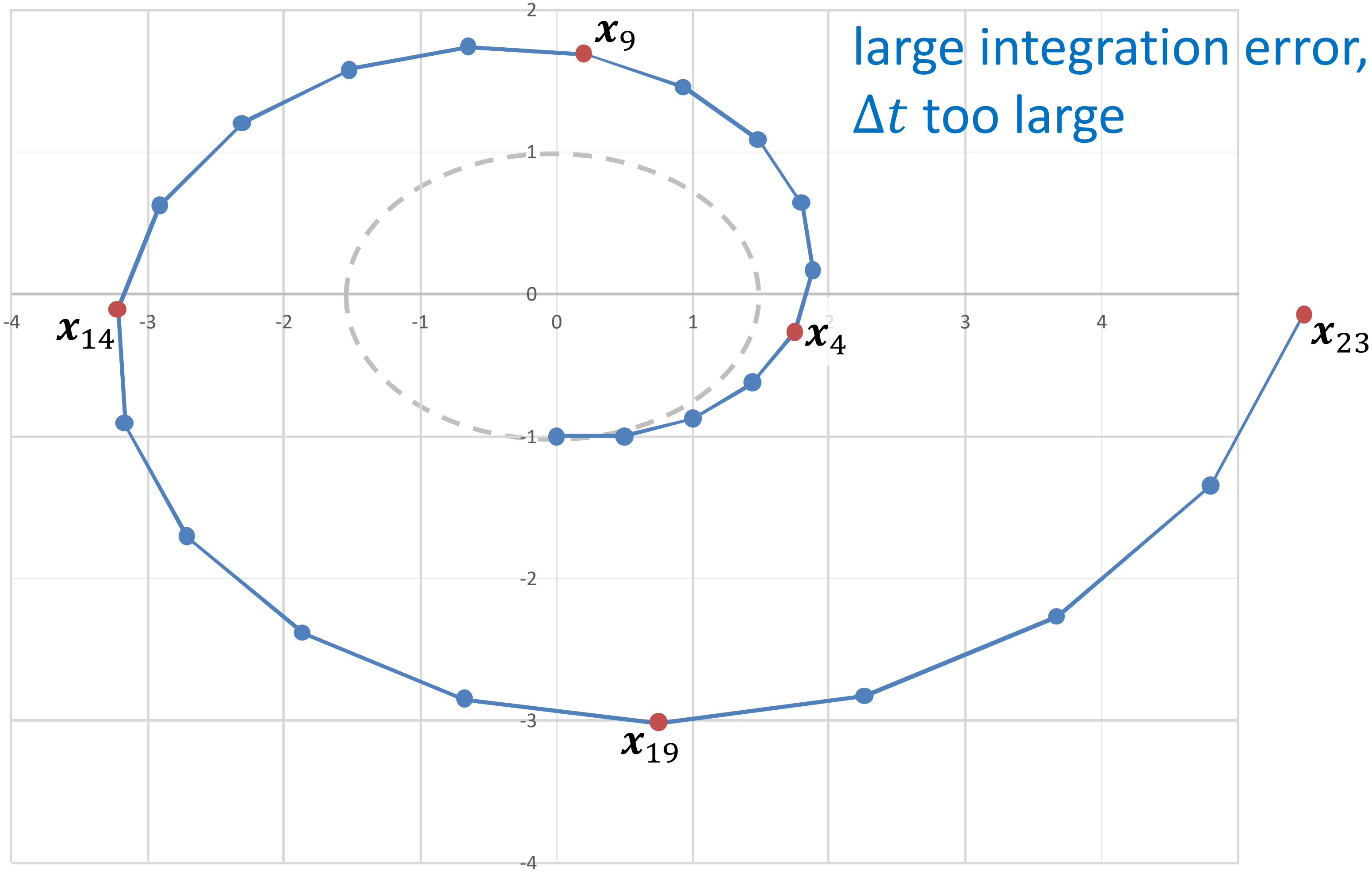


$$\boldsymbol{v}(x, y, t) = \left(-y, {}^{x}/_{2}\right)$$

Numerical Integration of ODEs

SIEMENS
Ingenuity for life

- Euler Integration after 23 steps $(\Delta t = 0.5)$

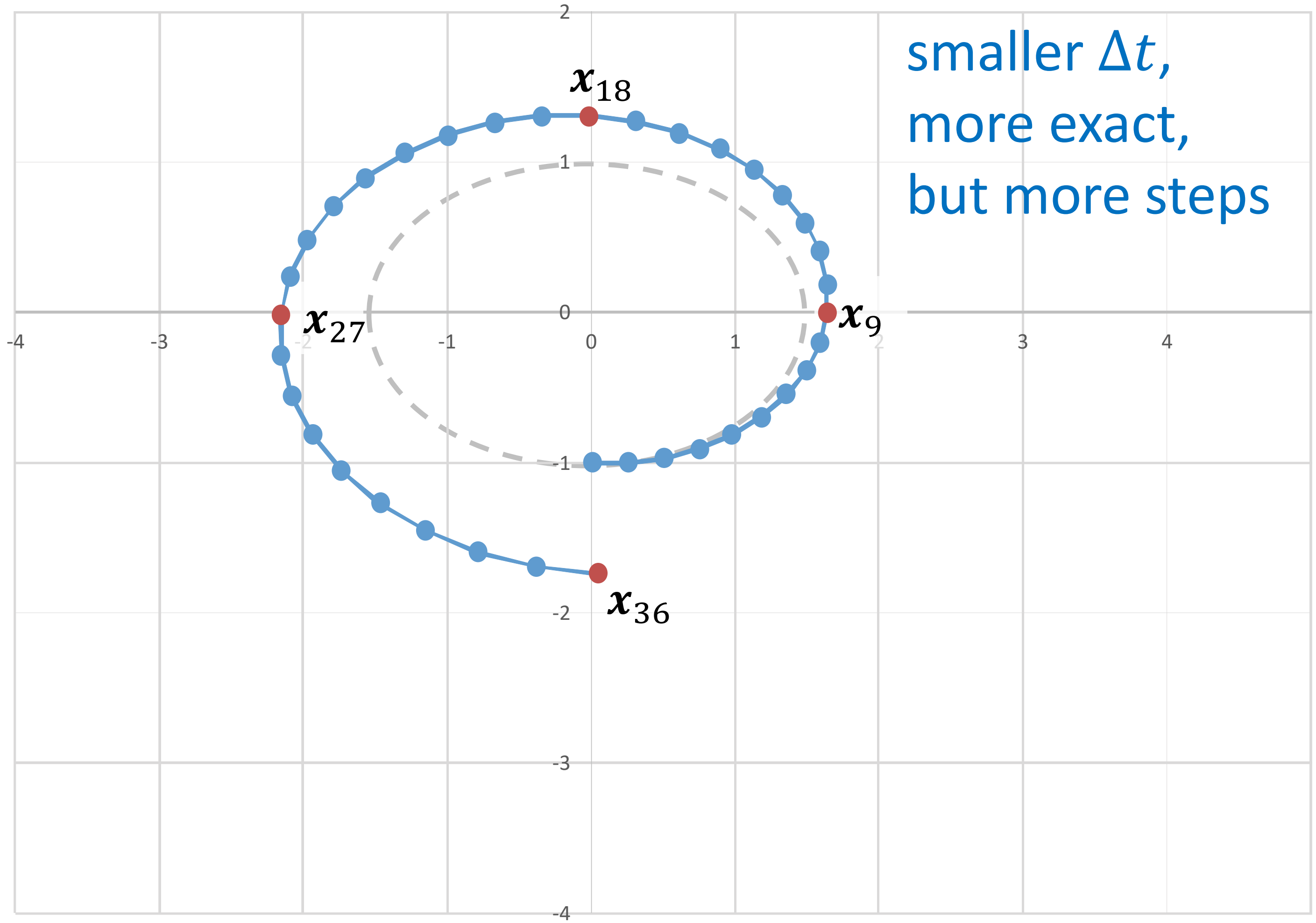


large integration error, Δt too large

Numerical Integration of ODEs

SIEMENS
Ingenuity for life

- Euler Integration after 36 steps $(\Delta t = 0.25)$

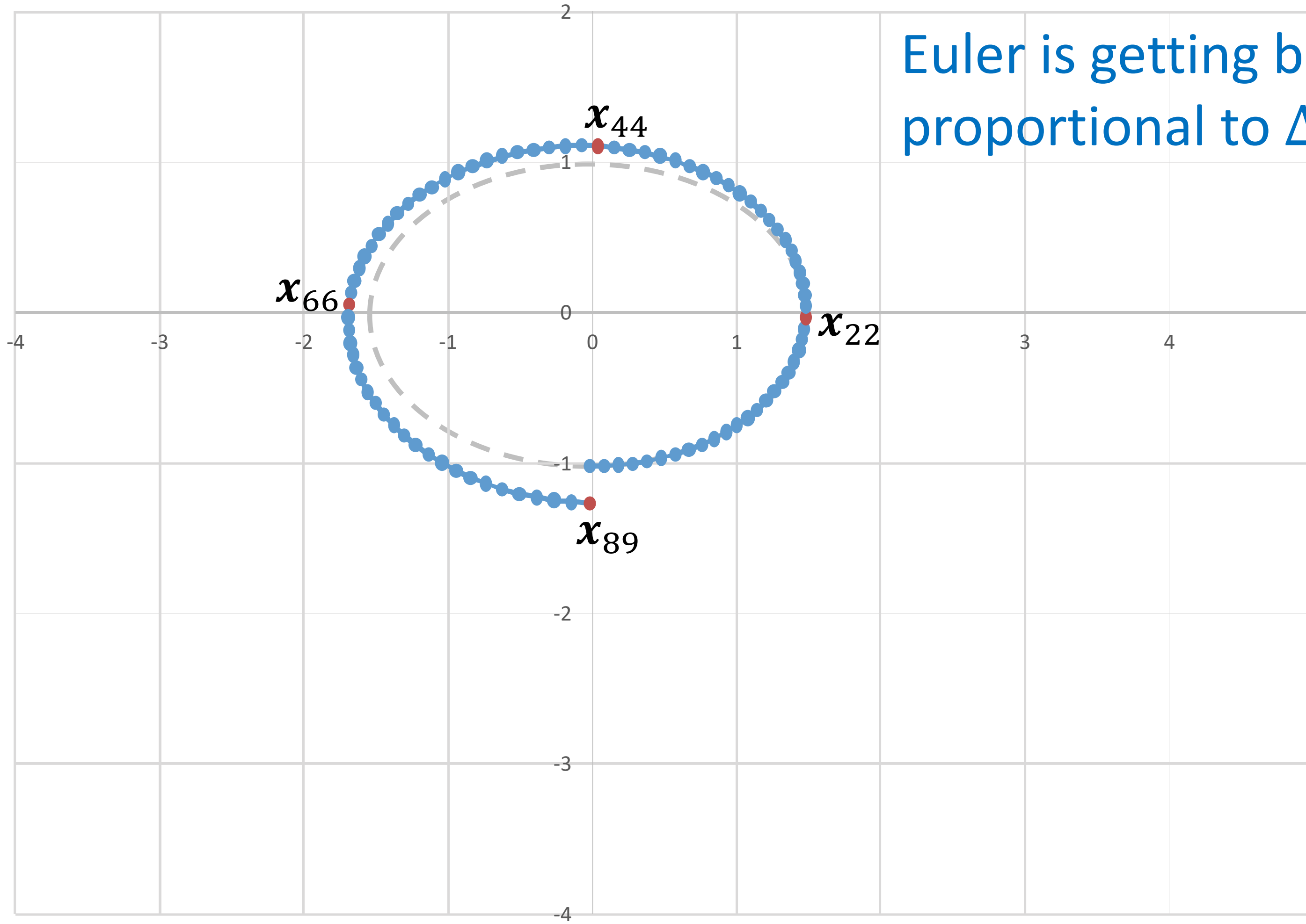


smaller Δt,
more exact,
but more steps

Numerical Integration of ODEs

SIEMENS
Ingenuity for life

- Euler Integration after 89 steps $(\Delta t = 0.1)$

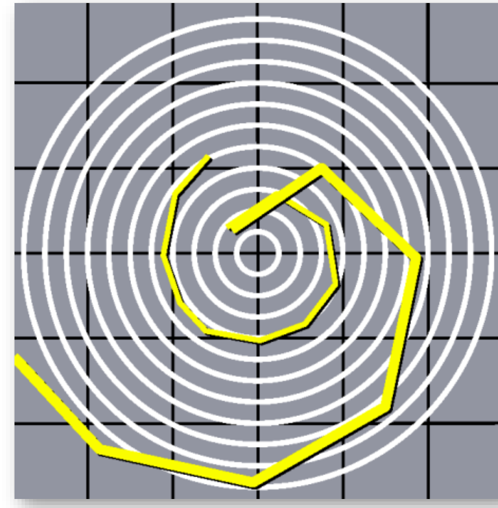


Euler is getting better proportional to Δt

Numerical Integration of ODEs

SIEMENS
Ingenuity for life

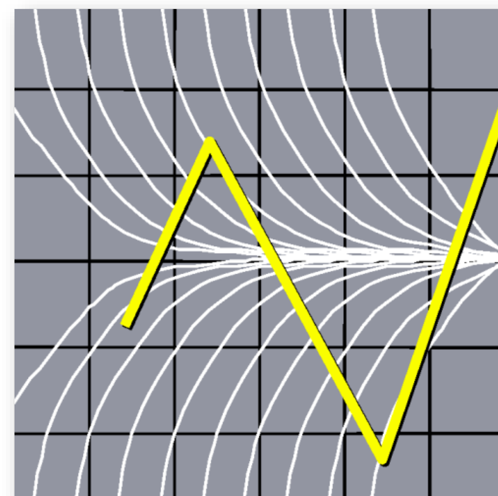
- Problem of Euler method
  - Inaccurate
  - Unstable



- Example:

$$\dot{\boldsymbol{x}}(t) = -k\boldsymbol{x}(t)$$
$$\boldsymbol{x}(t) = e^{-kt}$$

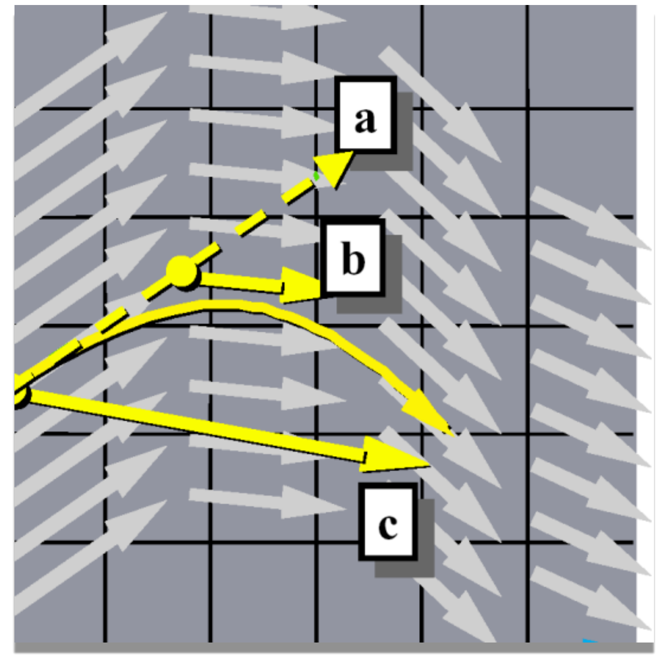
divergence for $\Delta t > 2/k$



Numerical Integration of ODEs

SIEMENS
Ingenuity for life

- Midpoint method (Runge-Kutta 2nd order)
  a. Euler step
  $$\Delta \boldsymbol{x} = \Delta t \cdot \boldsymbol{v}(\boldsymbol{x}, t)$$
  b. Evaluation of $\boldsymbol{v}$ at midpoint
  $$\boldsymbol{v}_{\text{mid}} = \boldsymbol{v}(\boldsymbol{x} + \frac{\Delta \boldsymbol{x}}{2}, t + \frac{\Delta t}{2})$$
  c. Complete step with vector at midpoint
  $$\boldsymbol{x}(t + \Delta t) = \boldsymbol{x}(t) + \Delta t \cdot \boldsymbol{v}_{\text{mid}}$$

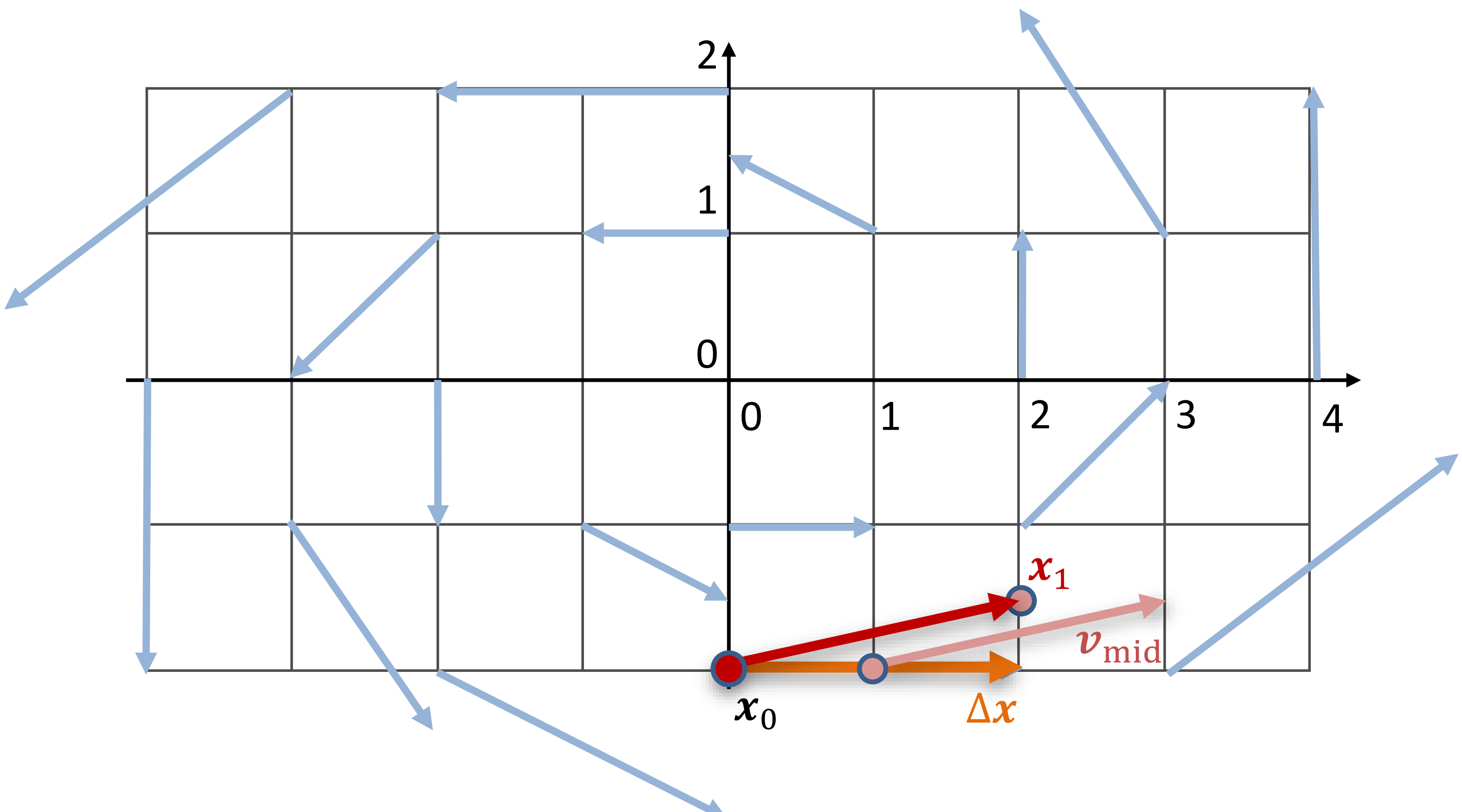


More accurate results with midpoint method

Numerical Integration of ODEs

SIEMENS
Ingenuity for life

- Example: Midpoint method $(\Delta t = 1)$
  - Seed point: $\boldsymbol{x}_0 = \begin{pmatrix} 0 \\ -2 \end{pmatrix}$, $\Delta\boldsymbol{x} = \begin{pmatrix} 2 \\ 0 \end{pmatrix}$, $\boldsymbol{v}_{\mathrm{mid}} = \boldsymbol{v}\left(\boldsymbol{x}_0 + \frac{\Delta\boldsymbol{x}}{2}, t_0 + \frac{\Delta t}{2}\right) = \begin{pmatrix} 2 \\ 0.5 \end{pmatrix}$

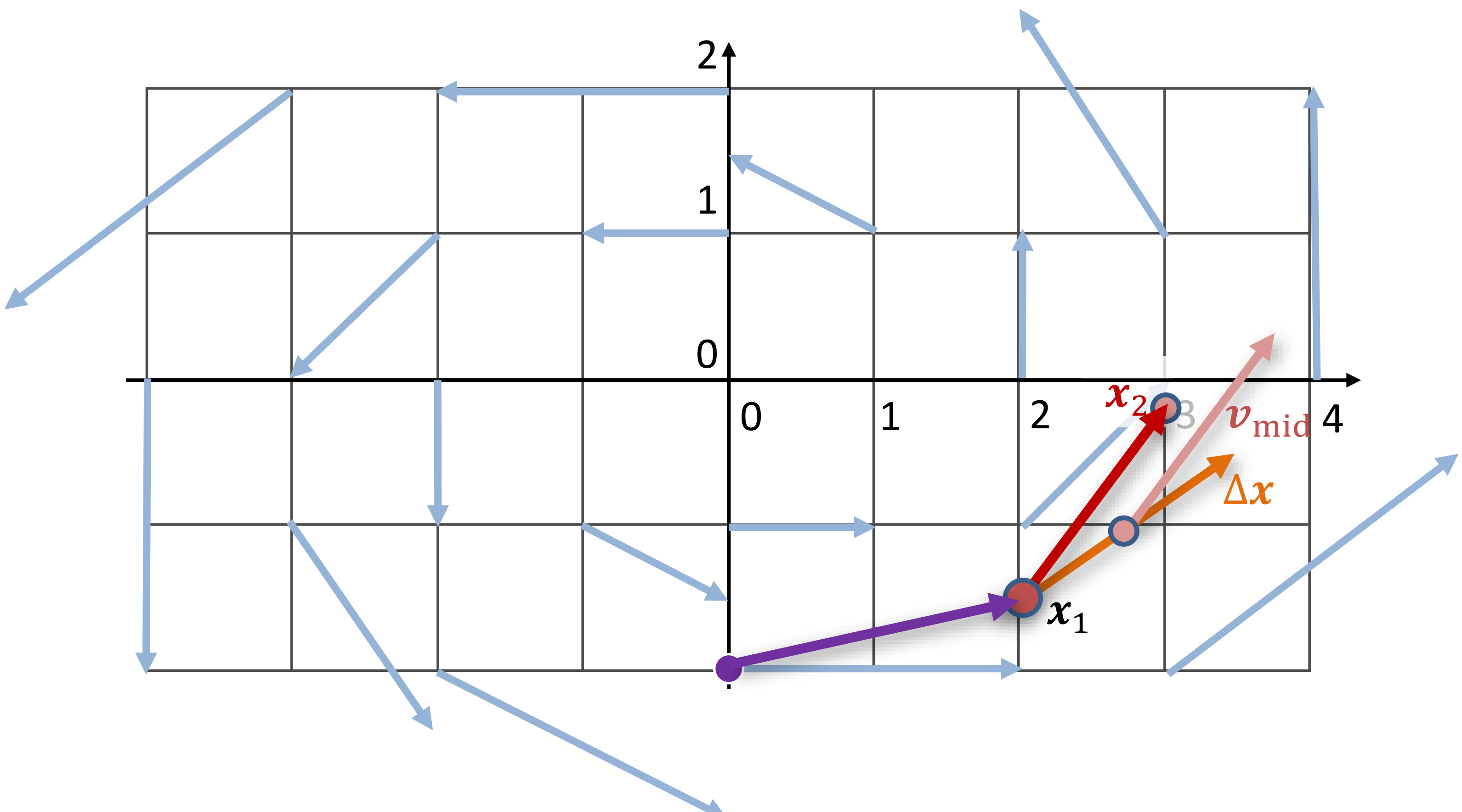


$$\boldsymbol{v}(x, y, t) = \left(-y, {}^{x}/_{2}\right)$$

Numerical Integration of ODEs

SIEMENS
Ingenuity for life

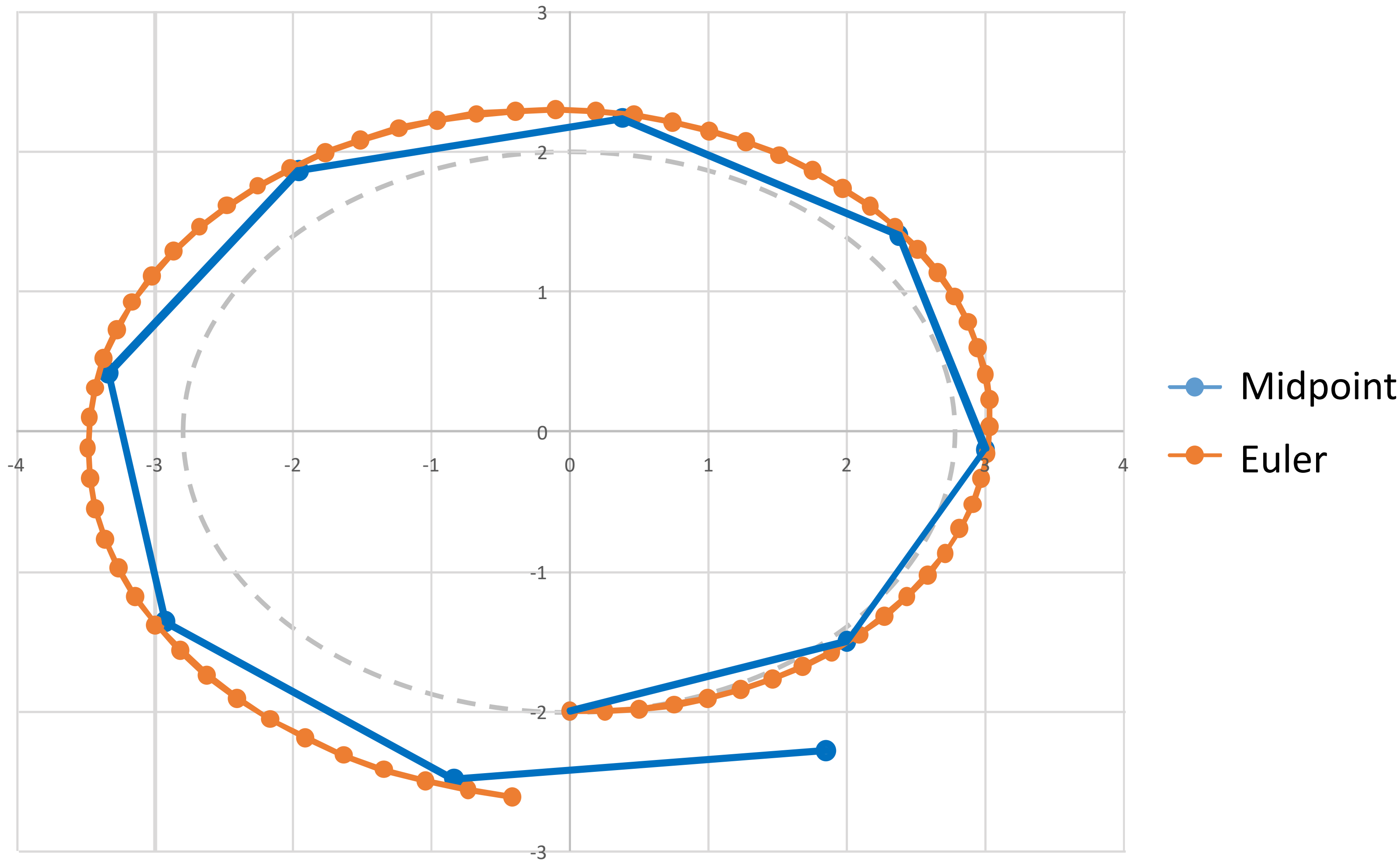
- Example: Midpoint method ($\Delta t = 1$)
  - New point: $\boldsymbol{x}_1 = \binom{2}{-1.5}$, $\Delta\boldsymbol{x} = \binom{1.5}{1}$, $\boldsymbol{v}_{\mathrm{mid}} = \boldsymbol{v}\left(\boldsymbol{x}_1 + \frac{\Delta\boldsymbol{x}}{2}, t_1 + \frac{\Delta t}{2}\right) \approx \binom{1}{1.4}$



$\boldsymbol{v}(x, y, t) = \left(-y, {}^{x}/_{2}\right)$

Numerical Integration of ODEs

- Comparison: Midpoint method even with $\Delta t = 1$ (9 steps) better than Euler with $\Delta t = 1/8$ (71 steps)



Numerical Integration of ODEs

SIEMENS
Ingenuity for life

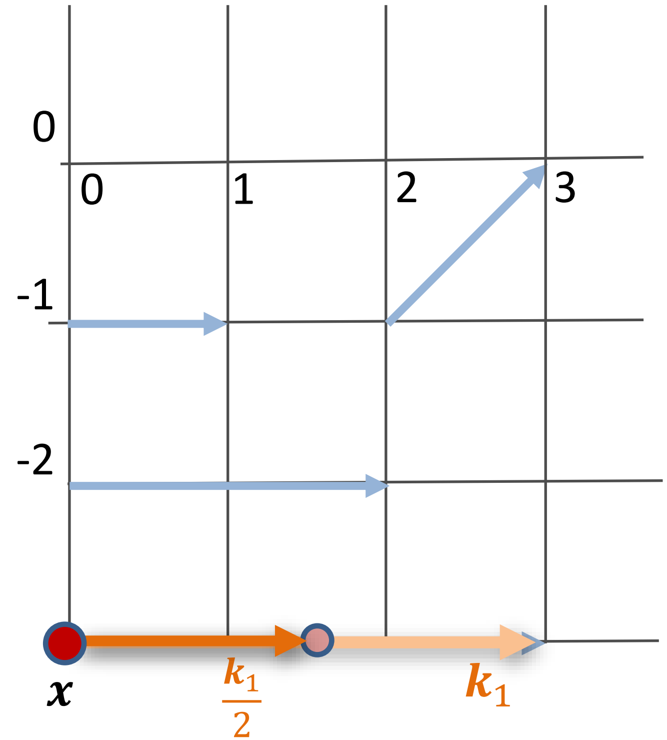
- Runge-Kutta of 4th order ($\Delta t = 1$)

$$\boxed{\boldsymbol{k}_1 = \Delta t \cdot \boldsymbol{v}(\boldsymbol{x}, t)}$$

$$\boldsymbol{k}_2 = \Delta t \cdot \boldsymbol{v}\left(\boxed{\boldsymbol{x} + \frac{\boldsymbol{k}_1}{2}}, t + \frac{\Delta t}{2}\right)$$

$$\boldsymbol{k}_3 = \Delta t \cdot \boldsymbol{v}\left(\boldsymbol{x} + \frac{\boldsymbol{k}_2}{2}, t + \frac{\Delta t}{2}\right)$$

$$\boldsymbol{k}_4 = \Delta t \cdot \boldsymbol{v}(\boldsymbol{x} + \boldsymbol{k}_3, t + \Delta t)$$



$$\boldsymbol{x}(t + \Delta t) = \boldsymbol{x} + \frac{\boldsymbol{k}_1}{6} + \frac{\boldsymbol{k}_2}{3} + \frac{\boldsymbol{k}_3}{3} + \frac{\boldsymbol{k}_4}{6}$$

Numerical Integration of ODEs

SIEMENS
Ingenuity for life

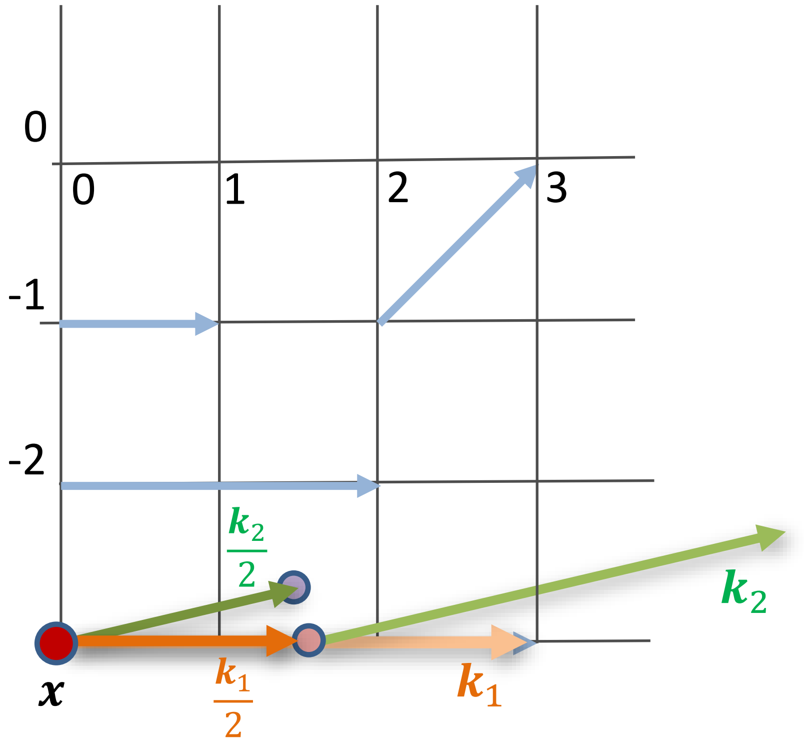
- Runge-Kutta of 4th order ($\Delta t = 1$)

$$\boldsymbol{k}_1 = \Delta t \cdot \boldsymbol{v}(\boldsymbol{x}, t)$$

$$\boldsymbol{k}_2 = \Delta t \cdot \boldsymbol{v}\left(\boldsymbol{x} + \frac{\boldsymbol{k}_1}{2}, t + \frac{\Delta t}{2}\right)$$

$$\boldsymbol{k}_3 = \Delta t \cdot \boldsymbol{v}\left(\boldsymbol{x} + \frac{\boldsymbol{k}_2}{2}, t + \frac{\Delta t}{2}\right)$$

$$\boldsymbol{k}_4 = \Delta t \cdot \boldsymbol{v}(\boldsymbol{x} + \boldsymbol{k}_3, t + \Delta t)$$



$$\boldsymbol{x}(t + \Delta t) = \boldsymbol{x} + \frac{\boldsymbol{k}_1}{6} + \frac{\boldsymbol{k}_2}{3} + \frac{\boldsymbol{k}_3}{3} + \frac{\boldsymbol{k}_4}{6}$$

Numerical Integration of ODEs

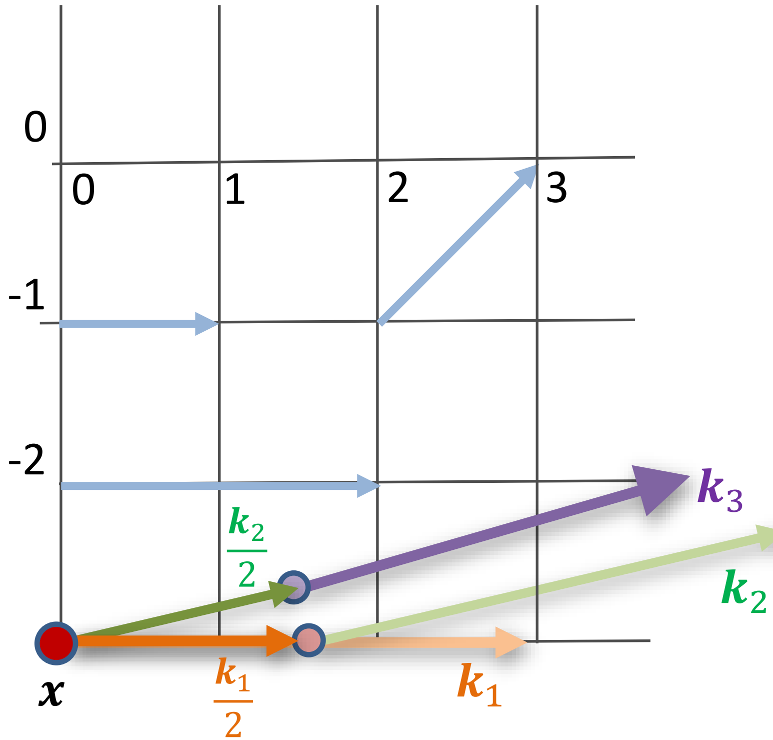
- Runge-Kutta of 4th order ($\Delta t = 1$)

$$\boldsymbol{k}_1 = \Delta t \cdot \boldsymbol{v}(\boldsymbol{x}, t)$$

$$\boldsymbol{k}_2 = \Delta t \cdot \boldsymbol{v}\left(\boldsymbol{x} + \frac{\boldsymbol{k}_1}{2}, t + \frac{\Delta t}{2}\right)$$

$$\boldsymbol{k}_3 = \Delta t \cdot \boldsymbol{v}\left(\boldsymbol{x} + \frac{\boldsymbol{k}_2}{2}, t + \frac{\Delta t}{2}\right)$$

$$\boldsymbol{k}_4 = \Delta t \cdot \boldsymbol{v}(\boldsymbol{x} + \boldsymbol{k}_3, t + \Delta t)$$



$$\boldsymbol{x}(t + \Delta t) = \boldsymbol{x} + \frac{\boldsymbol{k}_1}{6} + \frac{\boldsymbol{k}_2}{3} + \frac{\boldsymbol{k}_3}{3} + \frac{\boldsymbol{k}_4}{6}$$

Numerical Integration of ODEs

SIEMENS
Ingenuity for life

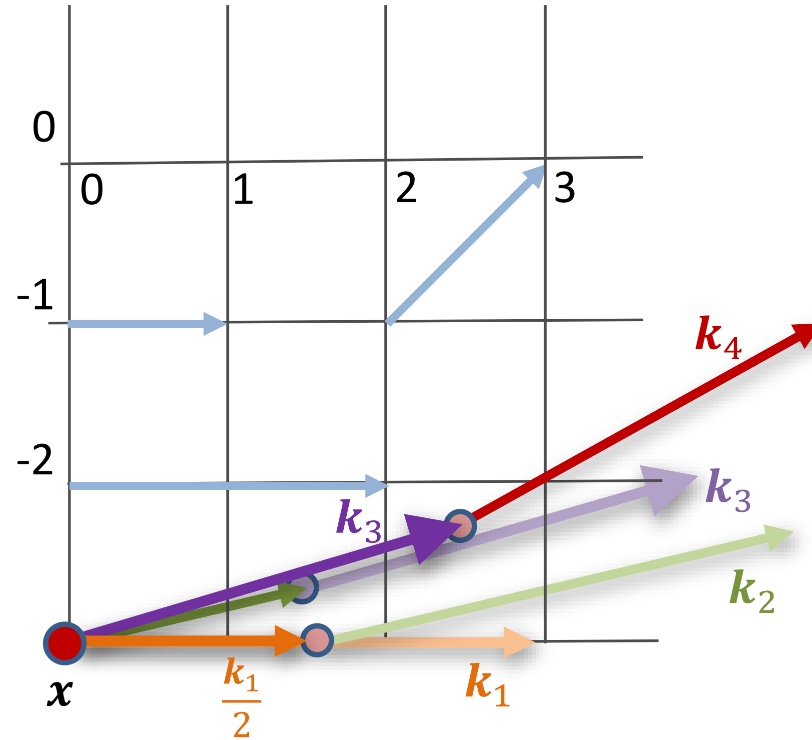
- Runge-Kutta of 4th order ($\Delta t = 1$)

$$\boldsymbol{k}_1 = \Delta t \cdot \boldsymbol{v}(\boldsymbol{x}, t)$$

$$\boldsymbol{k}_2 = \Delta t \cdot \boldsymbol{v}\left(\boldsymbol{x} + \frac{\boldsymbol{k}_1}{2}, t + \frac{\Delta t}{2}\right)$$

$$\boldsymbol{k}_3 = \Delta t \cdot \boldsymbol{v}\left(\boldsymbol{x} + \frac{\boldsymbol{k}_2}{2}, t + \frac{\Delta t}{2}\right)$$

$$\boldsymbol{k}_4 = \Delta t \cdot \boldsymbol{v}(\boldsymbol{x} + \boldsymbol{k}_3, t + \Delta t)$$



$$\boldsymbol{x}(t + \Delta t) = \boldsymbol{x} + \frac{\boldsymbol{k}_1}{6} + \frac{\boldsymbol{k}_2}{3} + \frac{\boldsymbol{k}_3}{3} + \frac{\boldsymbol{k}_4}{6}$$

Numerical Integration of ODEs

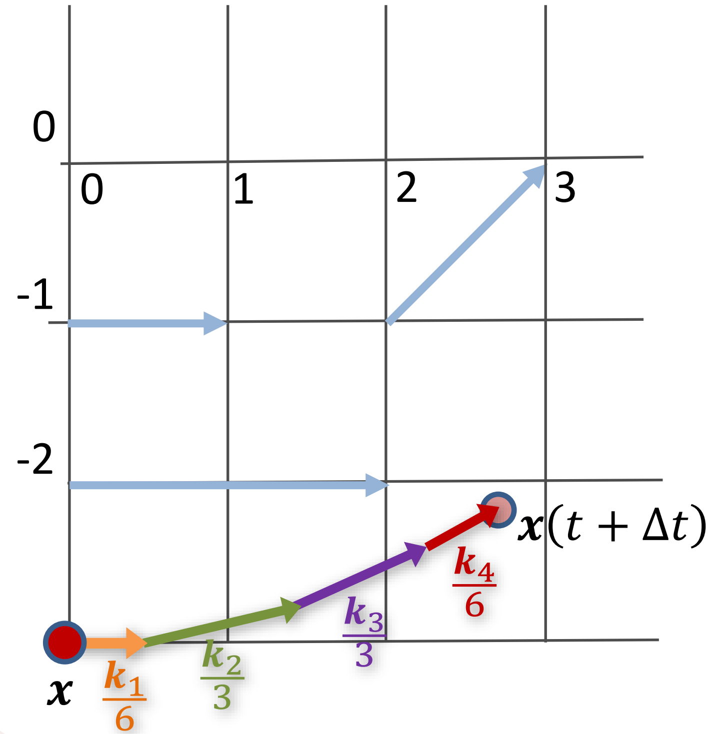
- Runge-Kutta of 4th order ($\Delta t = 1$)

$$\boldsymbol{k}_1 = \Delta t \cdot \boldsymbol{v}(\boldsymbol{x}, t)$$

$$\boldsymbol{k}_2 = \Delta t \cdot \boldsymbol{v}\left(\boldsymbol{x} + \frac{\boldsymbol{k}_1}{2}, t + \frac{\Delta t}{2}\right)$$

$$\boldsymbol{k}_3 = \Delta t \cdot \boldsymbol{v}\left(\boldsymbol{x} + \frac{\boldsymbol{k}_2}{2}, t + \frac{\Delta t}{2}\right)$$

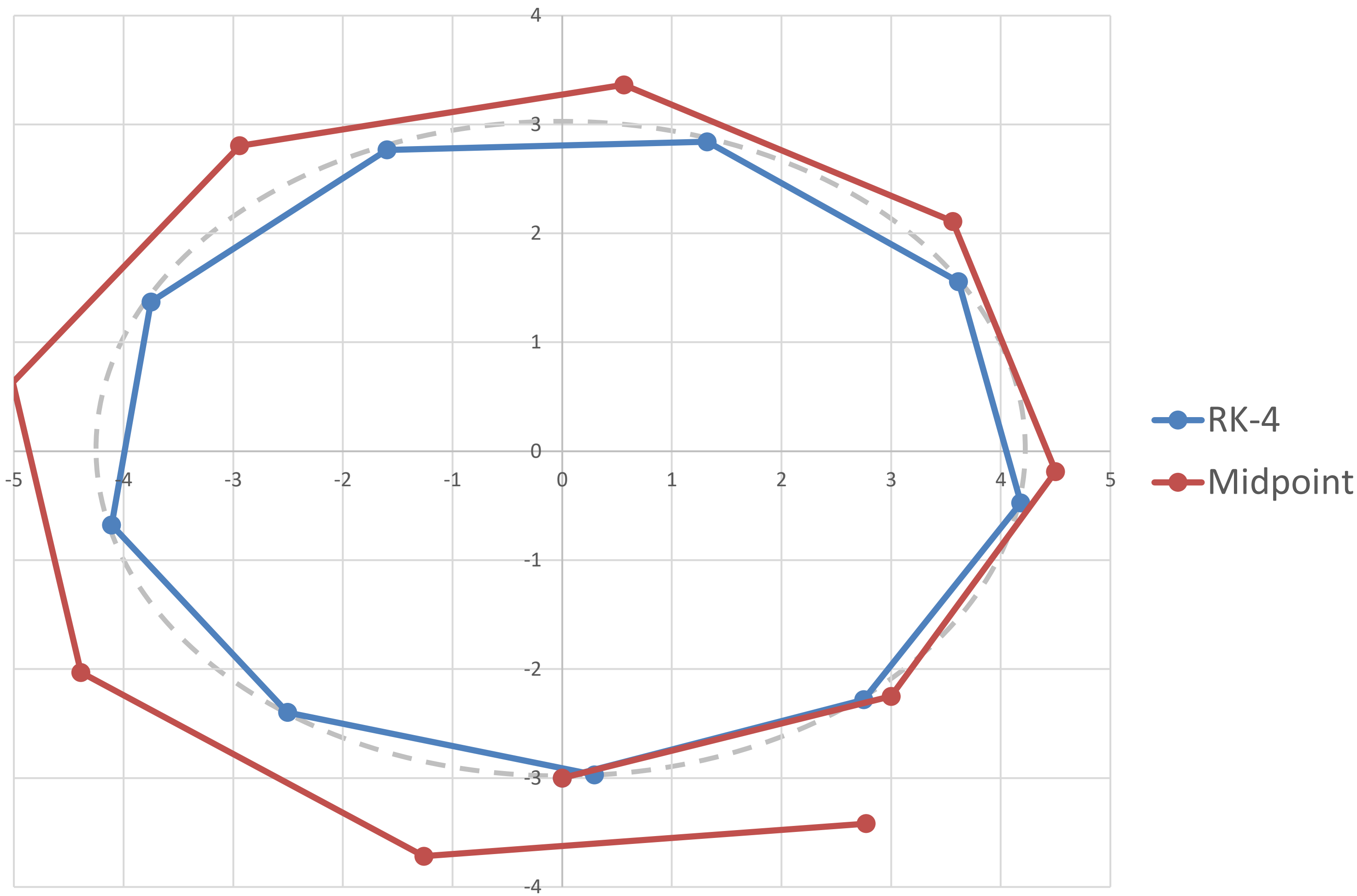
$$\boldsymbol{k}_4 = \Delta t \cdot \boldsymbol{v}(\boldsymbol{x} + \boldsymbol{k}_3, t + \Delta t)$$



$$\boldsymbol{x}(t + \Delta t) = \boldsymbol{x} + \frac{\boldsymbol{k}_1}{6} + \frac{\boldsymbol{k}_2}{3} + \frac{\boldsymbol{k}_3}{3} + \frac{\boldsymbol{k}_4}{6}$$

Numerical Integration of ODEs

- Comparison: Runge-Kutta 4th order vs. midpoint method after 9 steps ($\Delta t = 1$)



Numerical Integration of ODEs

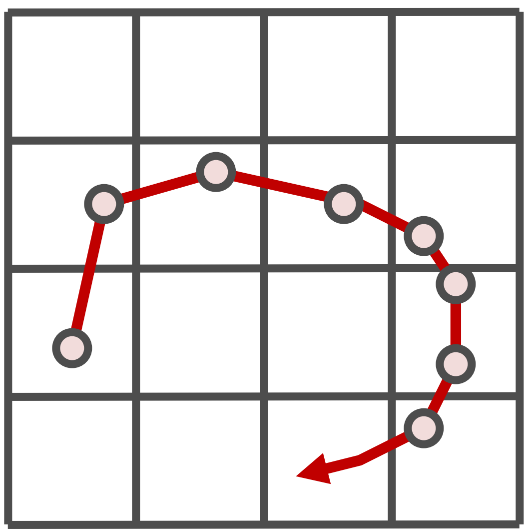
- Summary
  - Analytic determination of streamlines usually not possible
  - Hence, numerical integration
  - Several methods available
    - Euler: simple, imprecise, especially with larger Δt
    - Runge-Kutta: more accurate in higher orders,
      pays off with complex flows

Particle Tracing on Grids

SIEMENS
Ingenuity for life

- Vector field given on a grid
  - Solve $\boldsymbol{L}(0) = \boldsymbol{x}_0, \quad \frac{d\boldsymbol{L}(u)}{du} = \boldsymbol{v}(\boldsymbol{L}(u), t)$

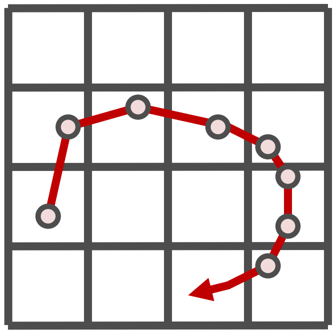
    for the path line
  - Incremental integration
  - Discretized path of the particle



Particle Tracing on Grids

SIEMENS
Ingenuity for life

- Most simple case: Cartesian grid
- <u>Basic algorithm:</u>



```
Select start point (seed point)
Find cell that contains start point // point location
While (particle in domain) do
    Interpolate vector field at     // interpolation
      current position
    Integrate to new position       // integration
    Find new cell                   // point location
    Draw line segment between latest
      particle positions
Endwhile
```

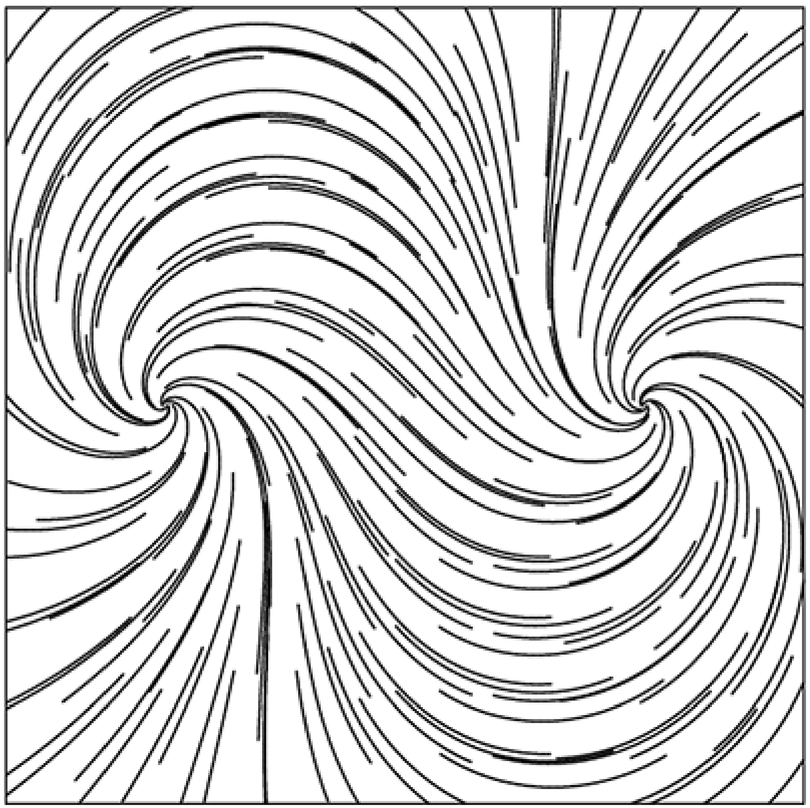
Particle Tracing on Grids

- Point location (cell search) on Cartesian grids
  - Indices of cell directly from position (x, y, z)
  - For example: $i_x = \lfloor (x - x_0) / \Delta x \rfloor$
  - Simple and fast
- Interpolation on Cartesian grids
  - Bilinear (in 2D) or trilinear (in 3D) interpolation
  - Required to compute the vector field (= velocity) inside a cell
  - Component-wise interpolation

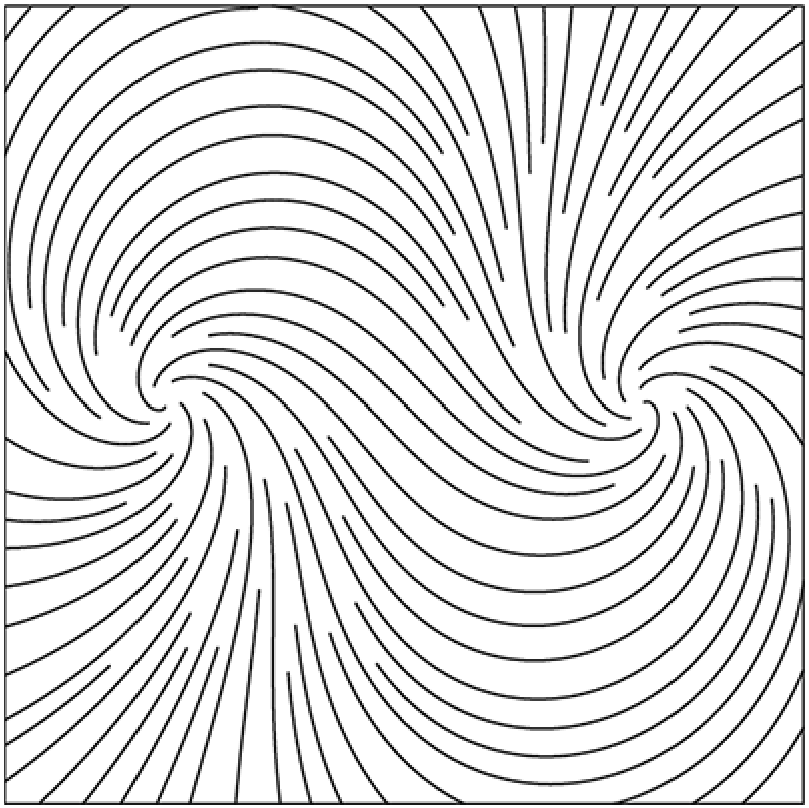
Stream line placement in 2D

SIEMENS
Ingenuity for life

- Stream line placement
  – Irregular results when using regular grid



Stream line placement from regular grid



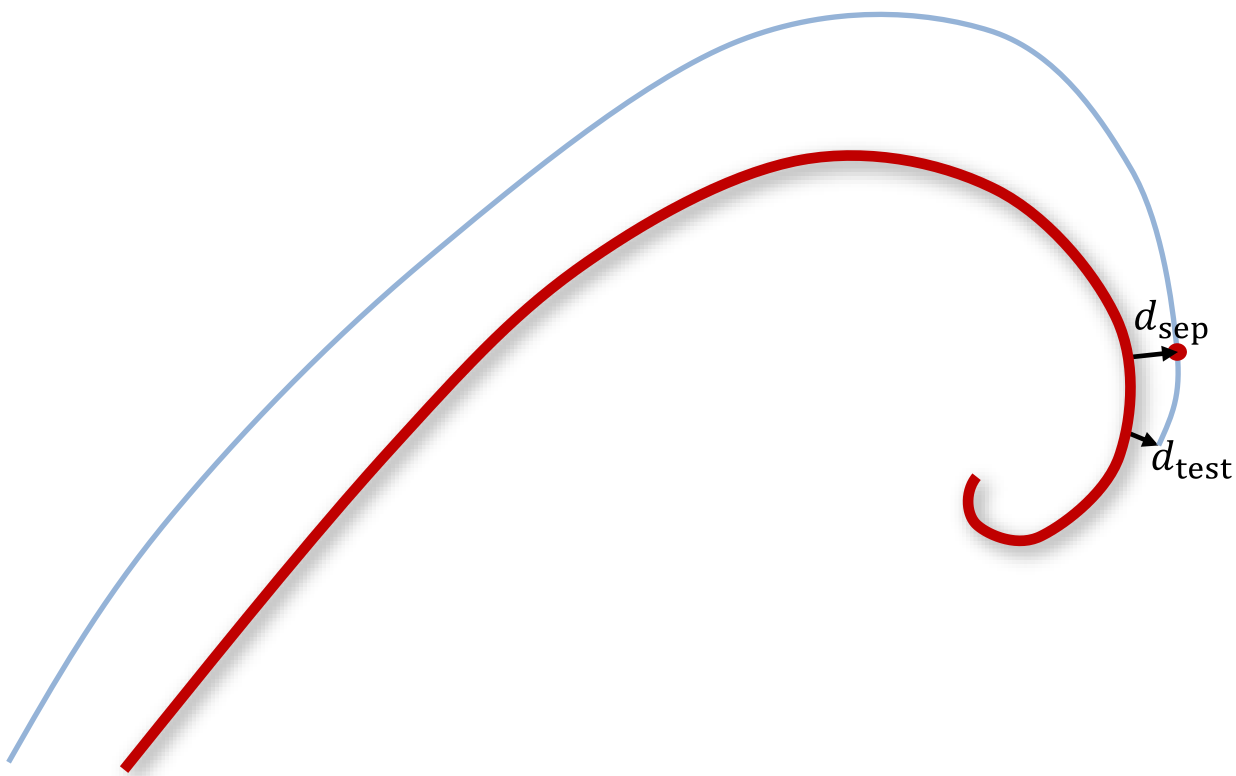
Evenly-spaced stream lines

[Jobard & Lefer 97]

Stream line placement in 2D

SIEMENS
Ingenuity for life

- Evenly-spaced streamlines
- Idea: stream lines should not get too close to each other
  - Choose seed point with distance d_{sep} from existing stream line
  - Forward- and backward-integration until distance d_{test}

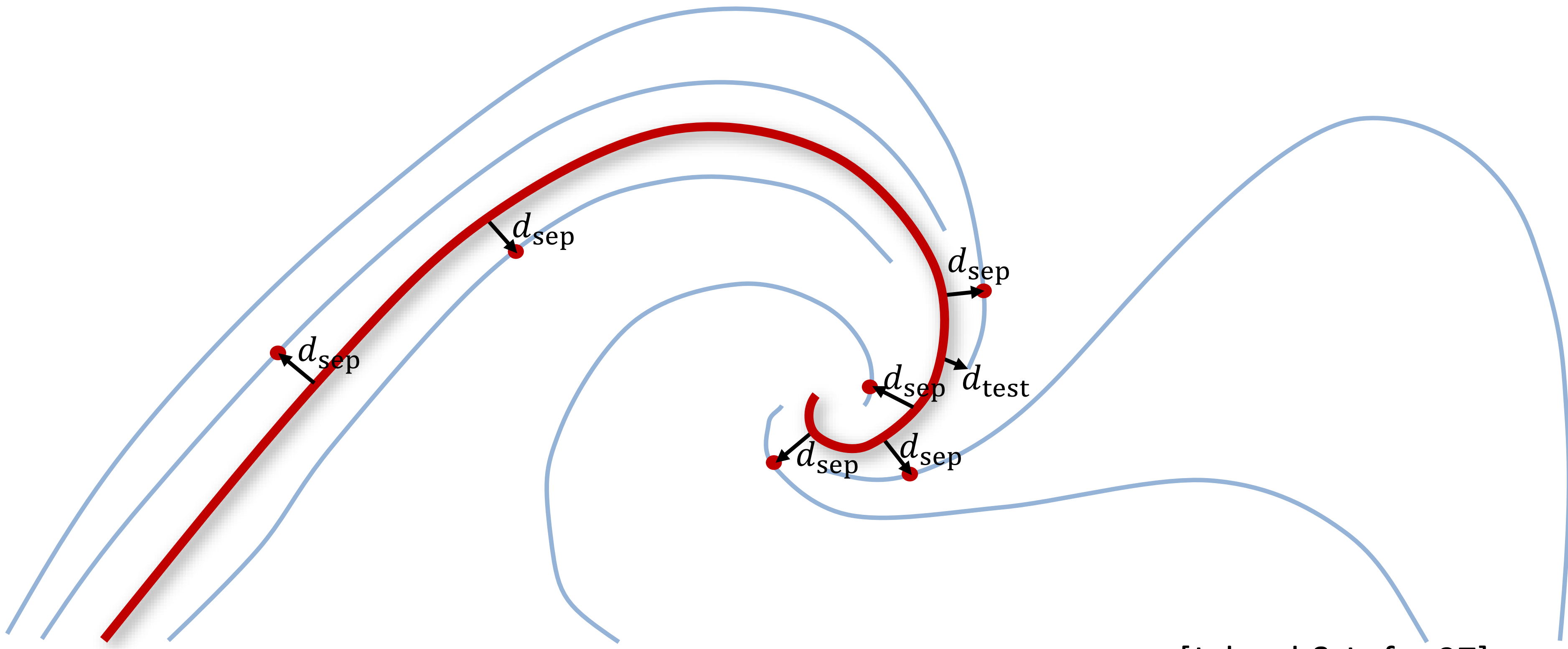


[Jobard & Lefer 97]

Stream line placement in 2D

SIEMENS
Ingenuity for life

- Evenly-spaced streamlines
- Idea: stream lines should not get too close to each other
  - Choose seed point with distance d_{sep} from existing stream line
  - Forward- and backward-integration until distance d_{test}



[Jobard & Lefer 97]

Stream line placement in 2D

Algorithm:

```
Compute initial stream line and put into the queue

Initial stream line becomes current stream line

While not finished do:
   – Try: get new seed point with distance d_sep from
     current stream line
   – If successful then compute new stream line and
     put into queue
   – Else If no more stream lines in queue then exit loop
   – Else next stream line in queue becomes current
     streamline

EndWhile
```

[Jobard & Lefer 97]

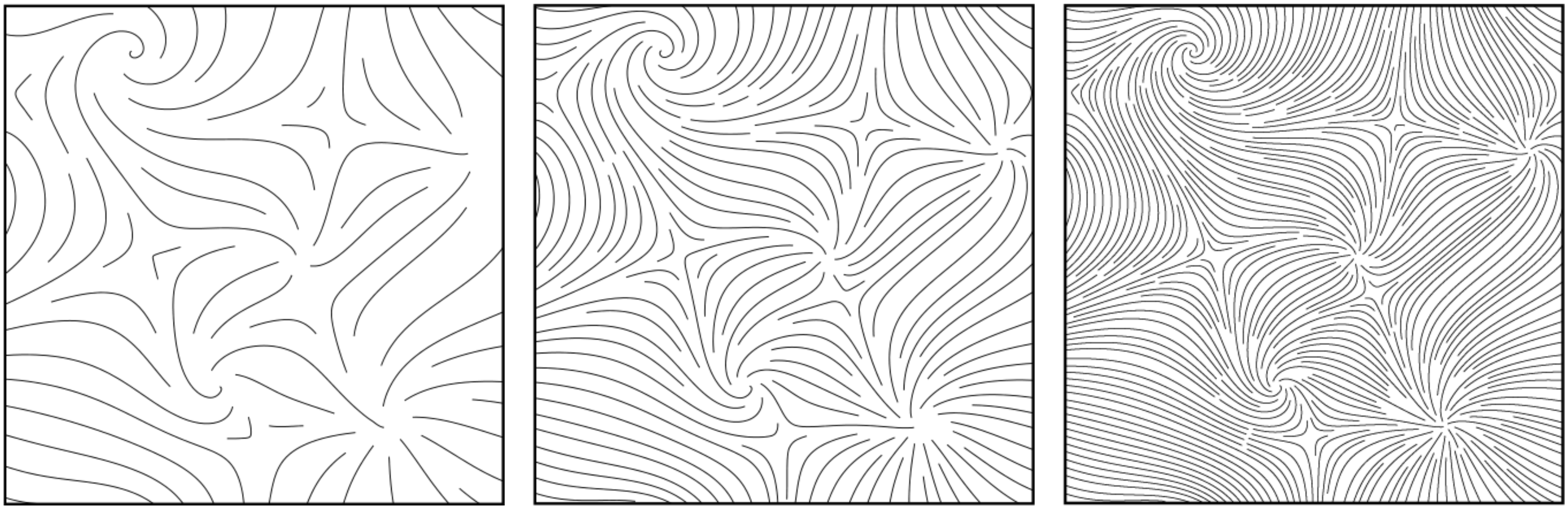
Stream line placement in 2D

- When to stop stream line integration
  - When distance to neighboring stream line $\leq d_{\text{test}}$
  - When stream line leaves flow domain
  - When stream line runs into fixed point $(\boldsymbol{v}(\boldsymbol{x}^*) = 0)$
  - When stream line gets too close to itself
  - After a certain number of maximal steps

[Jobard & Lefer 97]

Stream line placement in 2D

- Variations of d_{sep} in relation to image width

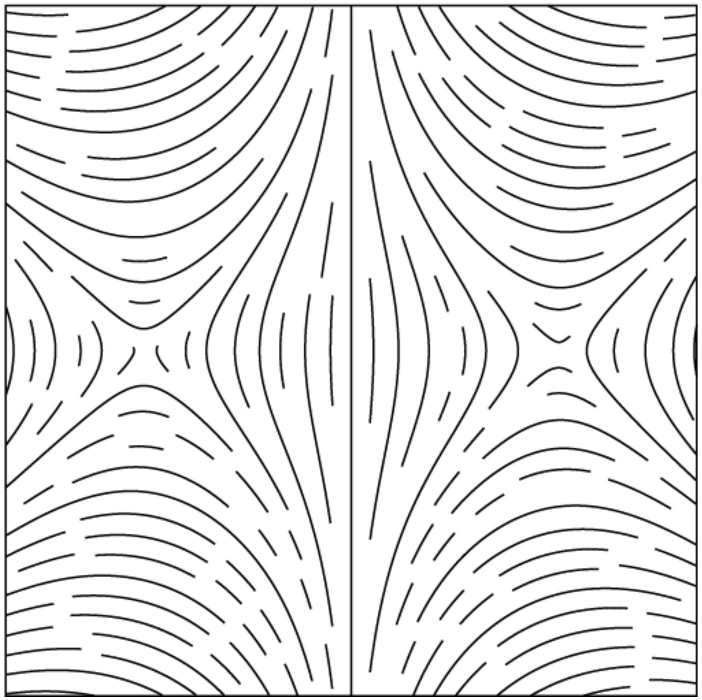


6% 3% 1.5%

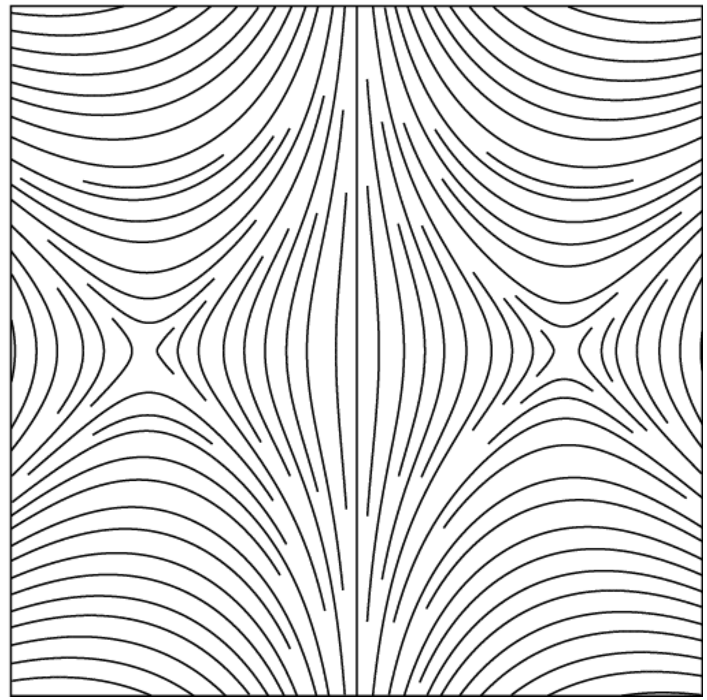
[Jobard & Lefer 97]

Stream line placement in 2D

- Variations of d_{test}



$d_{\text{test}} = 0.9\ d_{\text{sep}}$

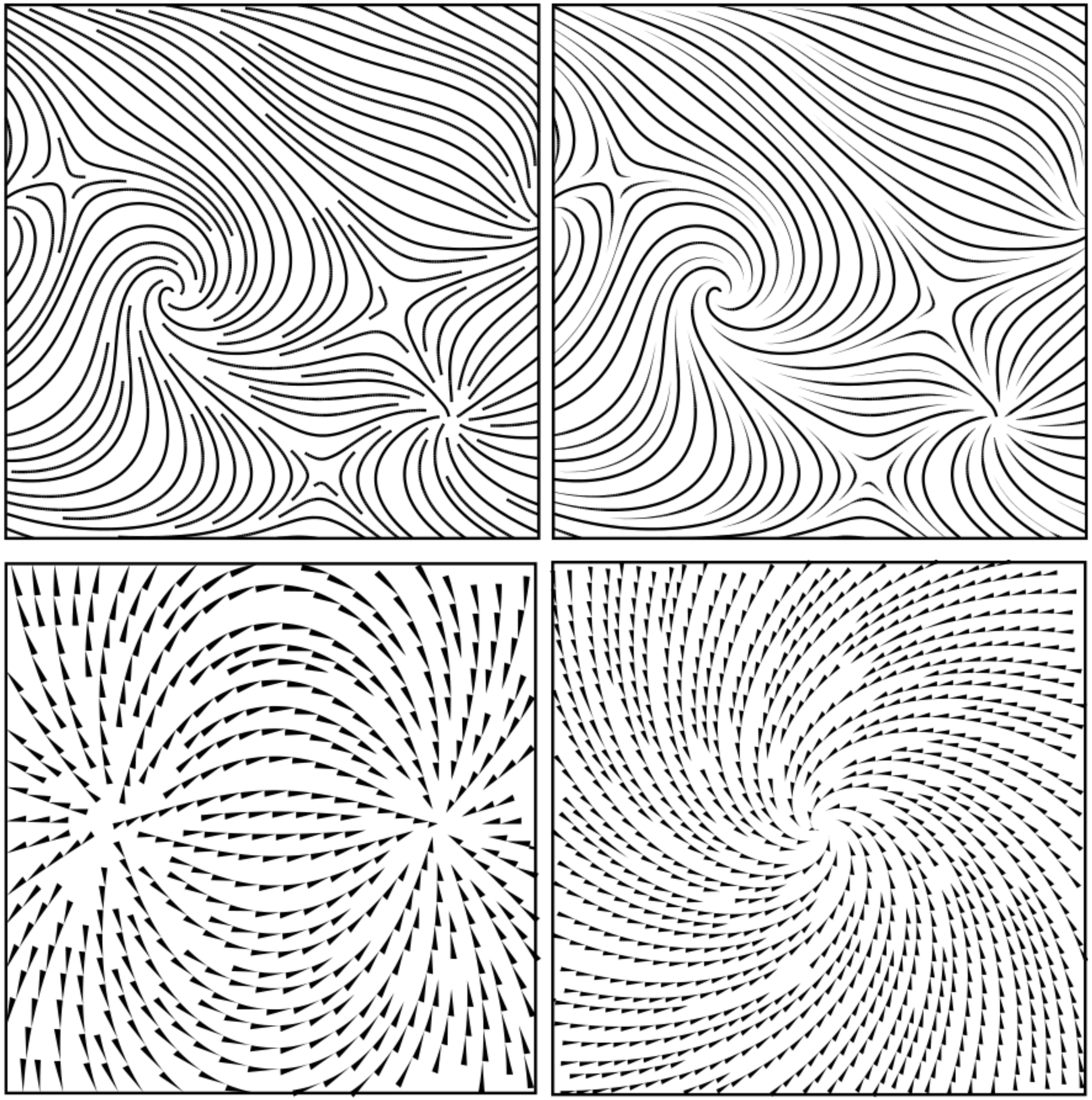


$d_{\text{test}} = 0.5\ d_{\text{sep}}$

[Jobard & Lefer 97]

Stream line placement in 2D

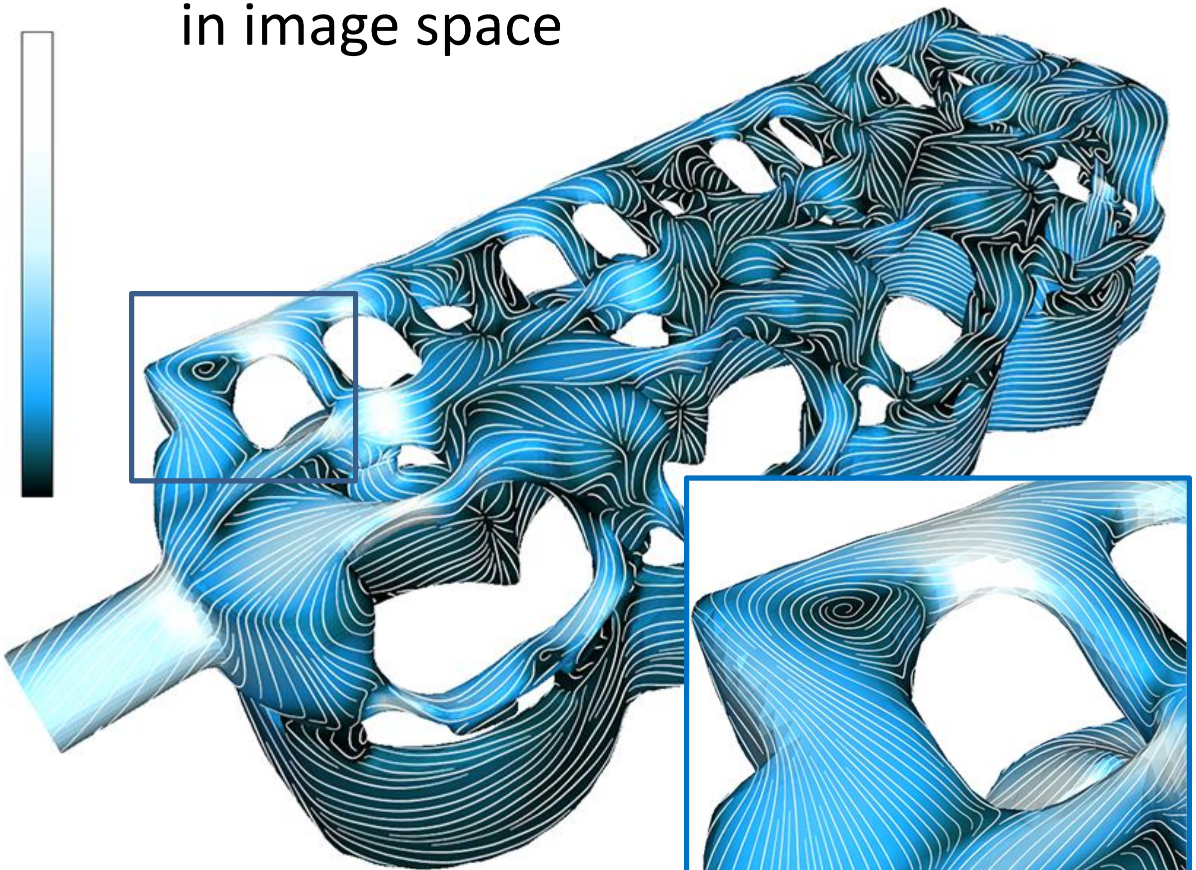
- Change thickness in relation to distance d
  - If $d \geq d_{\text{sep}}$: 1.0
  - If $d < d_{\text{sep}}$: $\frac{d - d_{\text{test}}}{d_{\text{sep}} - d_{\text{test}}}$
- Directional glyphs



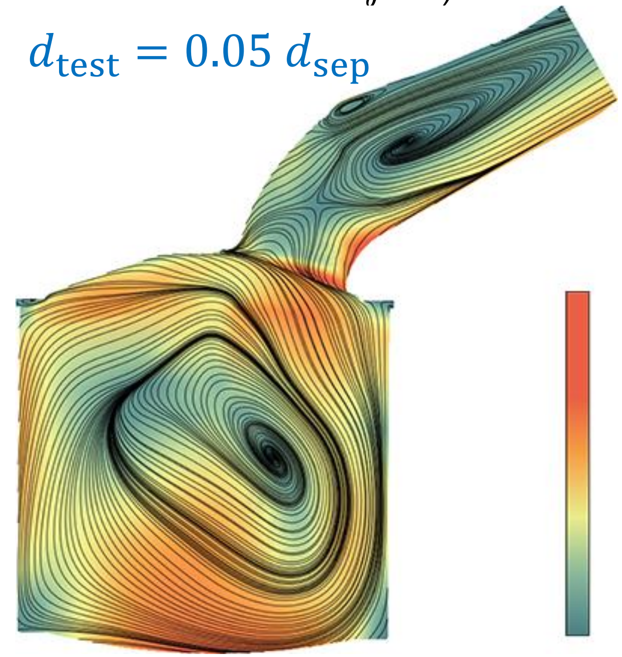
[Jobard & Lefer 97]

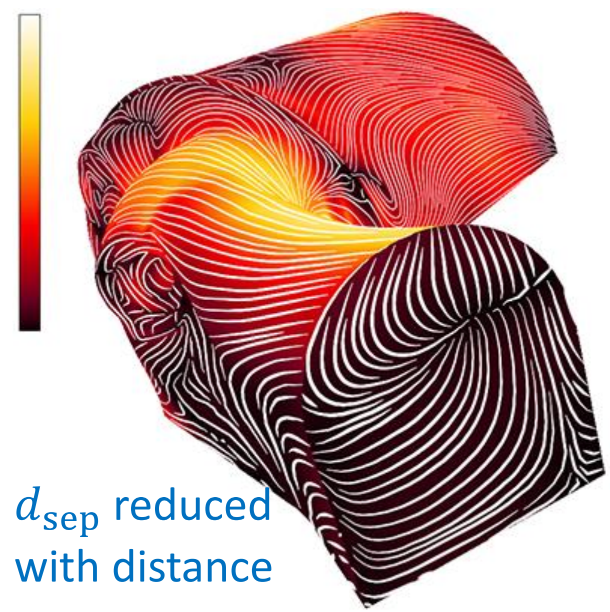
Stream line placement on surfaces

- Image-space technique
  - Vector field is first projected to 2D image
  - Seeding and integration happen in image space



[Spencer et al. 09]





Stream line placement on surfaces

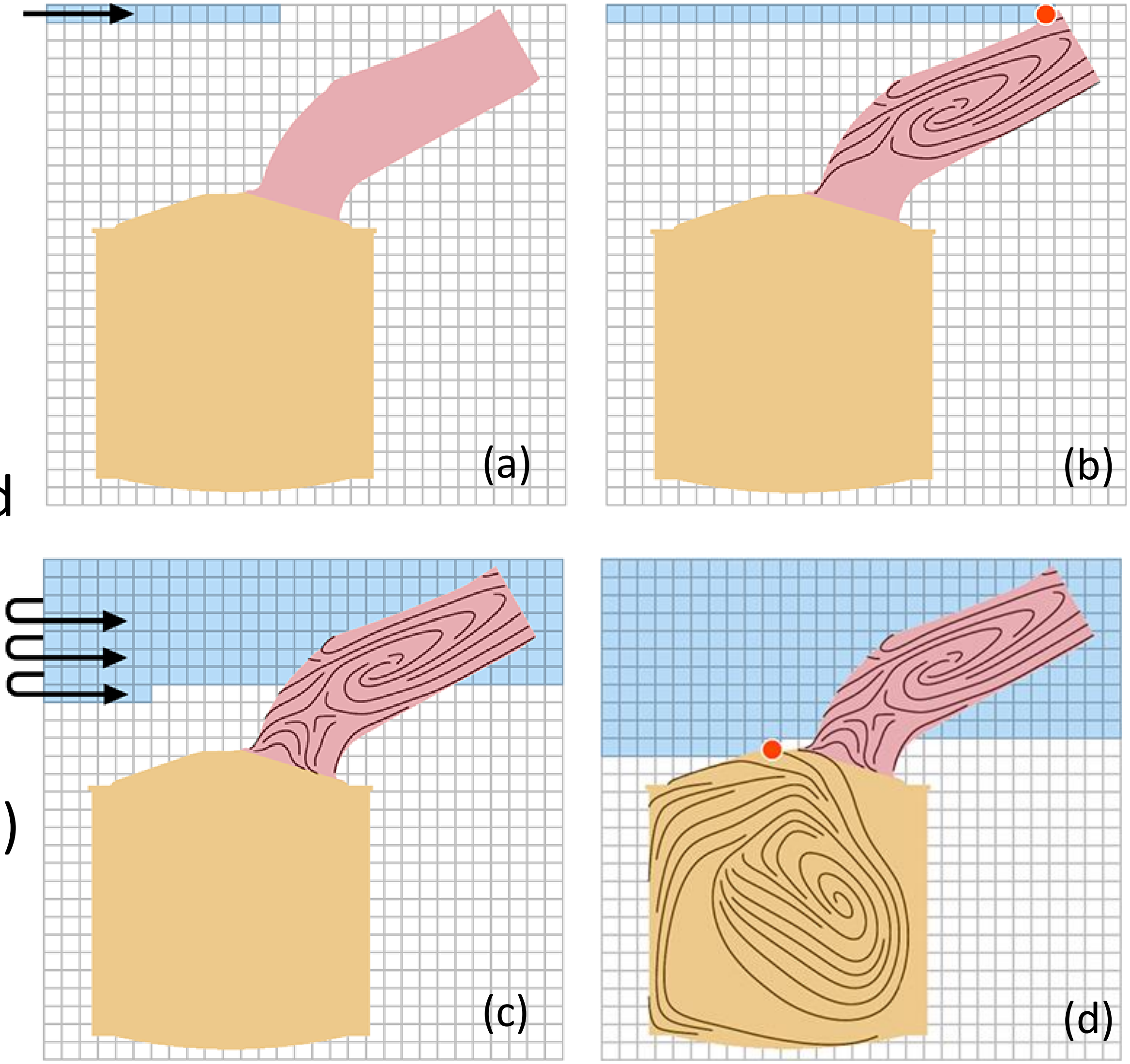
SIEMENS
Ingenuity for life

- Vector field is first projected to 2D image

a) 2D image is scanned at intervals d_{sep}

b) Seedpoint is found & stream lines are traced in that region

c) Scanning continues until seedpoint in new region is found (d)



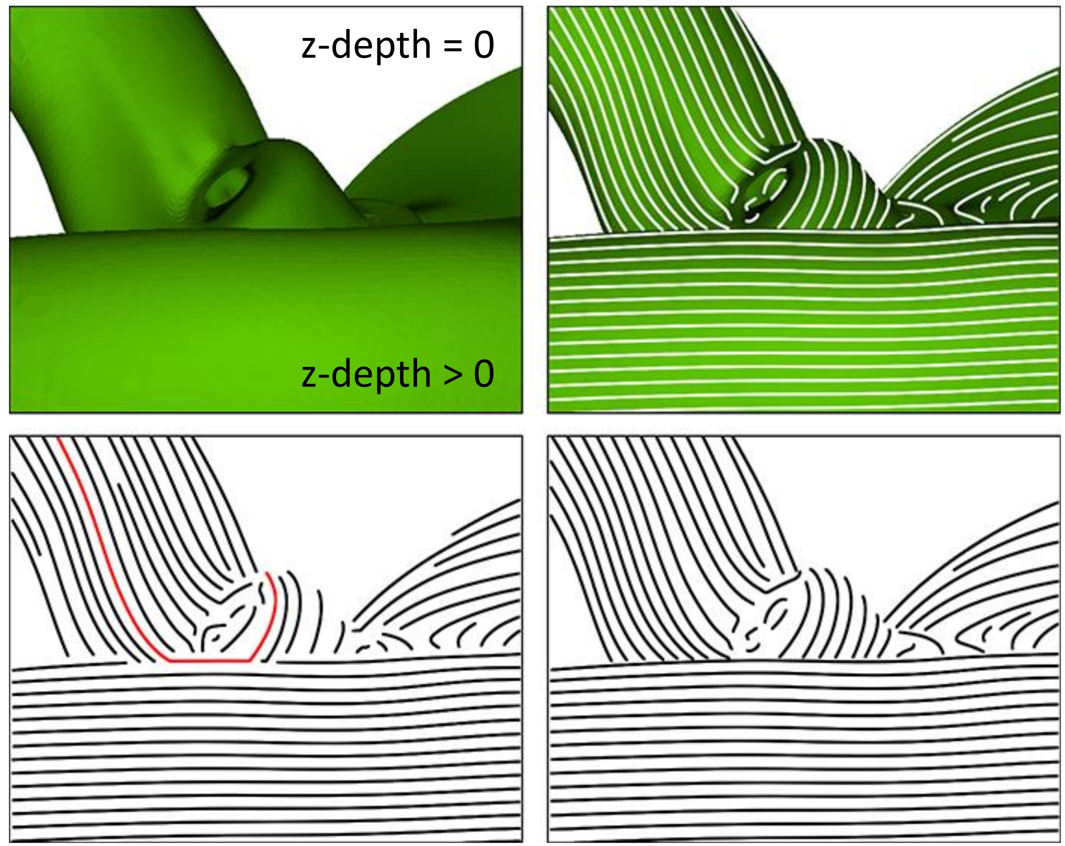
[Spencer et al. 09]

Stream line placement on surfaces

SIEMENS
Ingenuity for life

- Discontinuity detection
  - Stop stream line integration when z-depth drops to zero (edge of model)

  … or when z-depth changes too abruptly (edge of overlapping regions)



Without detection: stream lines run off edge of surface (red)

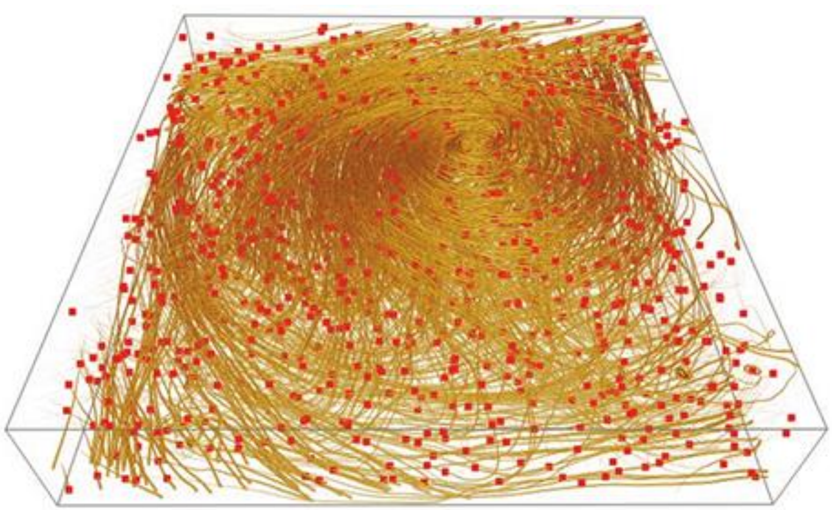
With detection: underlying surface is better reflected

[Spencer et al. 09]

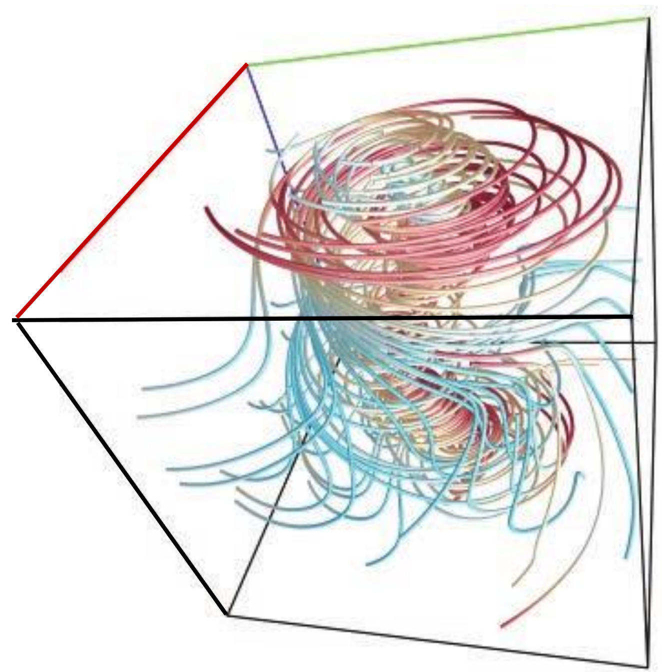
Stream line placement in 3D

SIEMENS
Ingenuity for life

- Placement of seeds directly affects the visual quality
  - Too many: scene cluttering (occlusion)
  - Too little: no patterns forming
- Seeding has to be at the right place and in the right amount!



A bad seeding example



[Tao et al. 13]

Finding best stream lines and best view point from many candidates

Stream line placement

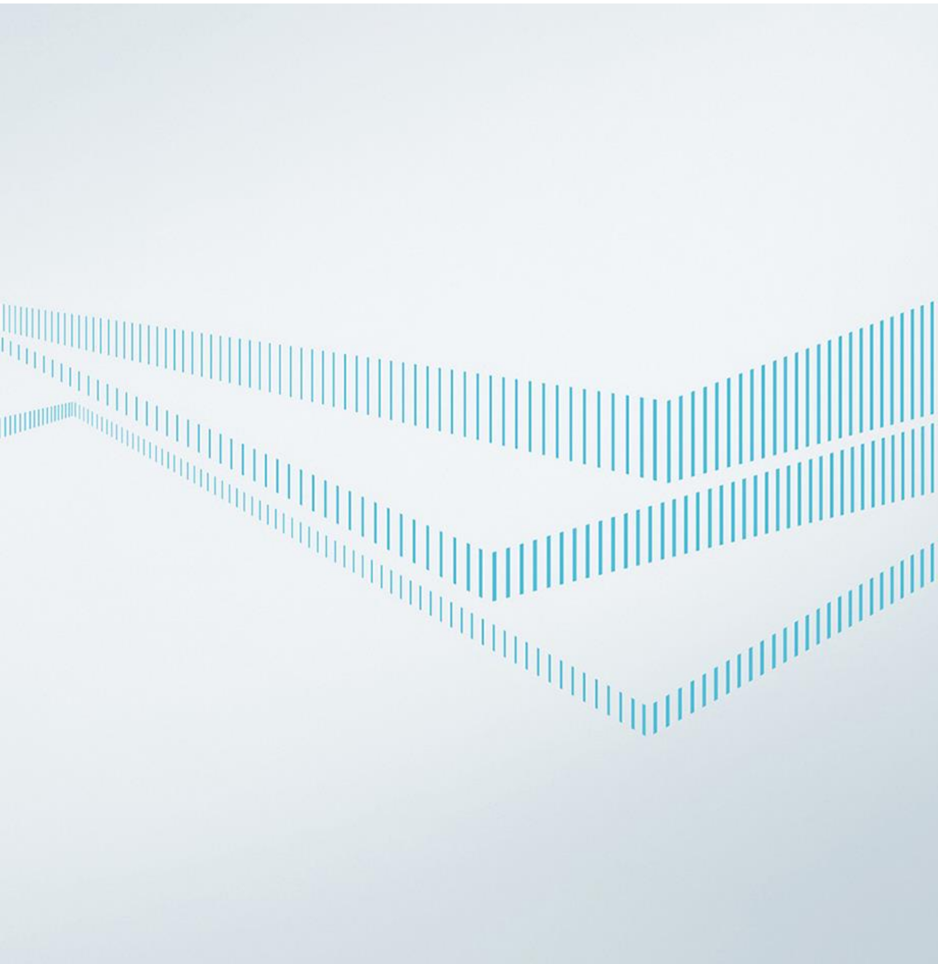
- Summary
  - Irregular results when using regular grid or random seeding
  - Stream lines should not get too close to each other
  - Seeding/placement techniques typically limed to 2D or surfaces and stationary vector fields
  - Not many methods for 3D and/or time-varying flows yet

Acknowledgements

- Helwig Hauser
- Andrea Brambilla
- Daniel Weiskopf
- Ronald Peikert
- Christoph Garth
- Alexandru C. Telea
- Many more

Contact information

SIEMENS
Ingenuity for life



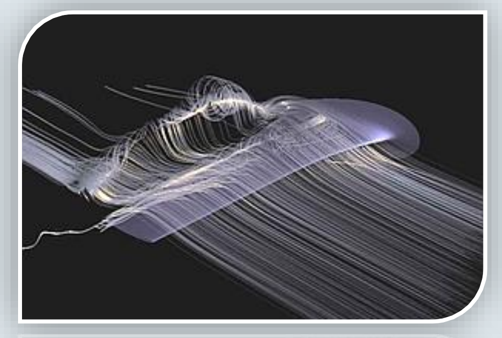
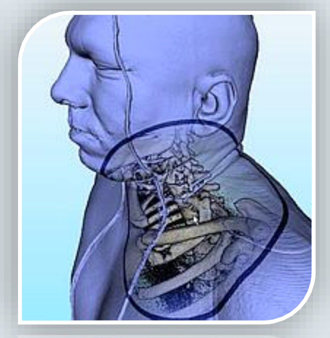
Dr. Johannes Kehrer

Siemens AG
Technology
T DAI HCA-DE
Otto-Hahn-Ring 6
81739 München, Deutschland

E-mail:
kehrer.johannes@siemens.com

Internet
siemens.com/innovation

Intranet
intranet.ct.siemens.com

Visual Data Analytics
Flow Visualization II

Dr. Johannes Kehrer

Disclaimer

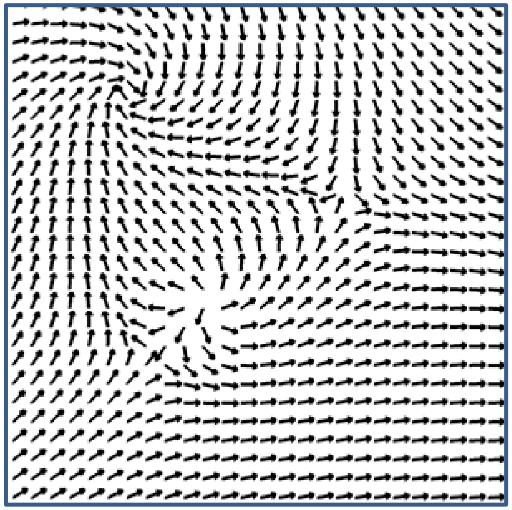
These lecture slides and materials are for personal use by students and are intended for educational purposes only. Distribution or reproduction of the material is prohibited.

The views and opinions expressed by any individual during the course of these lecturers are their own and do not necessarily represent the views, opinions, or positions of Siemens or the Technical University of Munich.

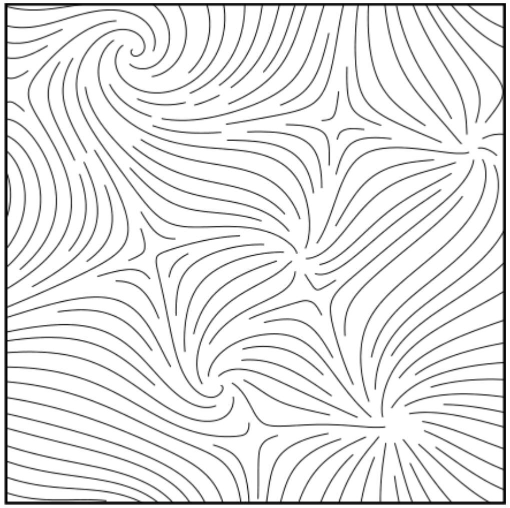
Although reasonable efforts have been made to ensure the information in these lecture slides is current and accurate, we do not assume any responsibility for the content of the posted material.

These slides also contain links to websites of third parties. As the content of these websites is not under our control, we cannot assume any liability for such external content.

Flow visualization – Approaches



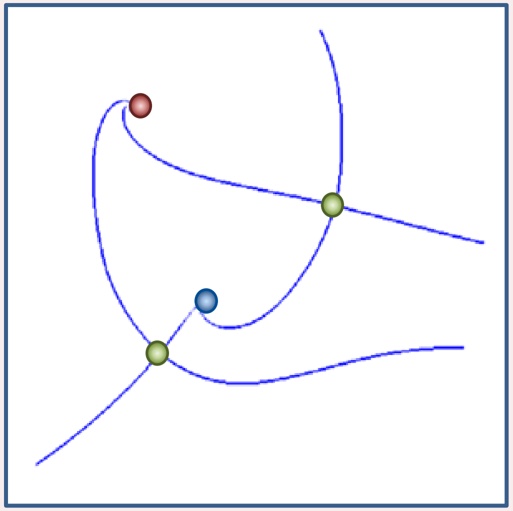
Direct flow visualization
(arrows, color coding, …)

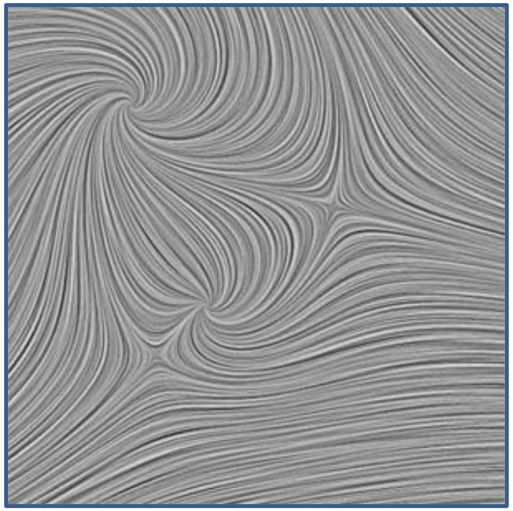


Geometric flow vis.
(stream lines/surfaces, …)

Sparse (feature-based) vis.

- Global computation of flow features
- Vortices, shockwaves, vector field topology



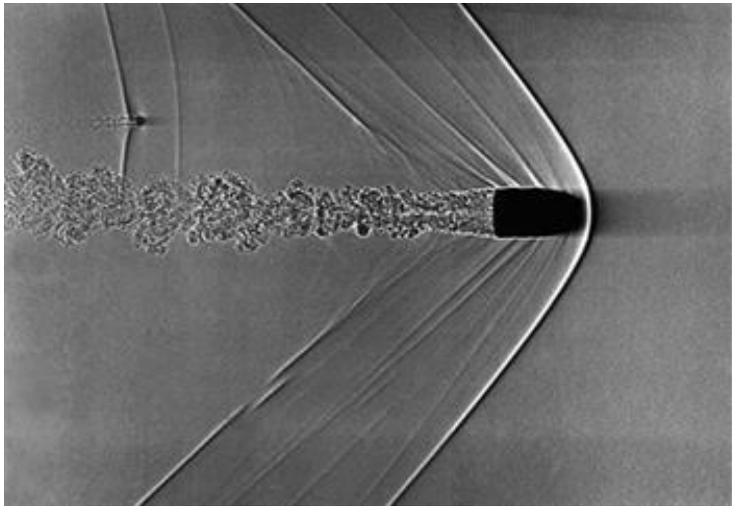


Dense (texture-based)

Flow features

SIEMENS
Ingenuity for life

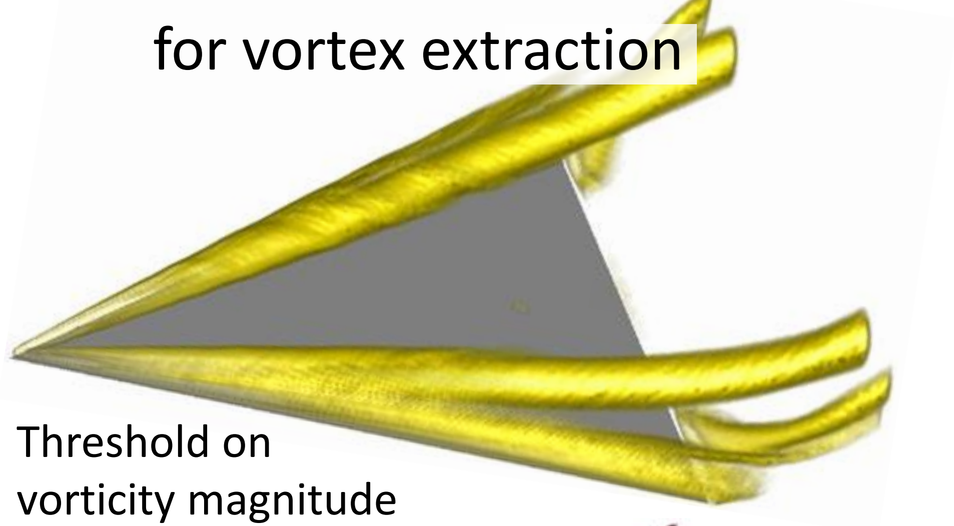
- Vortices
  - One of most prominent features
  - Important in many applications (turbulent flows)
  - No formal, well accepted definition yet ("something swirling")
- Shock waves
  - Characterized by sharp discontinuities in flow attributes (pressure, velocity magnitude, …)



Bullet traveling through air at about 1.5 times sound speed

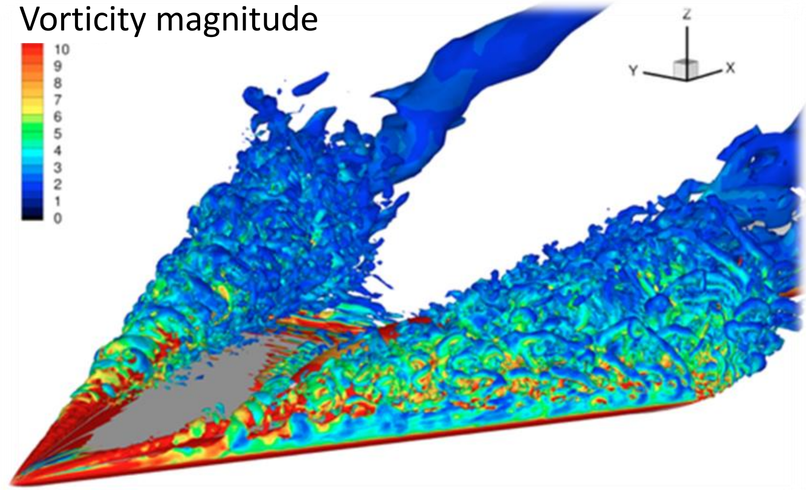
Flow features

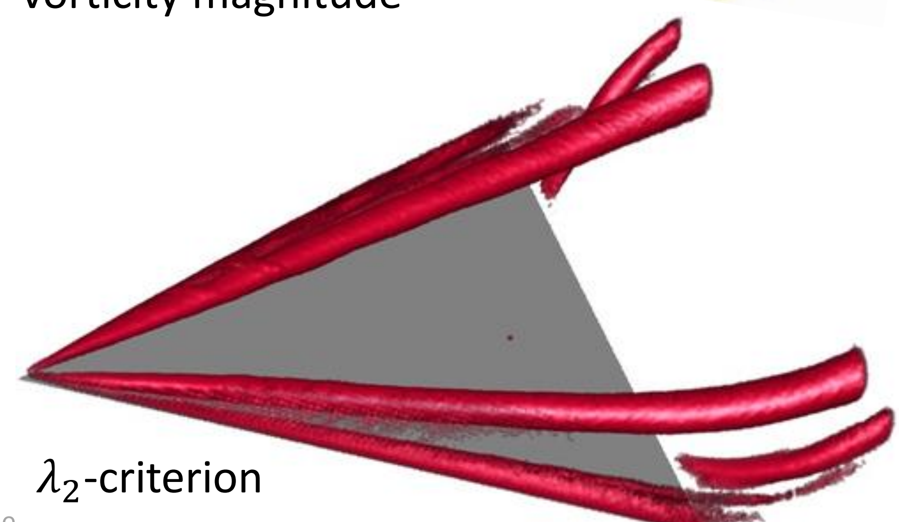
- Vortices around delta wing
  - Different metrics used for vortex extraction



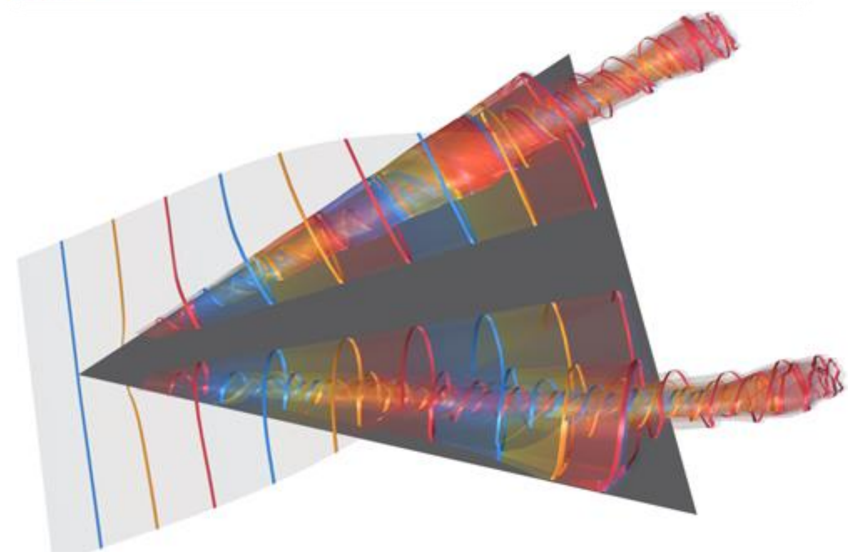
Threshold on vorticity magnitude

Vorticity magnitude





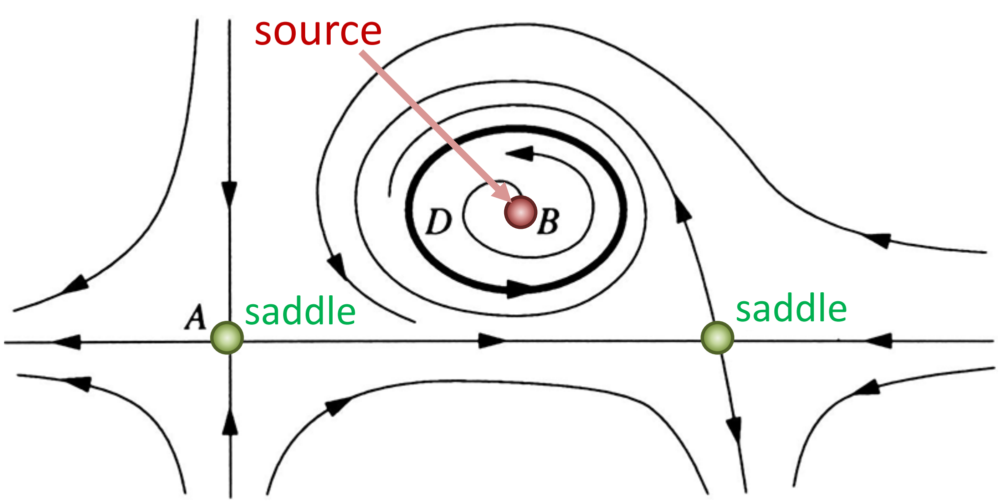
λ_2-criterion

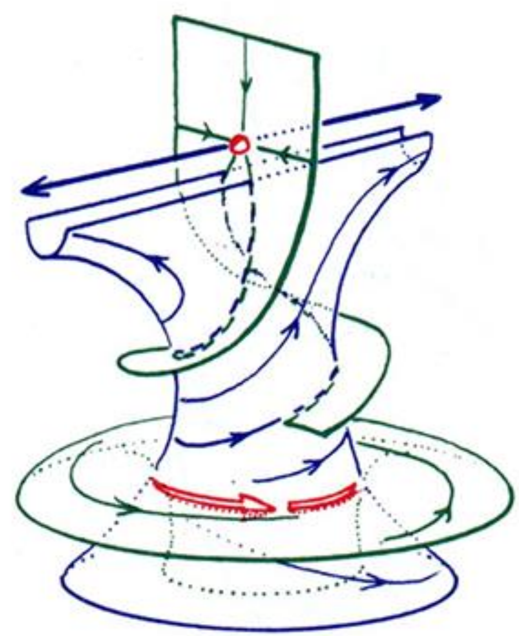


Vector field topology

SIEMENS
Ingenuity for life

- Idea: Do not draw “all” stream lines, but only the “important” ones
- Show only topological skeleton
  - Connection of critical points
  - Characterization of global flow structures

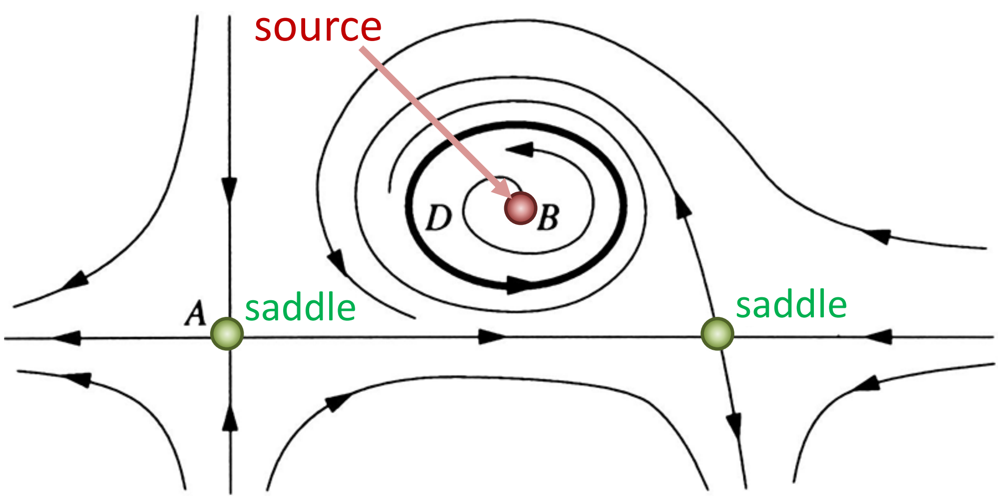


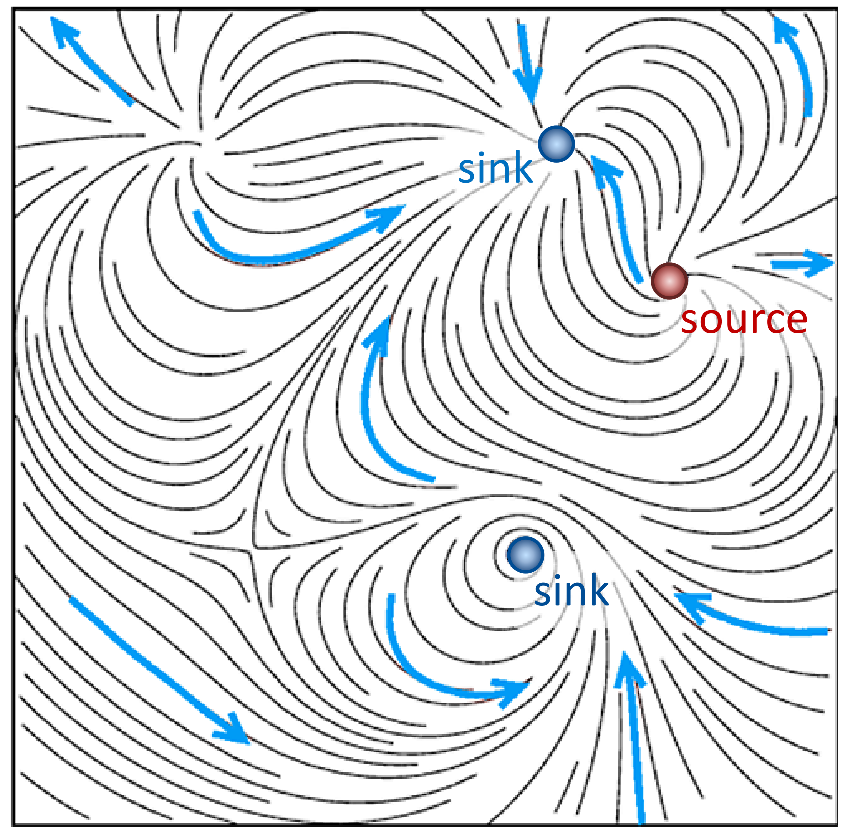


Vector field topology

SIEMENS
Ingenuity for life

- Critical points: singularities in vector field such that $\boldsymbol{v}(\boldsymbol{x}^*) = 0$
  - Points where magnitude of vector goes to zero and direction of vector is undefined
  - Stream lines reduced to single point

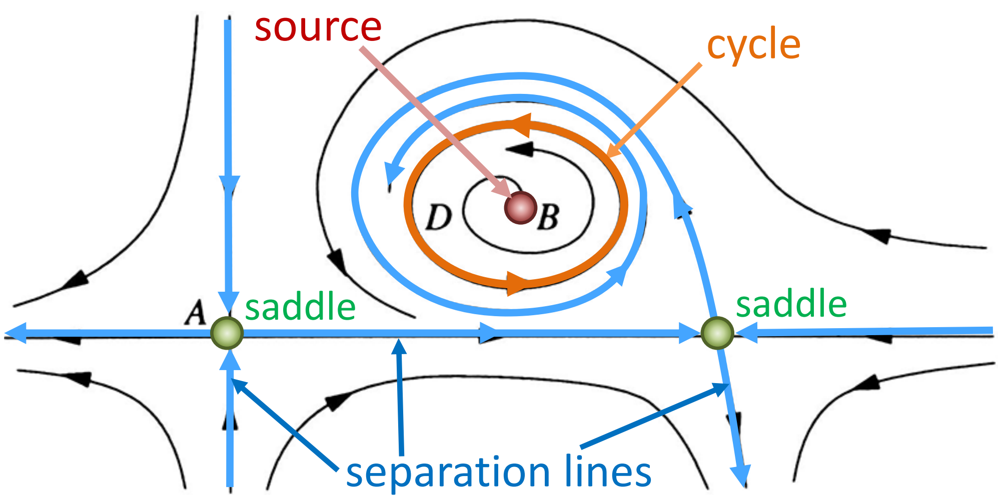


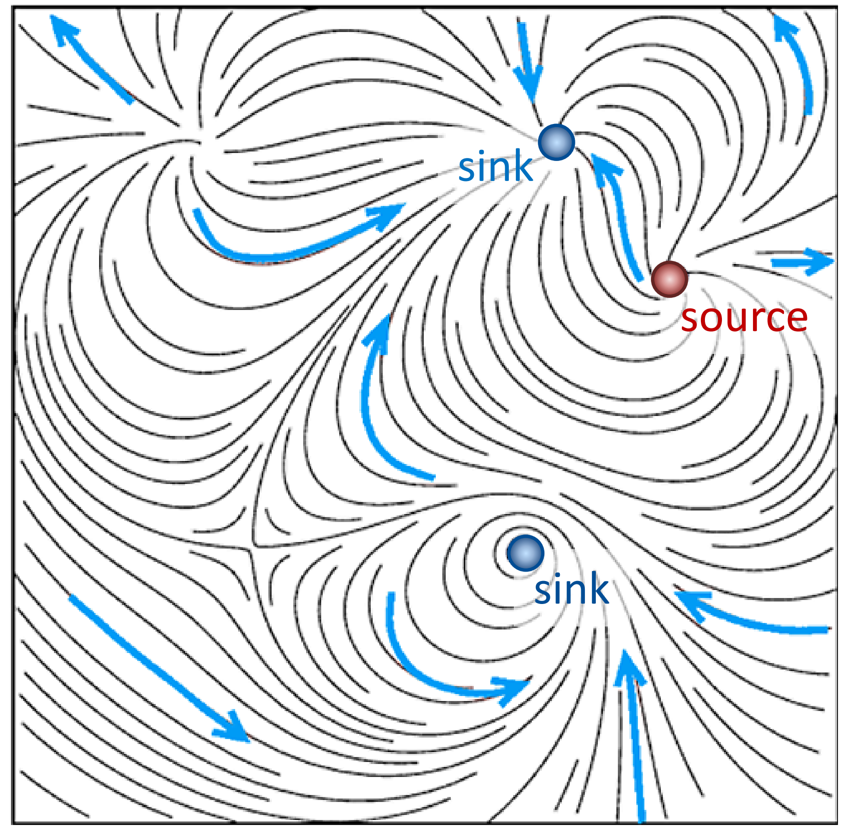


Vector field topology

SIEMENS
Ingenuity for life

- Critical points: singularities in vector field such that $\boldsymbol{v}(\boldsymbol{x}^*) = 0$
  - Points where magnitude of vector goes to zero and direction of vector is undefined
  - Stream lines reduced to single point
  - Type of critical point determines flow pattern around it

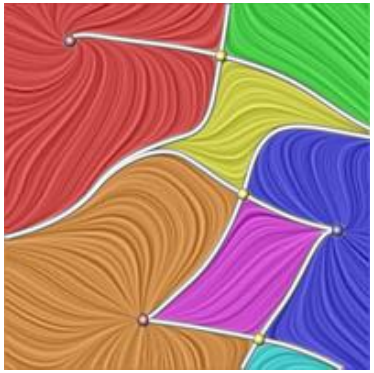


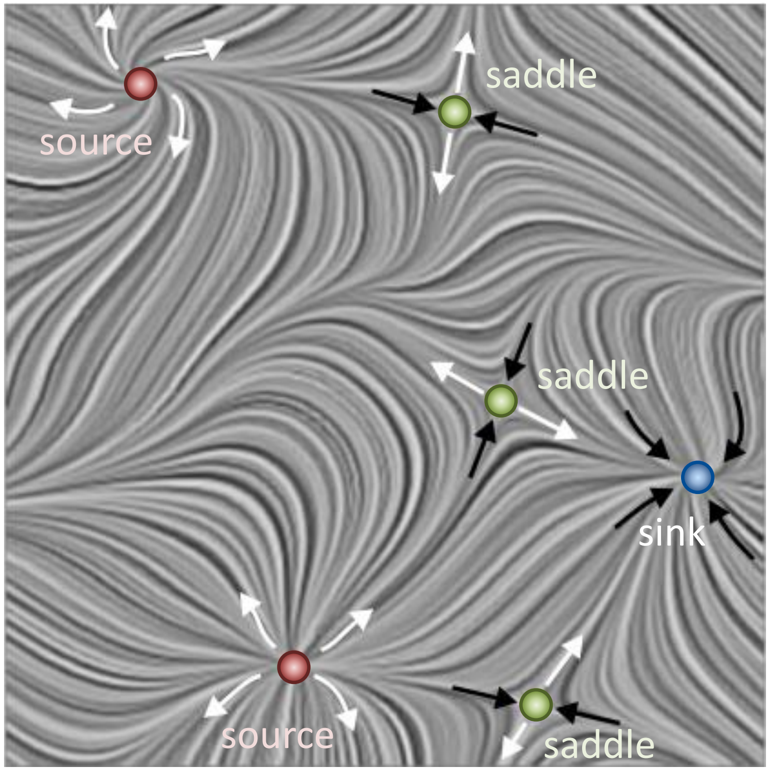


Vector field topology

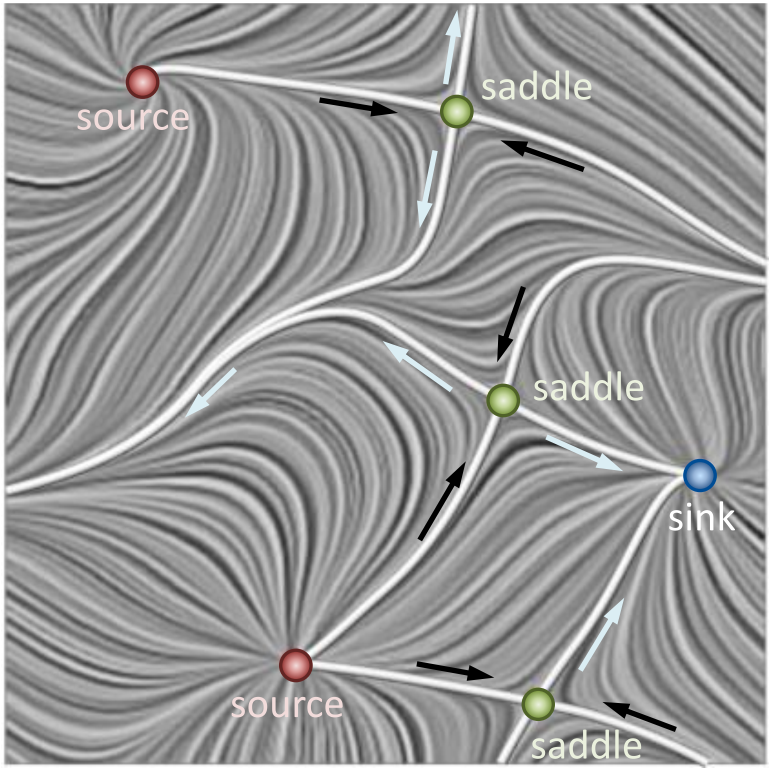
SIEMENS
Ingenuity for life

- Topological skeleton / graph
  - Nodes: critical points
  - Edges: separation lines and cycles
  - Flow divided into regions with similar properties





Critical points



Critical points + separation lines

[Weinkauf]

Vector field topology (2D)

SIEMENS
Ingenuity for life

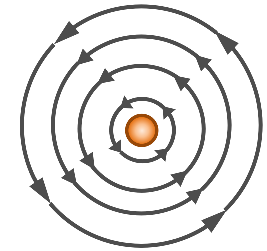
- How to find critical points $\boldsymbol{x}^*$?
  - Points where $\boldsymbol{v}(\boldsymbol{x}^*) = 0$
- How to classify critical points $\boldsymbol{x}^*$?
  - Jacobian matrix $\mathbf{J} = \begin{pmatrix} \frac{\partial}{\partial x}\boldsymbol{v}_x & \frac{\partial}{\partial y}\boldsymbol{v}_x \\ \frac{\partial}{\partial x}\boldsymbol{v}_y & \frac{\partial}{\partial y}\boldsymbol{v}_y \end{pmatrix}$ governs the behavior near $\boldsymbol{x}^*$
  - For each $\boldsymbol{x}^*$, calculate eigenvalues λ_1, λ_2 of $\mathbf{J}$

$$\mathbf{J}\boldsymbol{u} = \lambda\boldsymbol{u}$$

$\mathbf{J}$ ... Jacobian matrix
$\boldsymbol{u}$ ... eigenvector (non-zero)
λ ... eigenvalue

Vector field topology (2D)

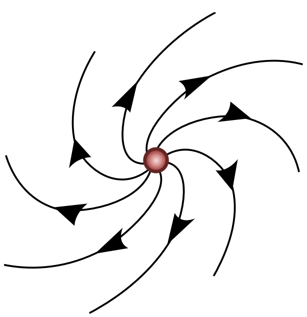
- Classify critical points by eigenvalue analysis of $\mathbf{J}$
  - λ_1, λ_2 both positive → local repulsion (source)
  - λ_1, λ_2 both negative → local attraction (sink)
  - $\lambda_1 \lambda_2 < 0$ → saddle point
  - λ_1, λ_2 both complex → rotation around $\boldsymbol{x}^*$



Center

$\mathrm{Im}(\lambda_{1,2}) \neq 0$

$\mathrm{Re}(\lambda_{1,2}) = 0$

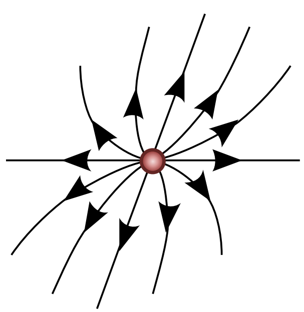


Circulating source (repelling focus)

$\mathrm{Im}(\lambda_{1,2}) \neq 0$

$\mathrm{Re}(\lambda_{1,2}) > 0$

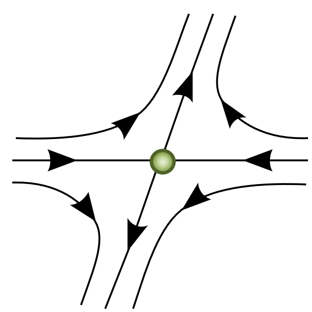
$\lambda_{1,2} = a \pm bi$



Noncirculating source (repelling node)

$\mathrm{Im}(\lambda_{1,2}) = 0$

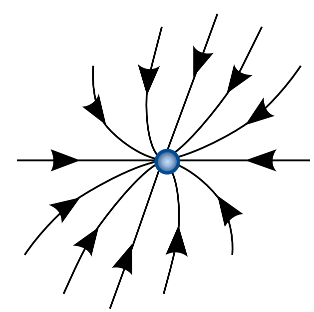
$\mathrm{Re}(\lambda_{1,2}) > 0$



Saddle point

$\mathrm{Im}(\lambda_{1,2}) = 0$

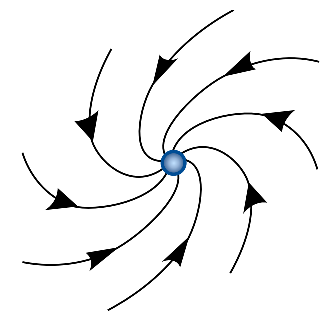
$\lambda_1 \lambda_2 < 0$



Noncirculating sink (attracting node)

$\mathrm{Im}(\lambda_{1,2}) = 0$

$\mathrm{Re}(\lambda_{1,2}) < 0$



Circulating sink (attracting focus)

$\mathrm{Im}(\lambda_{1,2}) \neq 0$

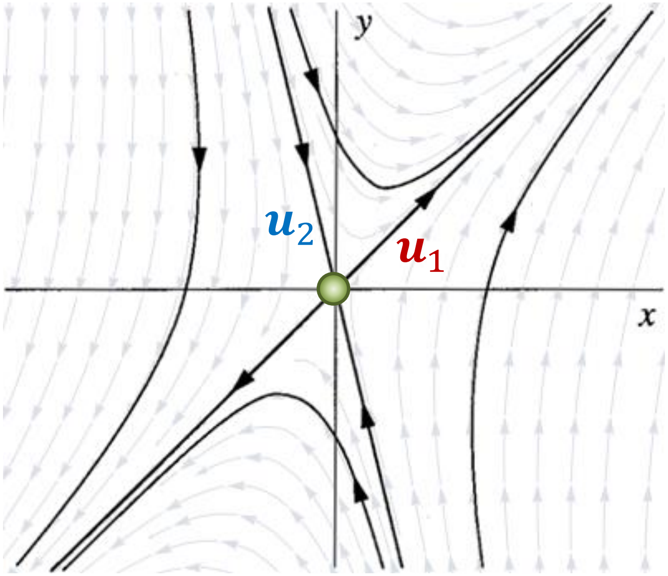
$\mathrm{Re}(\lambda_{1,2}) < 0$

$\lambda_{1,2} = a \pm bi$

Vector field topology (2D)

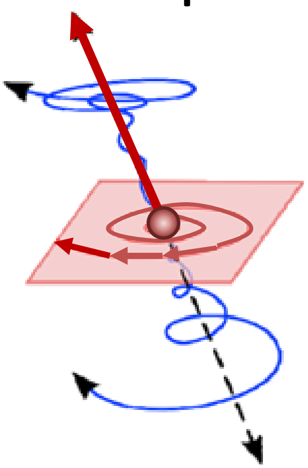
SIEMENS
Ingenuity for life

- Mapping to graphical primitives: streamlines
  - Start streamlines close to critical points
  - Initial direction along the eigenvectors $\boldsymbol{u}_1$, $\boldsymbol{u}_2$ (forward+backward integration)
- End particle tracing at
  - Other “real” critical points
  - Interior boundaries
  - Boundaries of computational domain



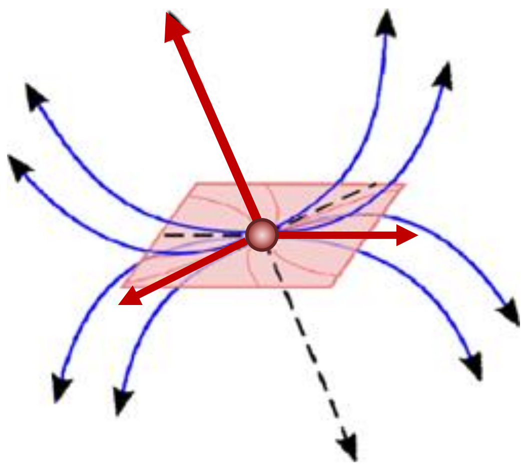
Vector field topology (3D)

- Critical points in 3D
  - More complicated
  - Line and surface separatrices exist
- Examples



Spiral source

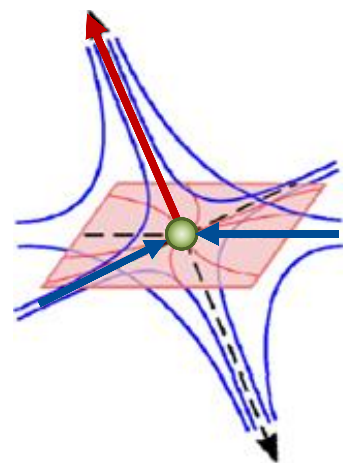
$\mathrm{Im}(\lambda_1) = 0,$
$\mathrm{Im}(\lambda_{2,3}) \neq 0$
$\mathrm{Re}(\lambda_{1,2,3}) > 0$



Noncirculating source

$\mathrm{Im}(\lambda_{1,2,3}) = 0$

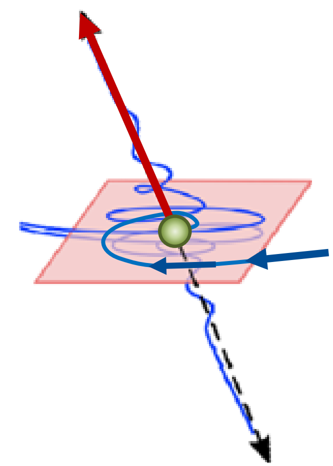
$\mathrm{Re}(\lambda_{1,2,3}) > 0$



Noncirculating saddle

$\mathrm{Im}(\lambda_{1,2,3}) = 0$

$\mathrm{Re}(\lambda_1) > 0,$
$\mathrm{Re}(\lambda_{2,3}) < 0$



Spiral saddle

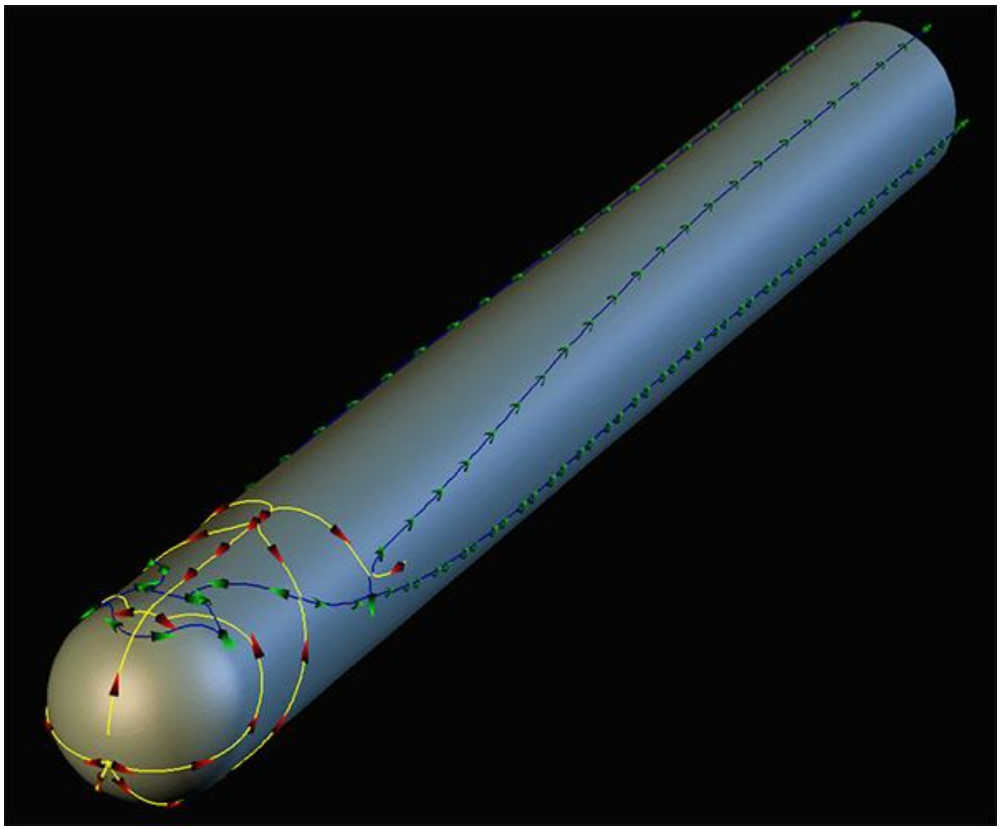
$\mathrm{Im}(\lambda_1) = 0,$
$\mathrm{Im}(\lambda_{2,3}) \neq 0$
$\mathrm{Re}(\lambda_1) > 0,$
$\mathrm{Re}(\lambda_{2,3}) < 0$

[Peikert & Sadlo 10]

Vector field topology (3D)

SIEMENS
Ingenuity for life

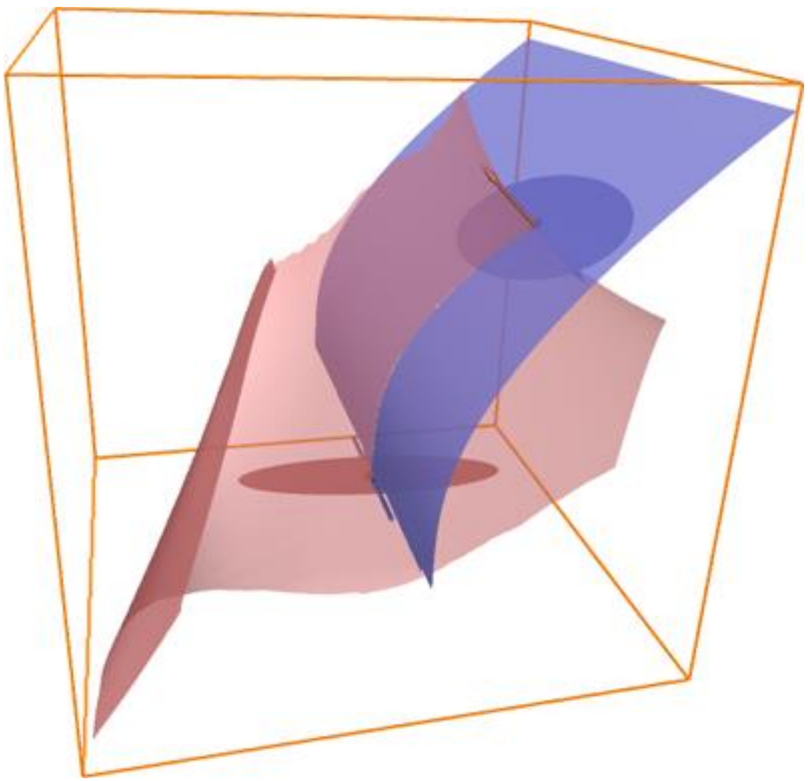
- Topology on surfaces
  - Critical points + separation lines applied to projections of vector field onto polygonal surface



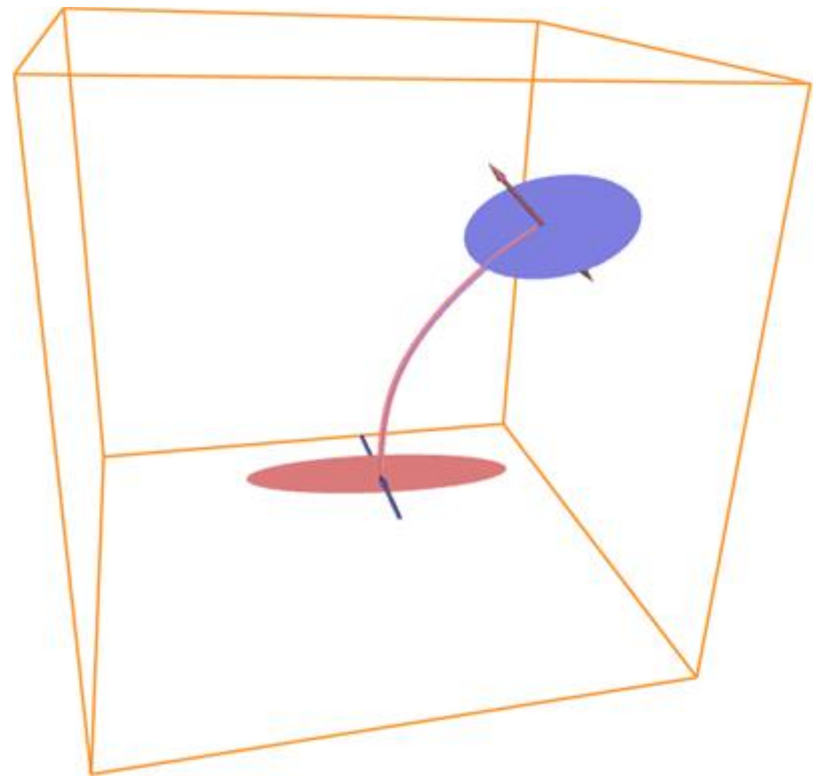
Vector field topology (3D)

SIEMENS
Ingenuity for life

- Saddle connectors in 3D



Separation surfaces of two saddles



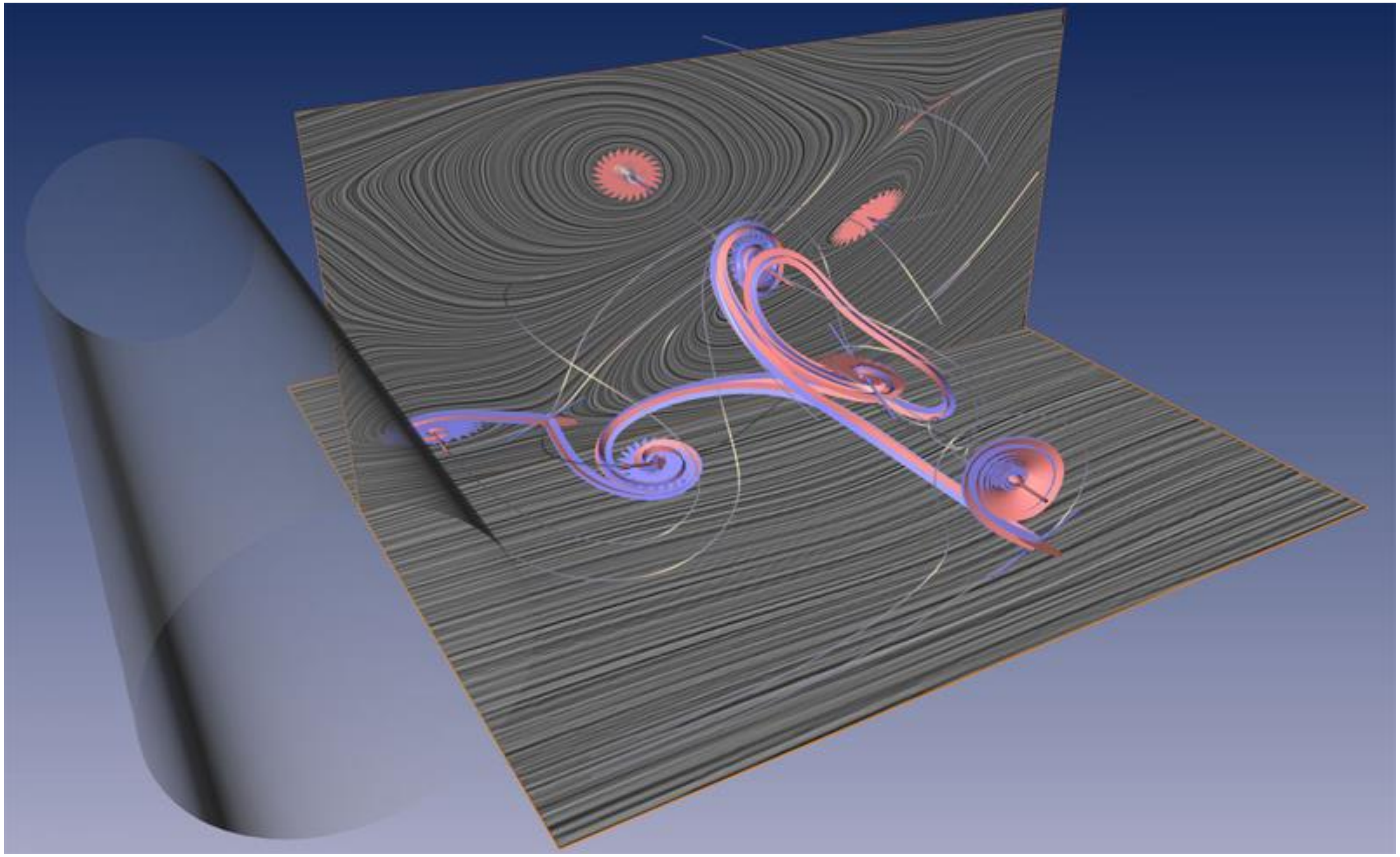
The intersection of the separation surfaces is the saddle connector

[Weinkauf & Theisel 05]

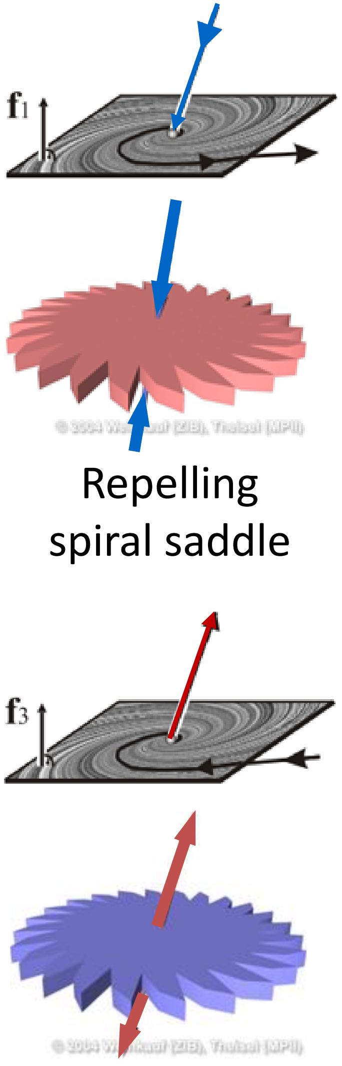
Vector field topology (3D)

SIEMENS
Ingenuity for life

- Saddle connectors in 3D



[Weinkauf & Theisel 05]



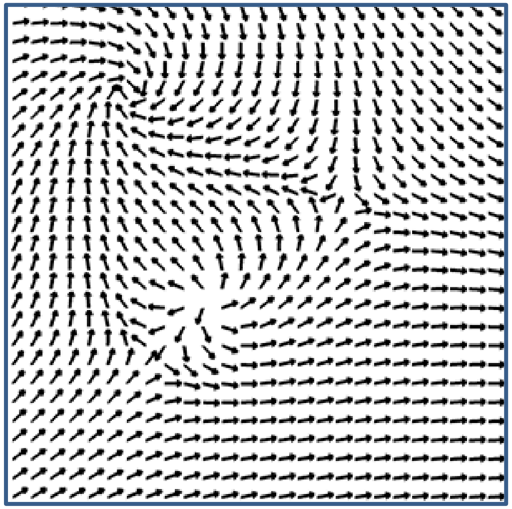
Vector field topology

SIEMENS
Ingenuity for life

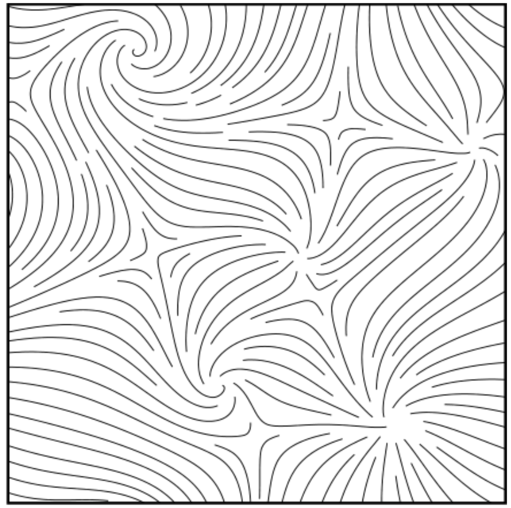
- Summary
  - Draw only relevant stream lines (topological skeleton)
  - Partition domain into regions with similar flow features
  - Based on critical points
  - Good for 2D steady flows
  - Unsteady flows?
  - 3D?

Flow visualization – Approaches

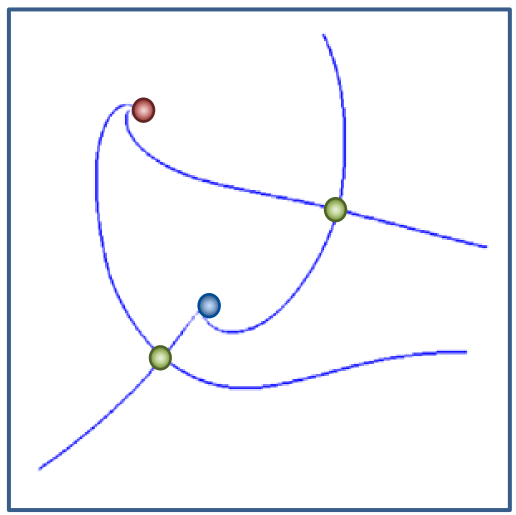
SIEMENS
Ingenuity for life



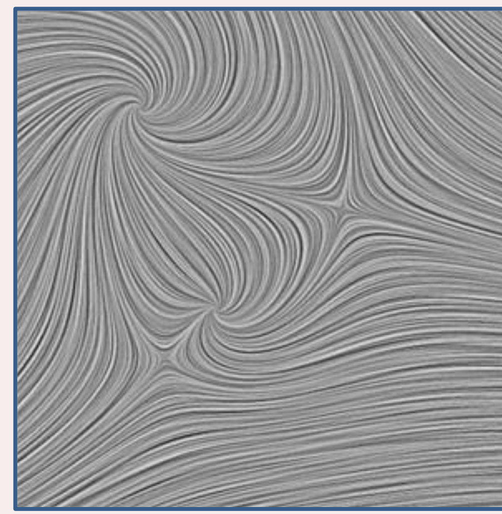
Direct flow visualization
(arrows, color coding, ...)



Geometric flow vis.
(stream lines/surfaces, ...)



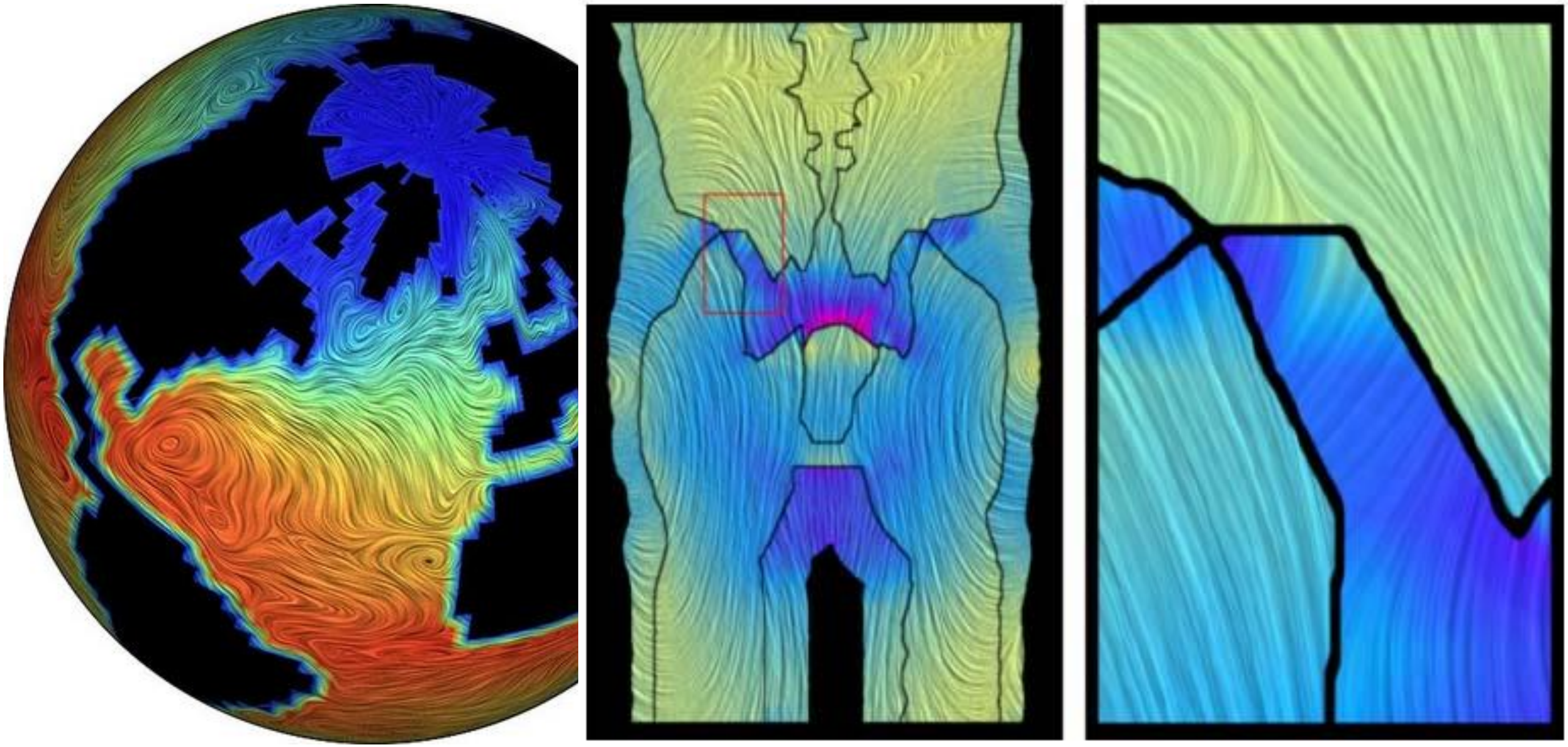
Sparse (feature-based) vis.



Dense (texture-based)

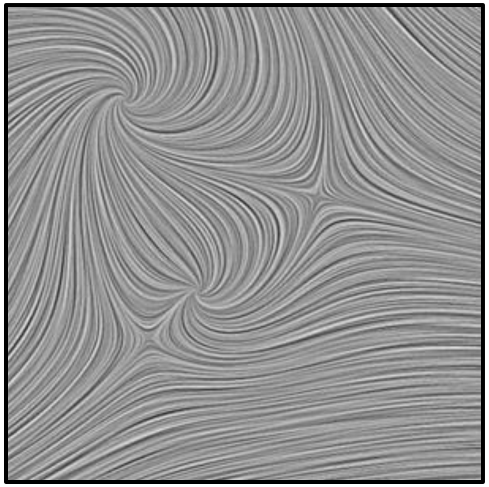
Texture-based flow visualization

- Global method to visualize vector fields

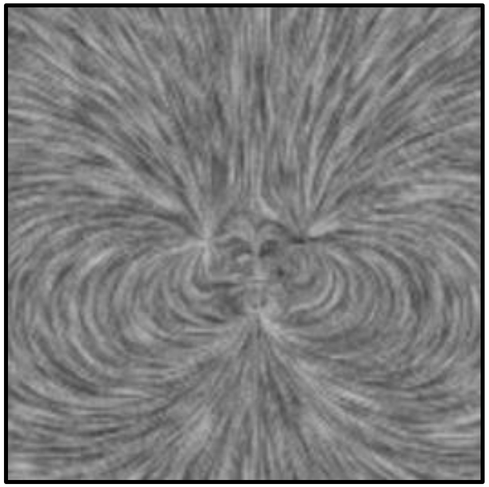


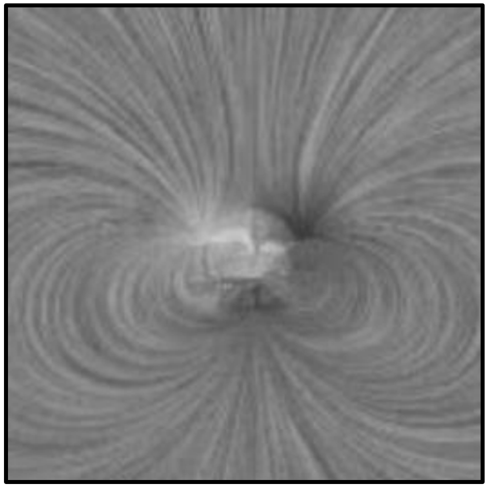
Why texture-based flow vis.

- Dense sampling
  - Better coverage of information
  - Critical point detection and classification
  - (Partially) solved problem of seeding



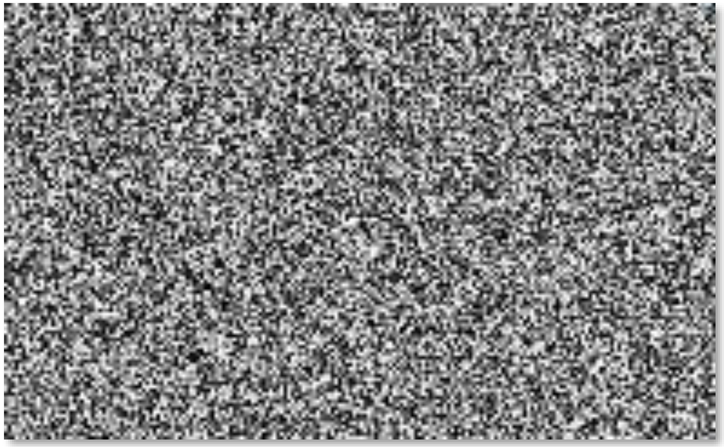
- Flexibility in visual representation
  - Good controllability of visual style
  - From line-like (crisp) to fuzzy



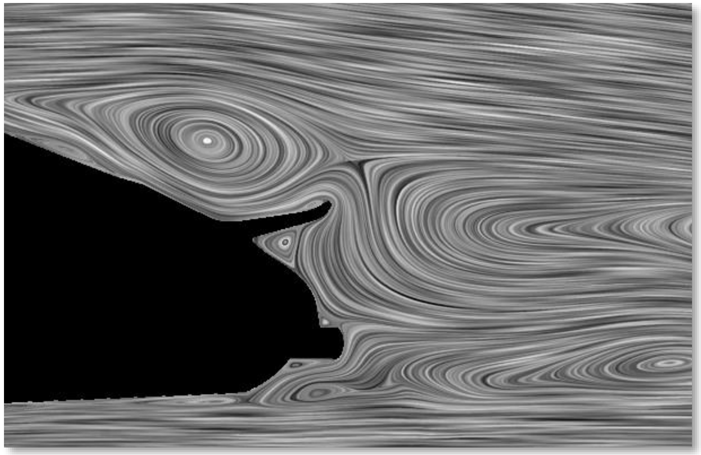


Line Integral Convolution

- Line Integral Convolution (LIC)
  - Global visualization technique (not only one particle path)
  - Start with a random texture (white noise)
  - Smear out the texture along trajectories of vector field
  - Results in low correlation between neighboring lines but high correlation along them (in flow direction)



White noise (no correlation)

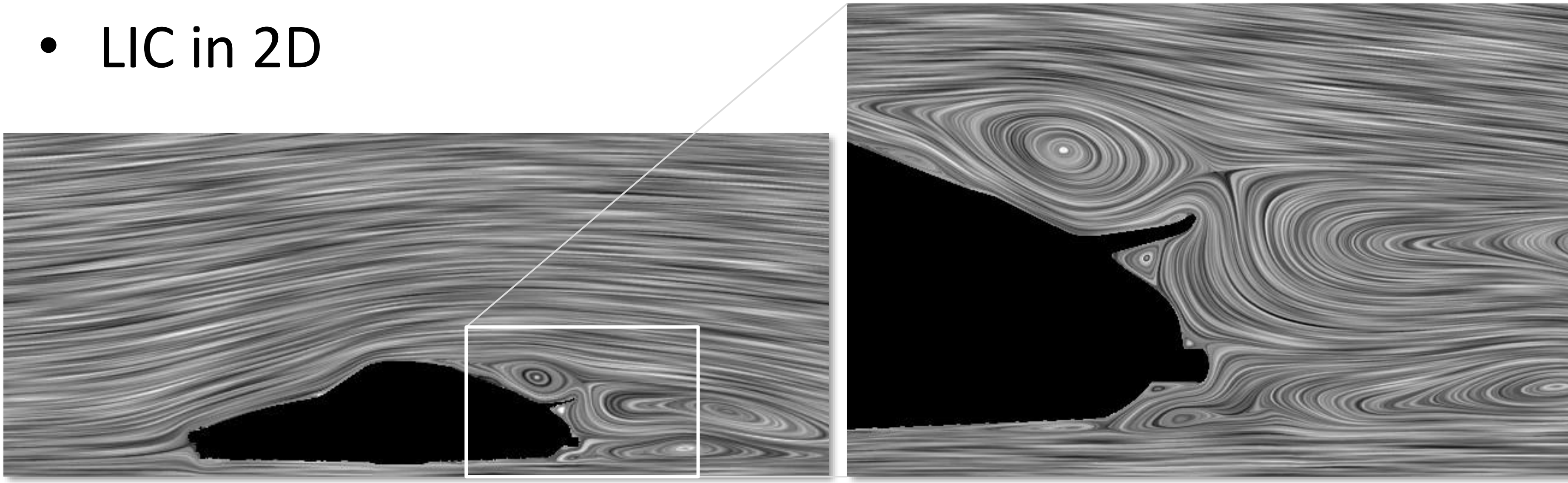


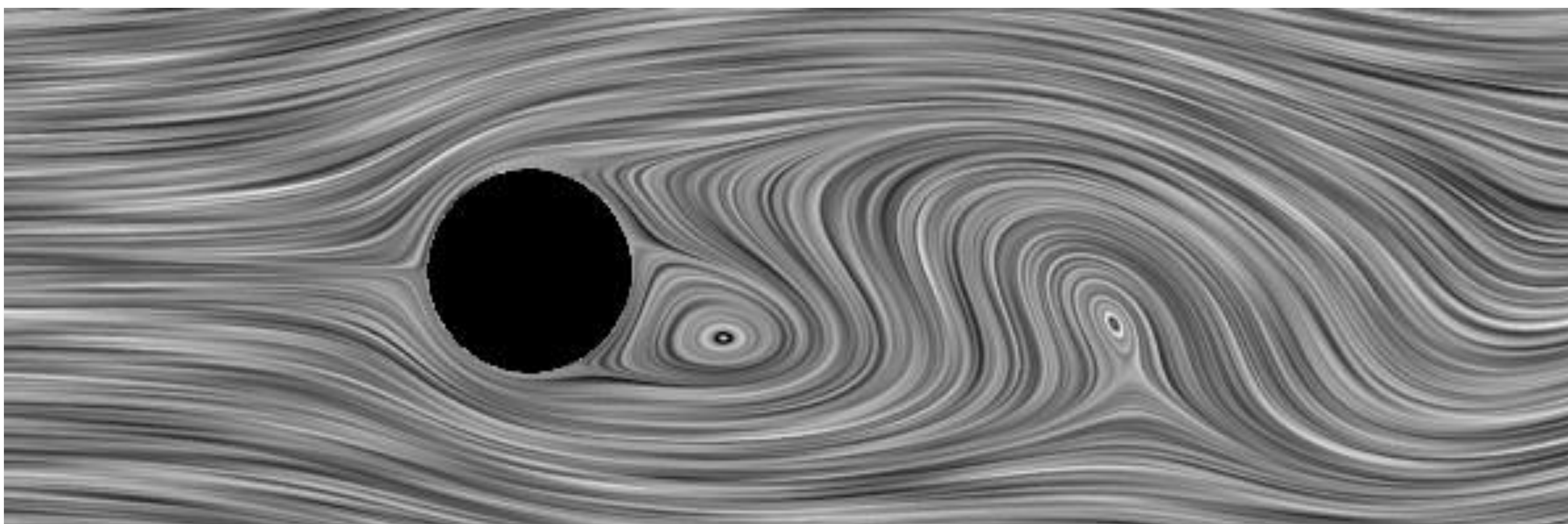
Texture values along trajectories are correlated (visually coherent)

[Cabral & Leedom 93]

Line Integral Convolution

- LIC in 2D

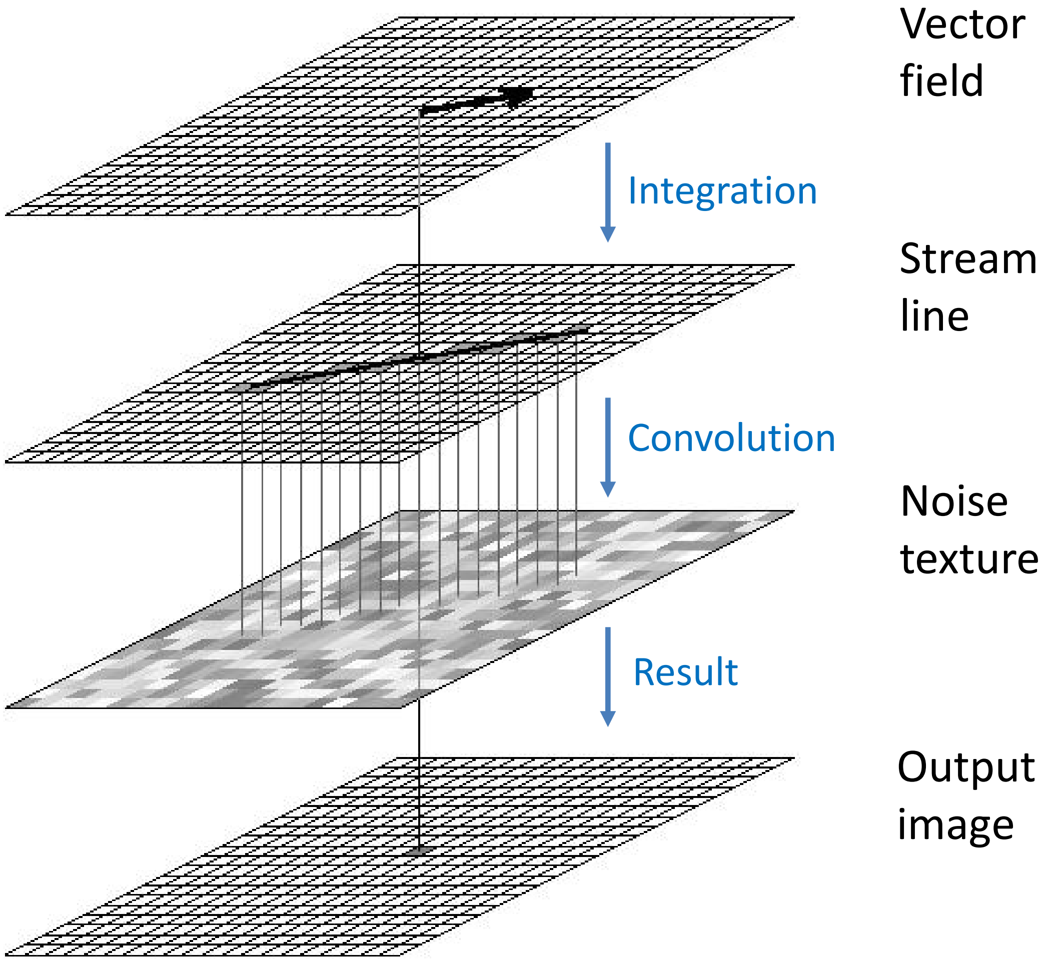




[Cabral & Leedom 93]

Line Integral Convolution

- Algorithm for 2D LIC
  - Look at stream line that passes through a pixel
  - Smear out - convolve - noise texture in direction of vector field (along stream line)



[Cabral & Leedom 93]

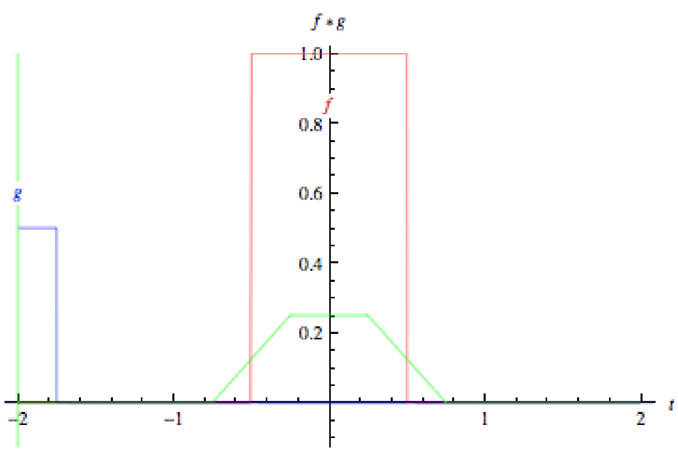
Filtering by convolution

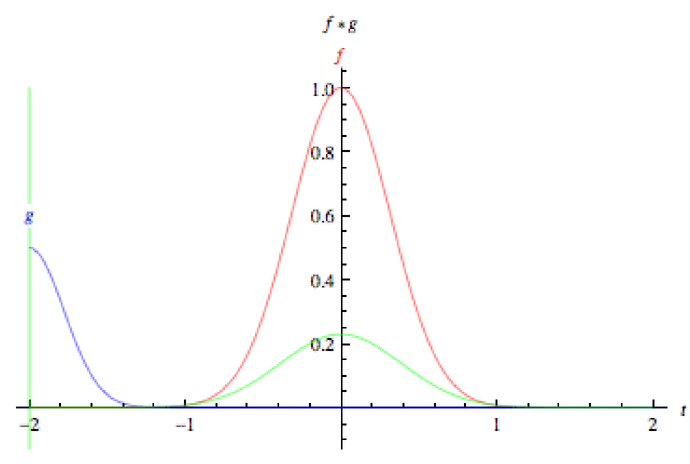
SIEMENS
Ingenuity for life

- Sliding a function $g(x)$ along a function $f(x)$

$$s(x') = [f * g](x') = \int_{-\infty}^{\infty} f(x)g(x' - x)dx$$

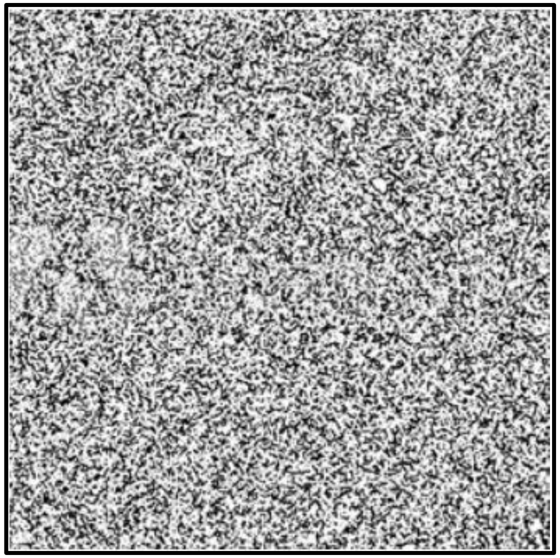
  – Function *f* is averaged with a weight function *g*
    - $(x' - x)$ centers *g* around x'



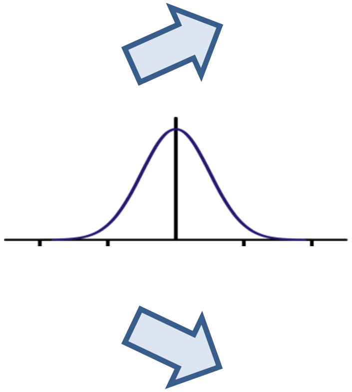


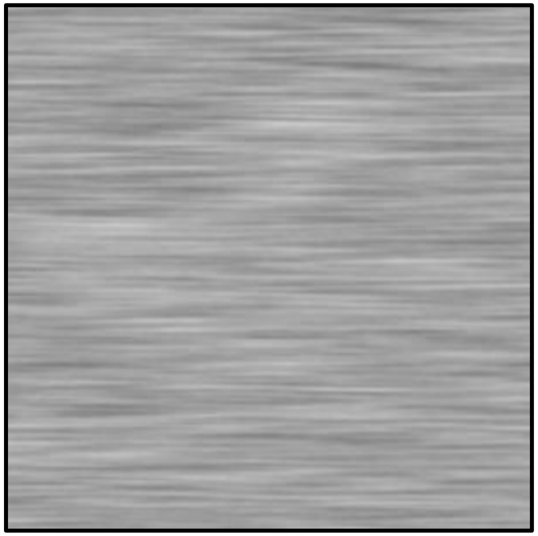
http://mathworld.wolfram.com/Convolution.html

Filtering by convolution

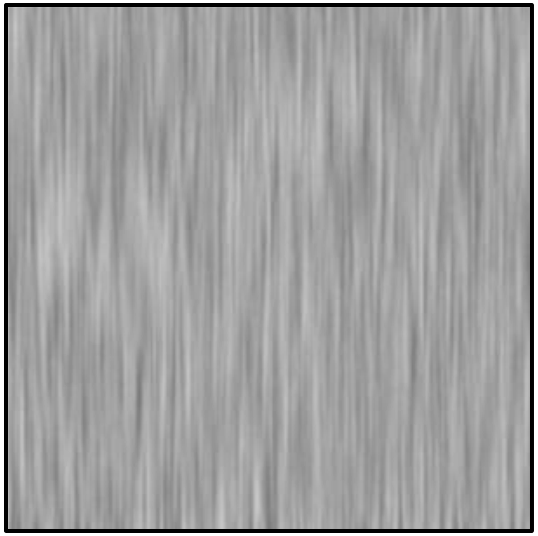


White noise (no correlation)





Horizontal Gaussian blur

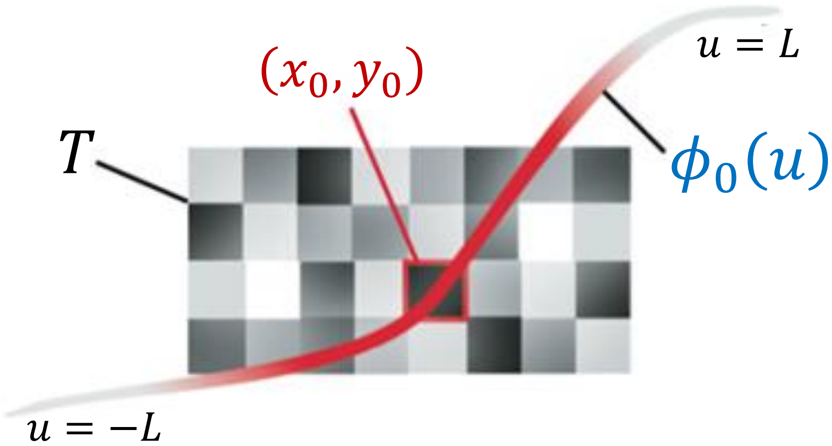


Vertical Gaussian blur

Line Integral Convolution

SIEMENS
Ingenuity for life

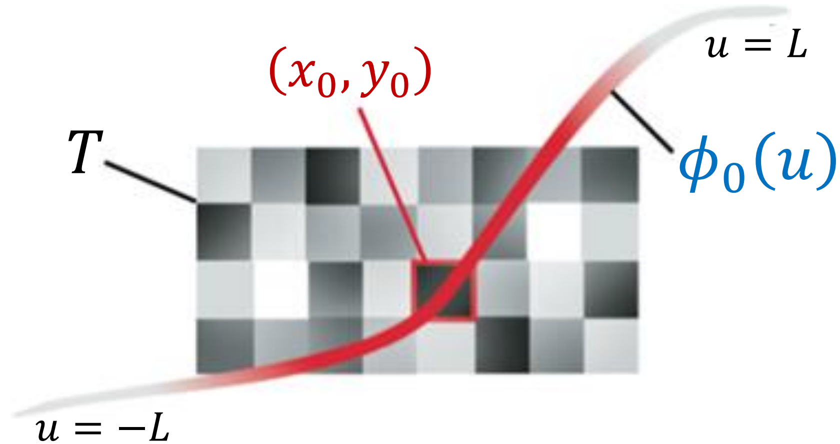
- Algorithm for 2D LIC
  - Let $\phi_0(u)$ be the stream line containing the point (x_0, y_0)
  - $T(x, y)$ is the randomly generated noise texture



[Cabral & Leedom 93]

Line Integral Convolution

- Algorithm for 2D LIC
  - Let $\phi_0(u)$ be the stream line containing the point (x_0, y_0)
  - $T(x, y)$ is the randomly generated noise texture



  - Compute the intensity at (x_0, y_0) as

$$I(x_0, y_0) = \int_{-L}^{L} k(u) \cdot T\big(\phi_0(u)\big)\, du$$

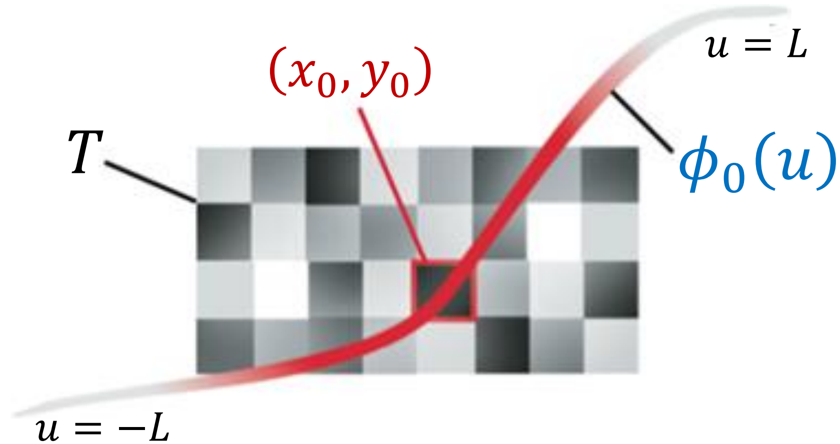
convolution with a kernel $k(u)$

[Cabral & Leedom 93]

Line Integral Convolution

SIEMENS
Ingenuity for life

- Algorithm for 2D LIC
  - Let $\phi_0(u)$ be the stream line containing the point (x_0, y_0)
  - $T(x, y)$ is the randomly generated noise texture

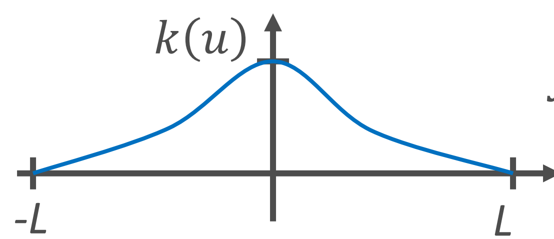


  - Compute the intensity at (x_0, y_0) as

$$I(x_0, y_0) = \int_{-L}^{L} k(u) \cdot T(\phi_0(u))\, du$$

convolution with a kernel $k(u)$

Smoothing filter kernel
  - Finite support [-*L*, *L*]
  - Normalized, usually symmetric
  - E.g., Gaussian or box filter

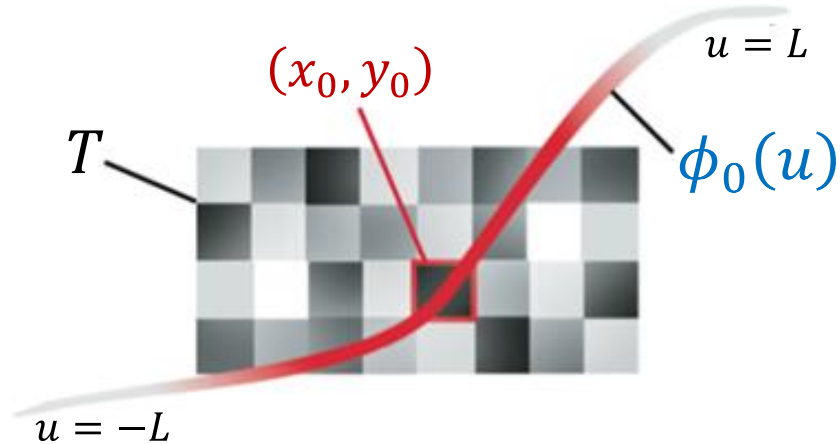


$$\int_{-L}^{L} k(u)\, du = 1$$

[Cabral & Leedom 93]

Line Integral Convolution

- Algorithm for 2D LIC
  - Let $\phi_0(u)$ be the stream line containing the point (x_0, y_0)
  - $T(x, y)$ is the randomly generated noise texture

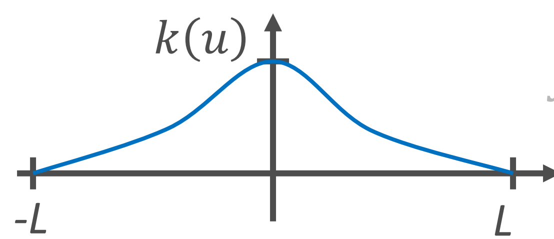


  - Compute the intensity at (x_0, y_0) as

$$I(x_0, y_0) = \int_{-L}^{L} k(u) \cdot T\big(\phi_0(u)\big)\, du$$

convolution with a kernel $k(u)$

- Smoothing filter kernel
  - Finite support [-L, L]
  - Normalized, usually symmetric
  - E.g., Gaussian or box filter



$$\int_{-L}^{L} k(u)\, du = 1$$

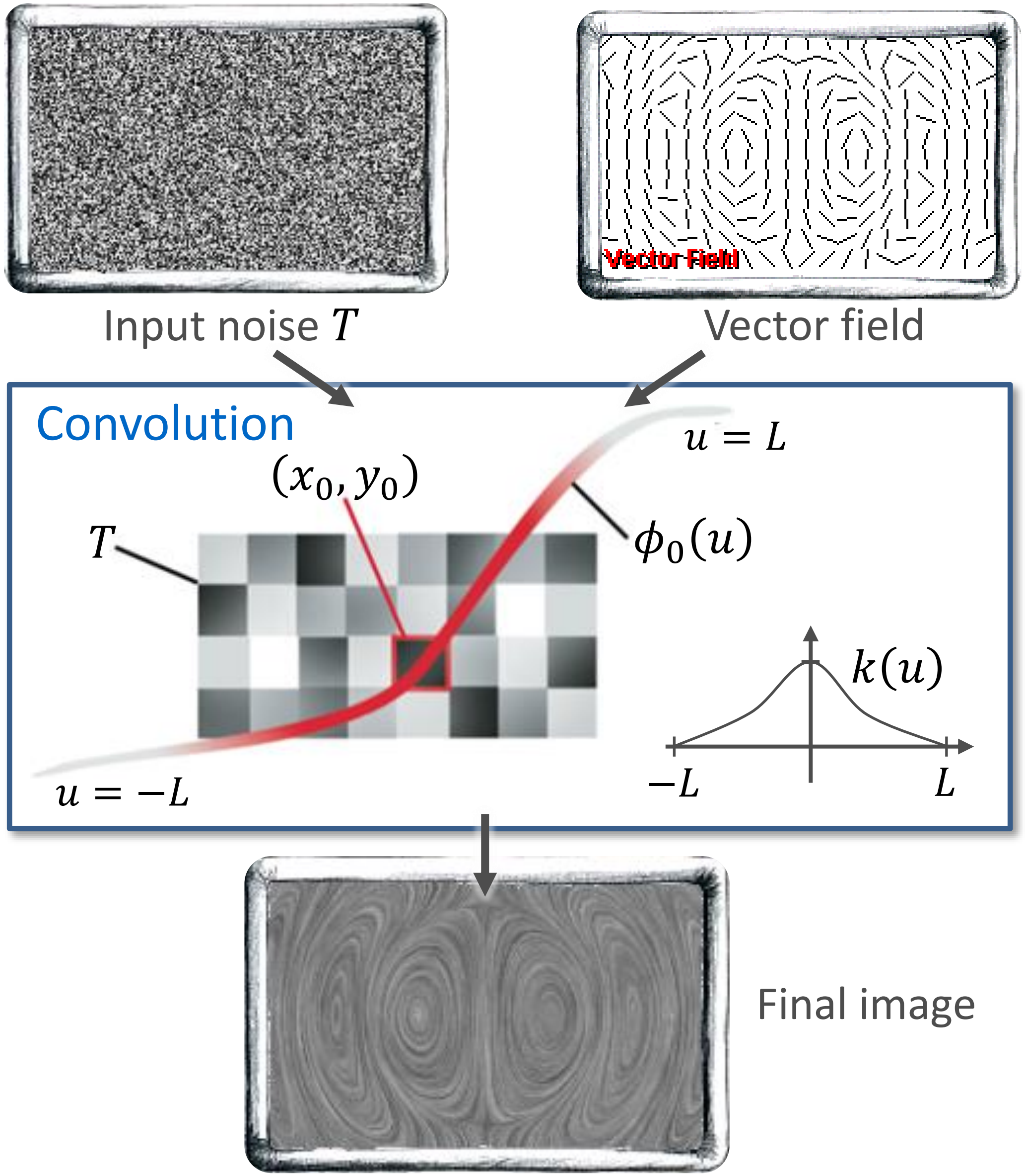
[Cabral & Leedom 93]

Line Integral Convolution

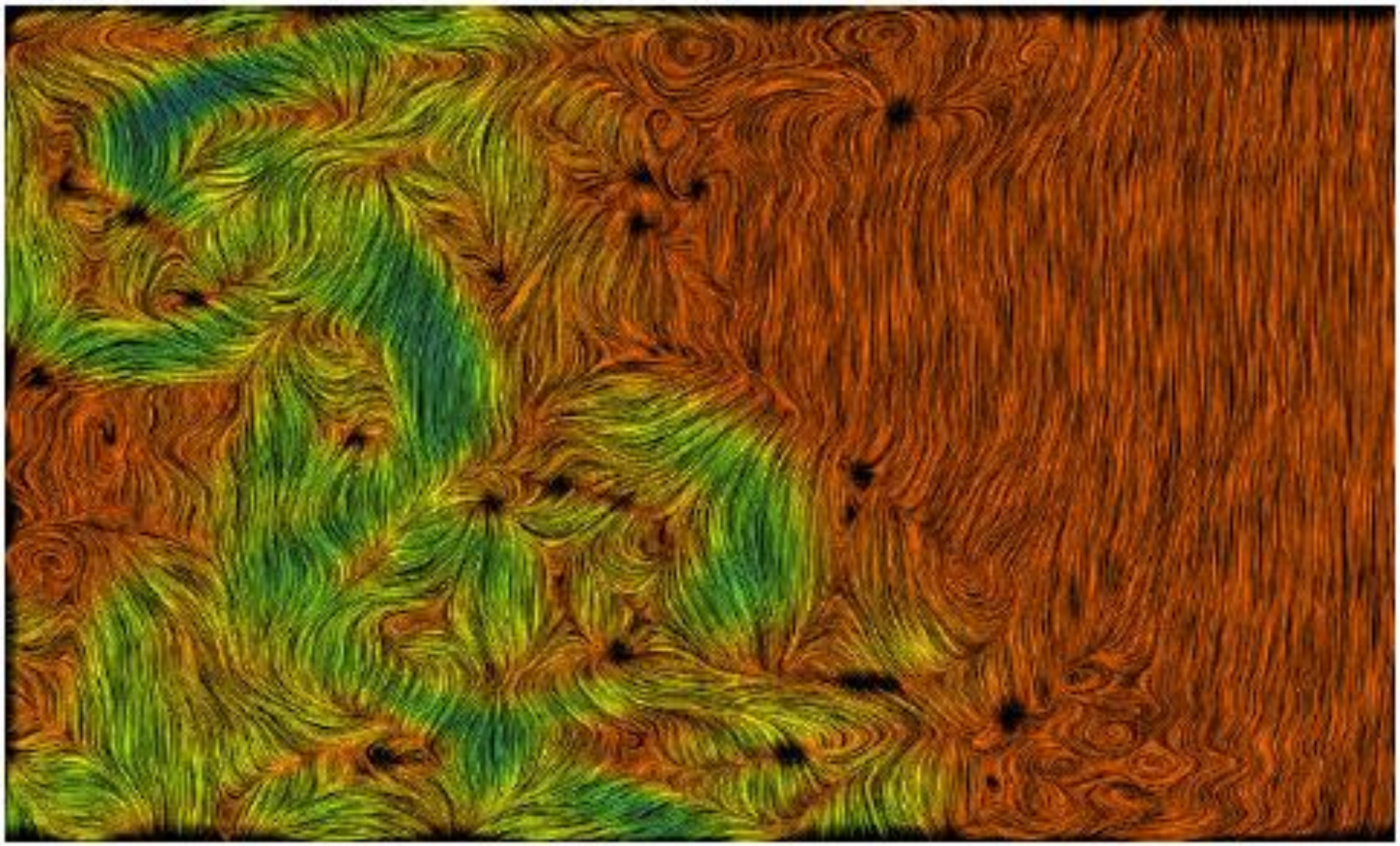
$$I(x_0, y_0) = \int_{-L}^{L} k(u) \cdot T\big(\phi_0(u)\big)\, du$$

- LIC is a convolution of
  - a noise texture $T(x, y)$
  - and a smoothing filter $k(u)$

- Noise texture values are picked up along the stream line $\phi_0(u)$ through $T\big(\phi_0(u)\big)$



Line Integral Convolution

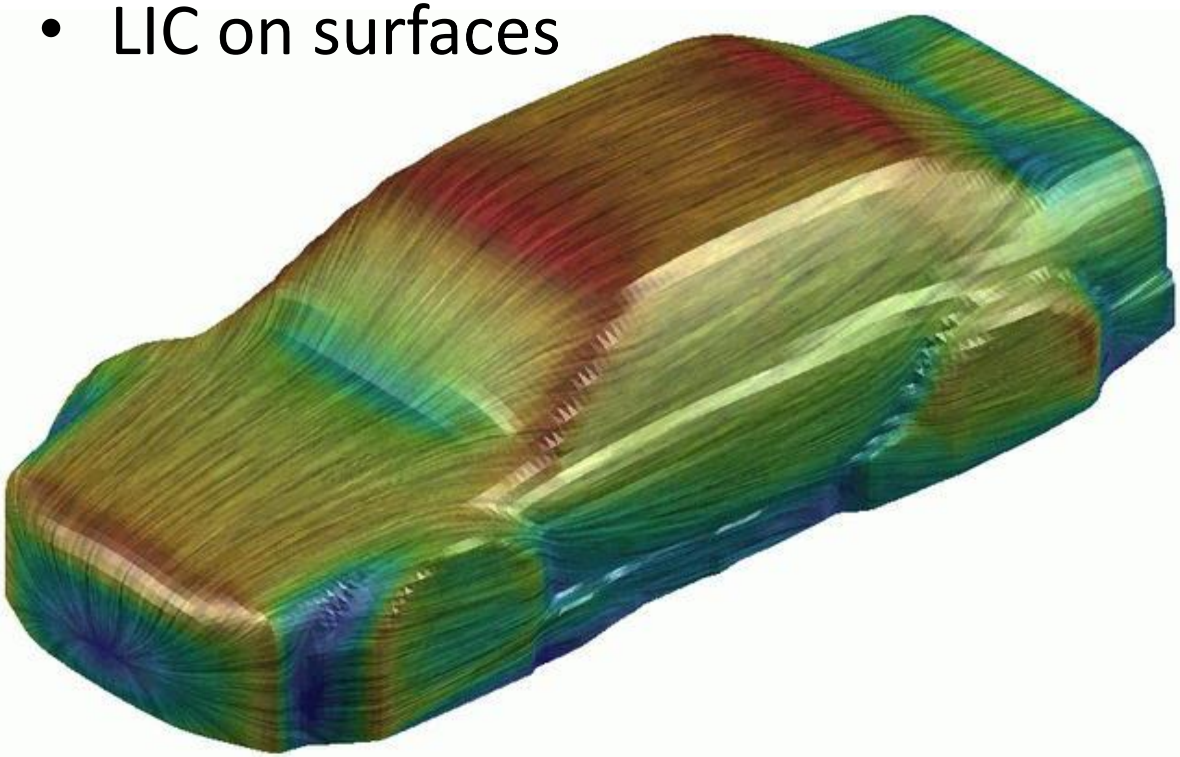


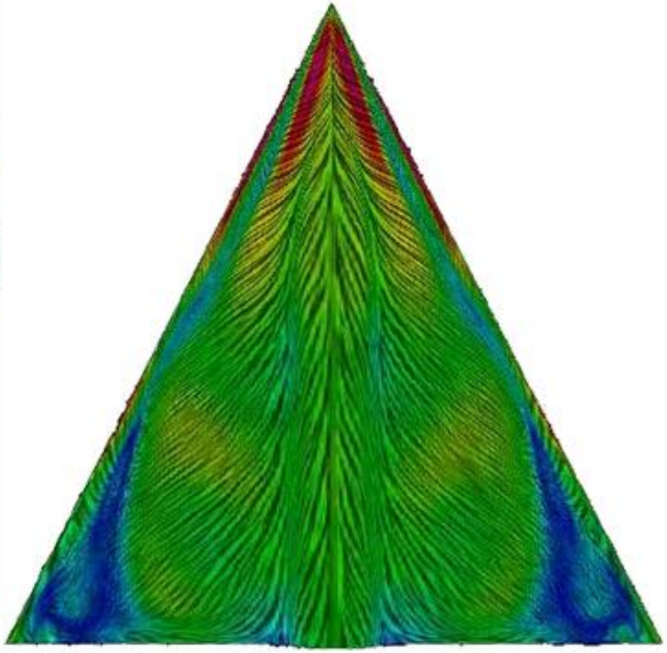
LIC and color coding of velocity magnitude

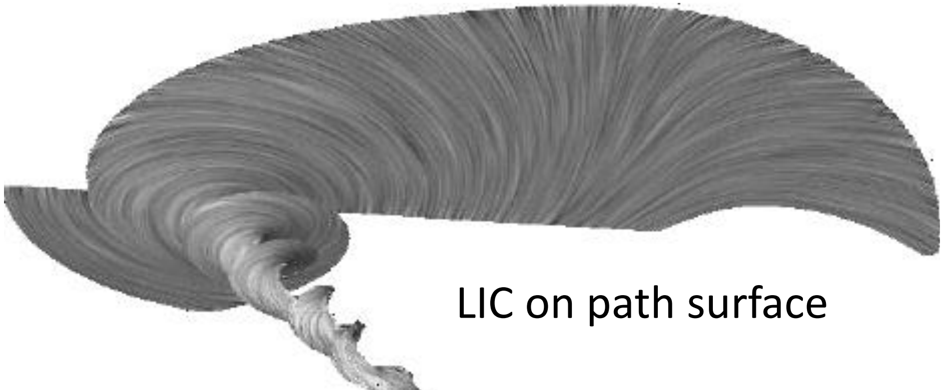
[Cabral & Leedom 93]

Line Integral Convolution

- LIC on surfaces



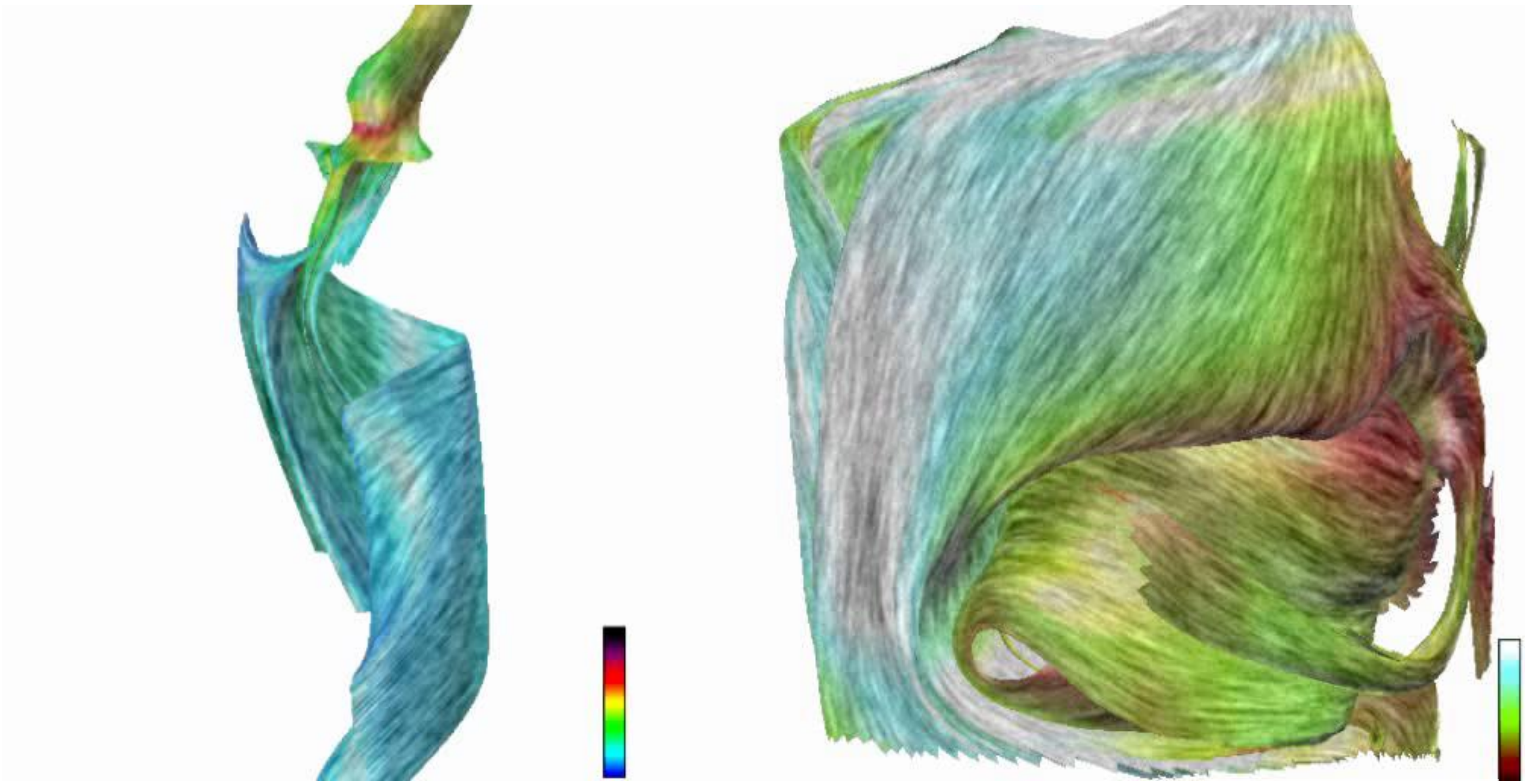




LIC on path surface

Data sources – artificial flows

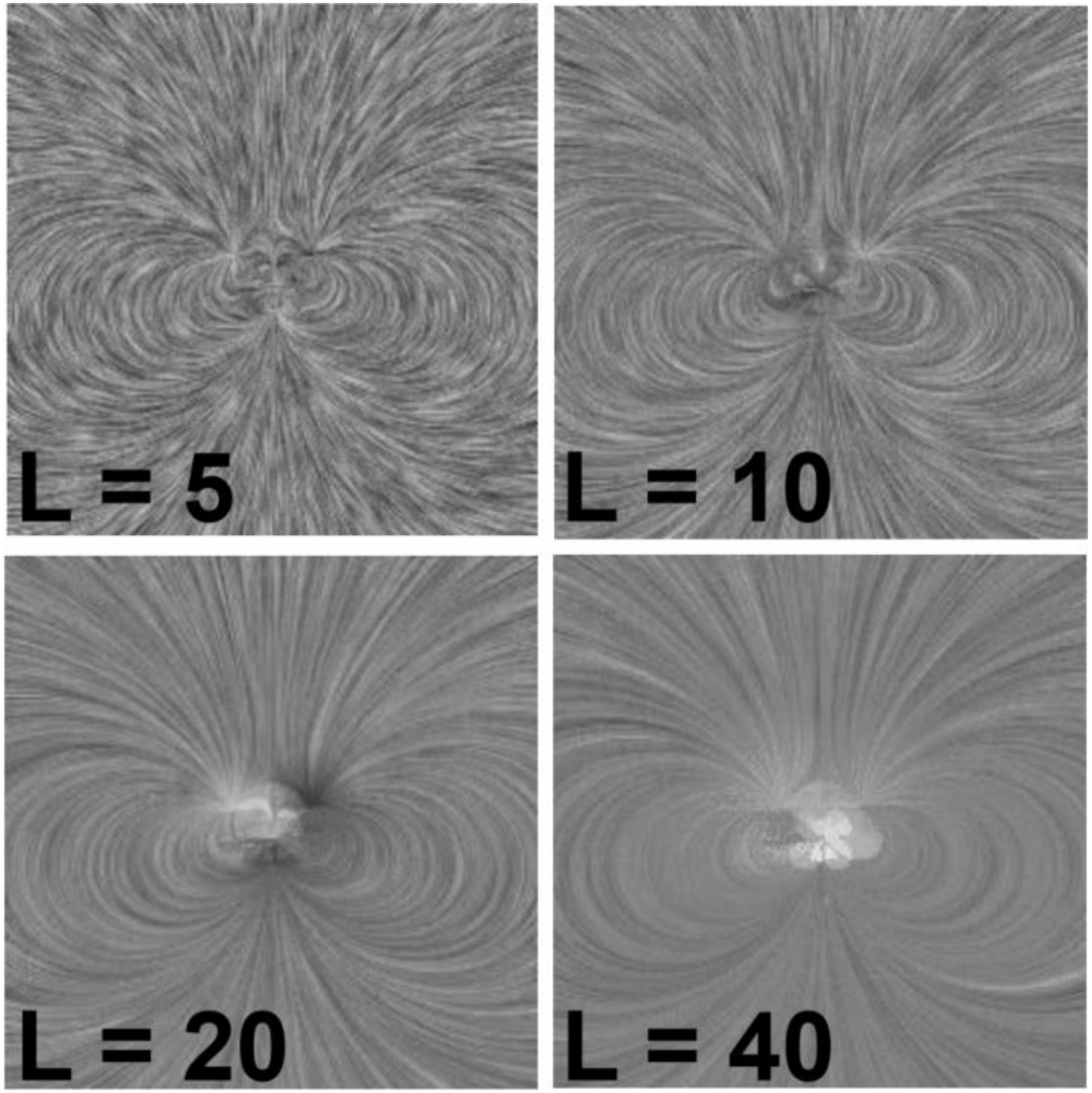
- Visualization of surface flows



[Laramee 2006]

Line Integral Convolution

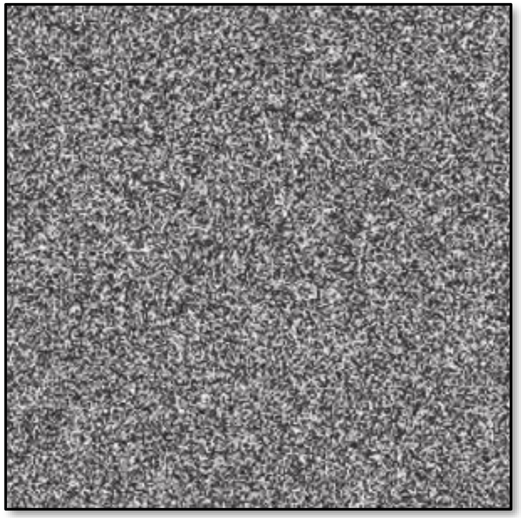
- Influence of filter length



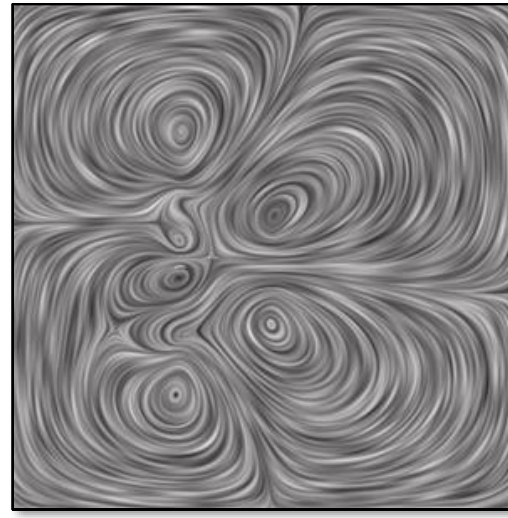
[Schumann & Müller 00]

Line Integral Convolution

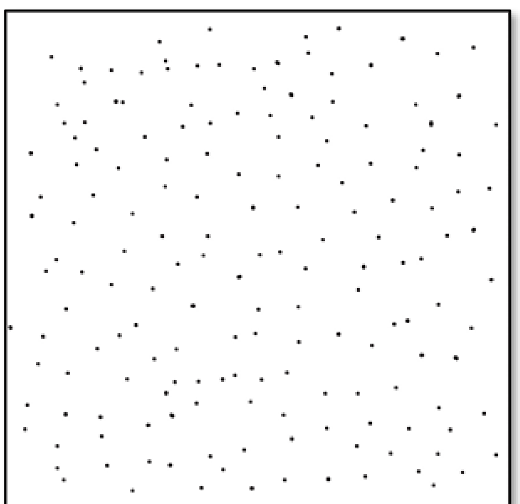
- Influence of input noise



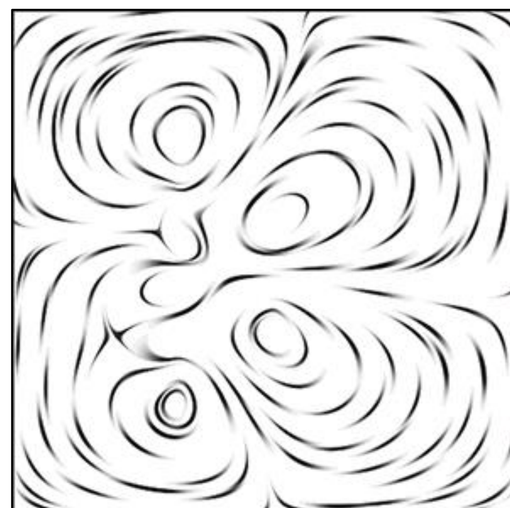
White noise texture



LIC result for white noise



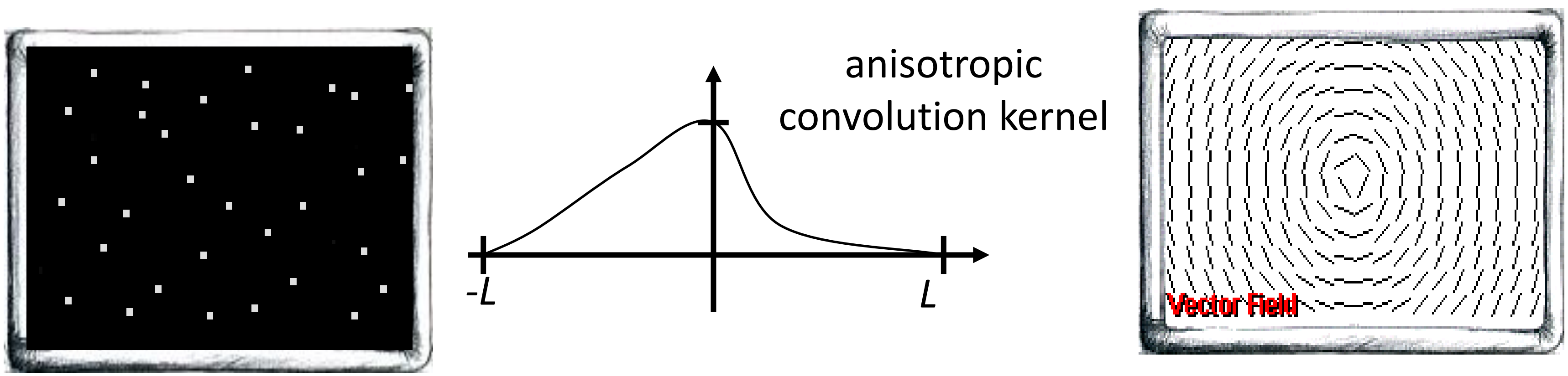
Sparse noise texture



LIC result for sparse noise

Line Integral Convolution

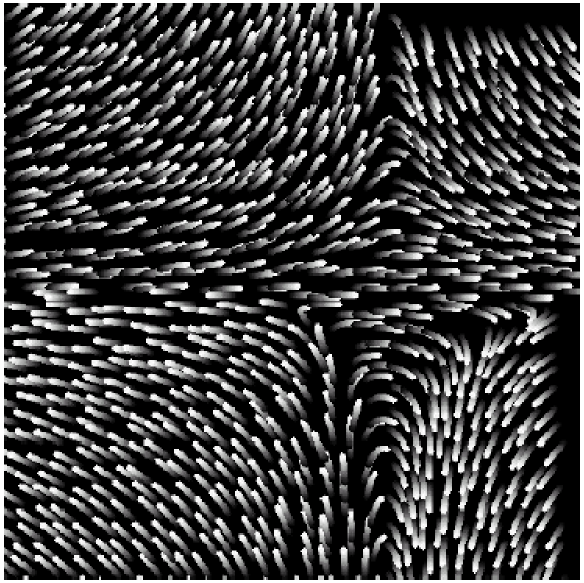
- Oriented LIC (OLIC)
  - Visualizes orientation (in addition to direction)
  - Uses a sparse texture; i.e. smearing of individual drops
  - Asymmetric convolution kernel



[Wegenkittel & Gröller 97]

Line Integral Convolution

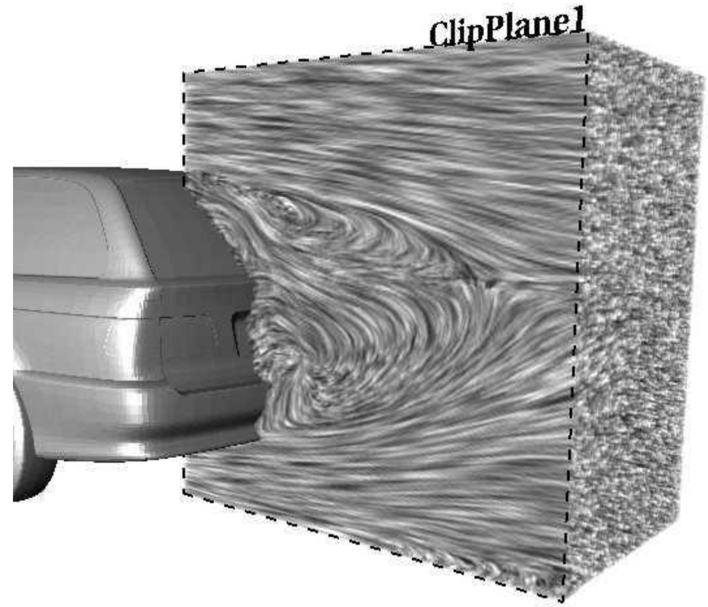
- Oriented LIC (OLIC)

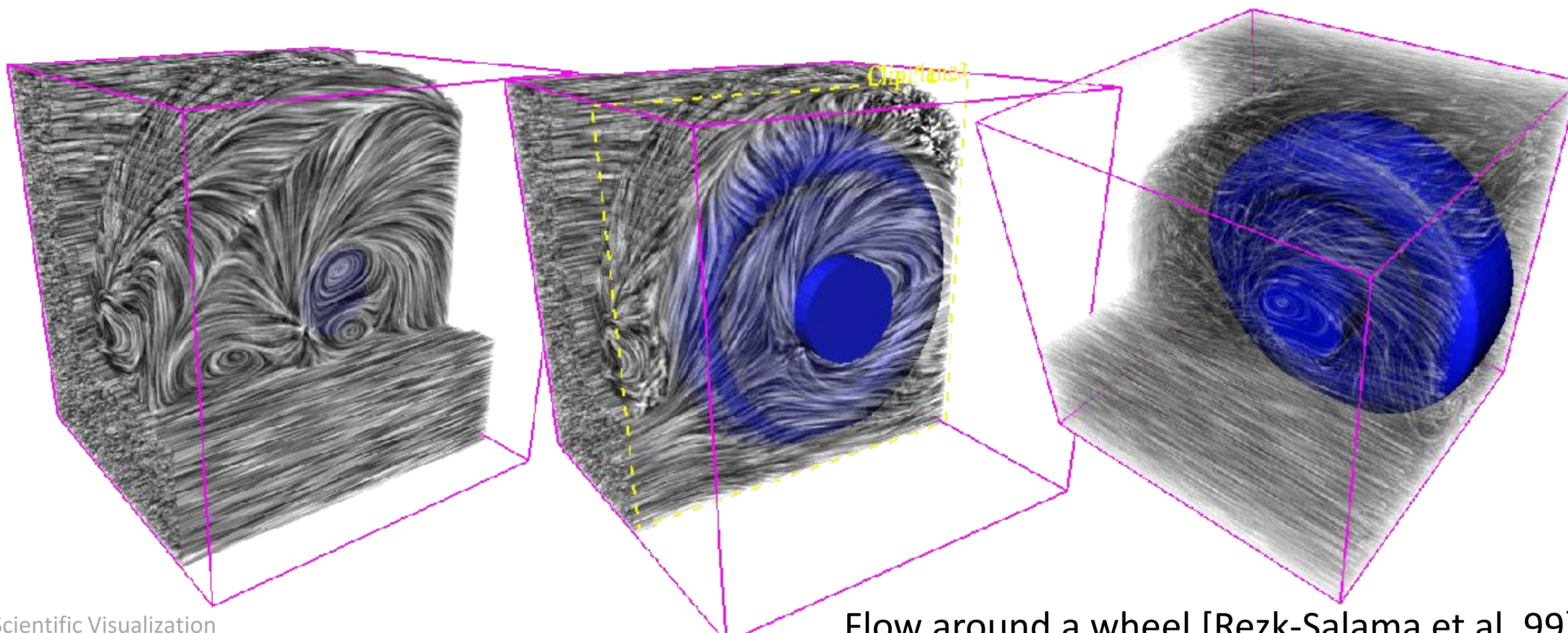


[Wegenkittel & Gröller 97]

Line Integral Convolution

- 3D LIC
  - Only good if non-relevant data is discarded

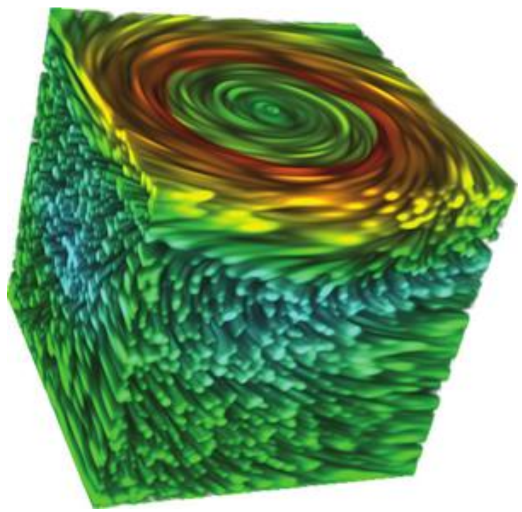




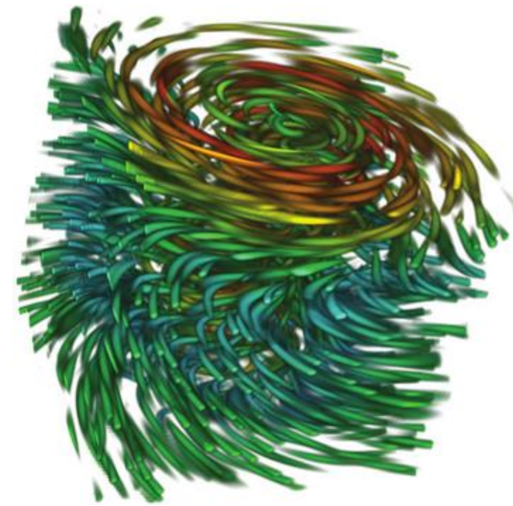
Flow around a wheel [Rezk-Salama et al. 99]

Line Integral Convolution

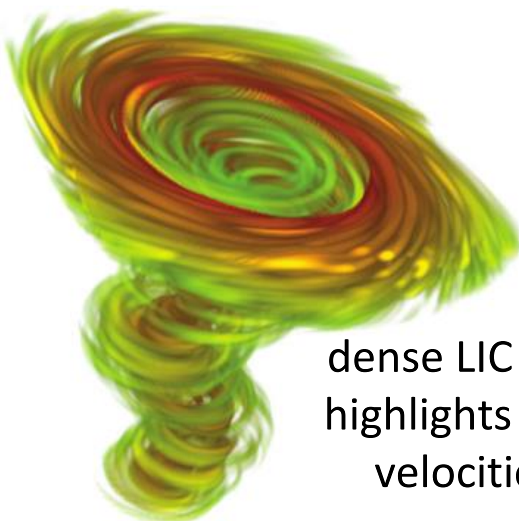
- 3D LIC



dense LIC with
white noise input



sparse LIC with
spares noise



dense LIC that
highlights high
velocities

sparse LIC that
highlights high
velocities

Line Integral Convolution

SIEMENS
Ingenuity for life

- Summary
  - Dense representation of flow fields
  - Convolution along characteristic lines
    → correlation along these lines
  - For 2D and (3D flows)

References

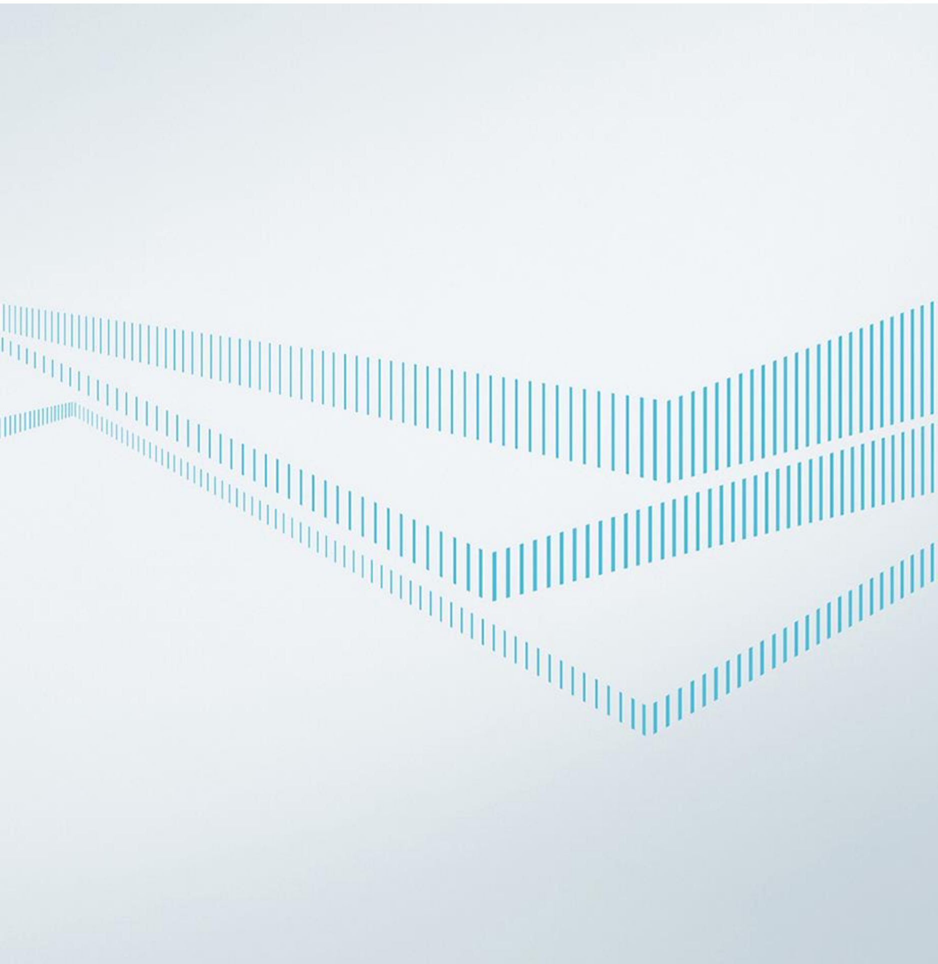
- Jobard & Lefer 97: Creating evenly-spaced streamlines of arbitrary density, In *Visualization in Scientific Computing*, 1997.
- Spencer et al. 09: Evenly-spaced streamlines for surfaces: An image-based approach, *Computer Graphics Forum*, 28(6), 2009.
- Cabral & Leedom 93: Imaging vector fields using line integral convolution. In *Proc. ACM SIGGRAPH*, 1993.
- Wegenkittel & M. Gröller 97: Fast oriented line integral convolution for vector field visualization via the Internet. In *Proc. IEEE Visualization*, 1997.

Acknowledgements

SIEMENS
Ingenuity for life

- Helwig Hauser
- Andrea Brambilla
- Daniel Weiskopf
- Ronald Peikert
- Christoph Garth
- Alexandru C. Telea
- Many more

Contact information



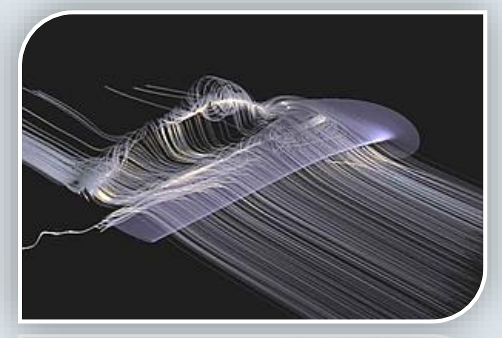
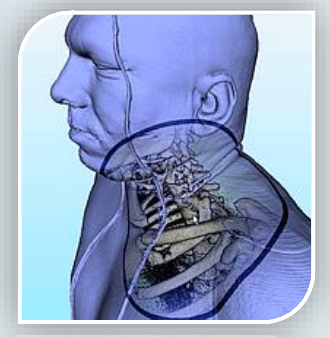
Dr. Johannes Kehrer

Siemens Technology
T RDA BAM IBI-DE
Otto-Hahn-Ring 6
81739 München, Deutschland

E-mail:
kehrer.johannes@siemens.com

Internet
siemens.com/innovation

Intranet
intranet.ct.siemens.com

Visual Data Analysis

Visualization Mapping I

Dr. Johannes Kehrer

Disclaimer

These lecture slides and materials are for personal use by students and are intended for educational purposes only. Distribution or reproduction of the material is prohibited.

The views and opinions expressed by any individual during the course of these lecturers are their own and do not necessarily represent the views, opinions, or positions of Siemens or the Technical University of Munich.

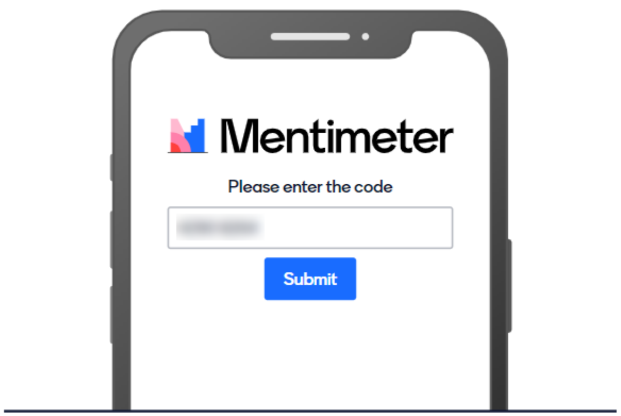
Although reasonable efforts have been made to ensure the information in these lecture slides is current and accurate, we do not assume any responsibility for the content of the posted material.

These slides also contain links to websites of third parties. As the content of these websites is not under our control, we cannot assume any liability for such external content.

Any comments/questions?

Go to

www.menti.com



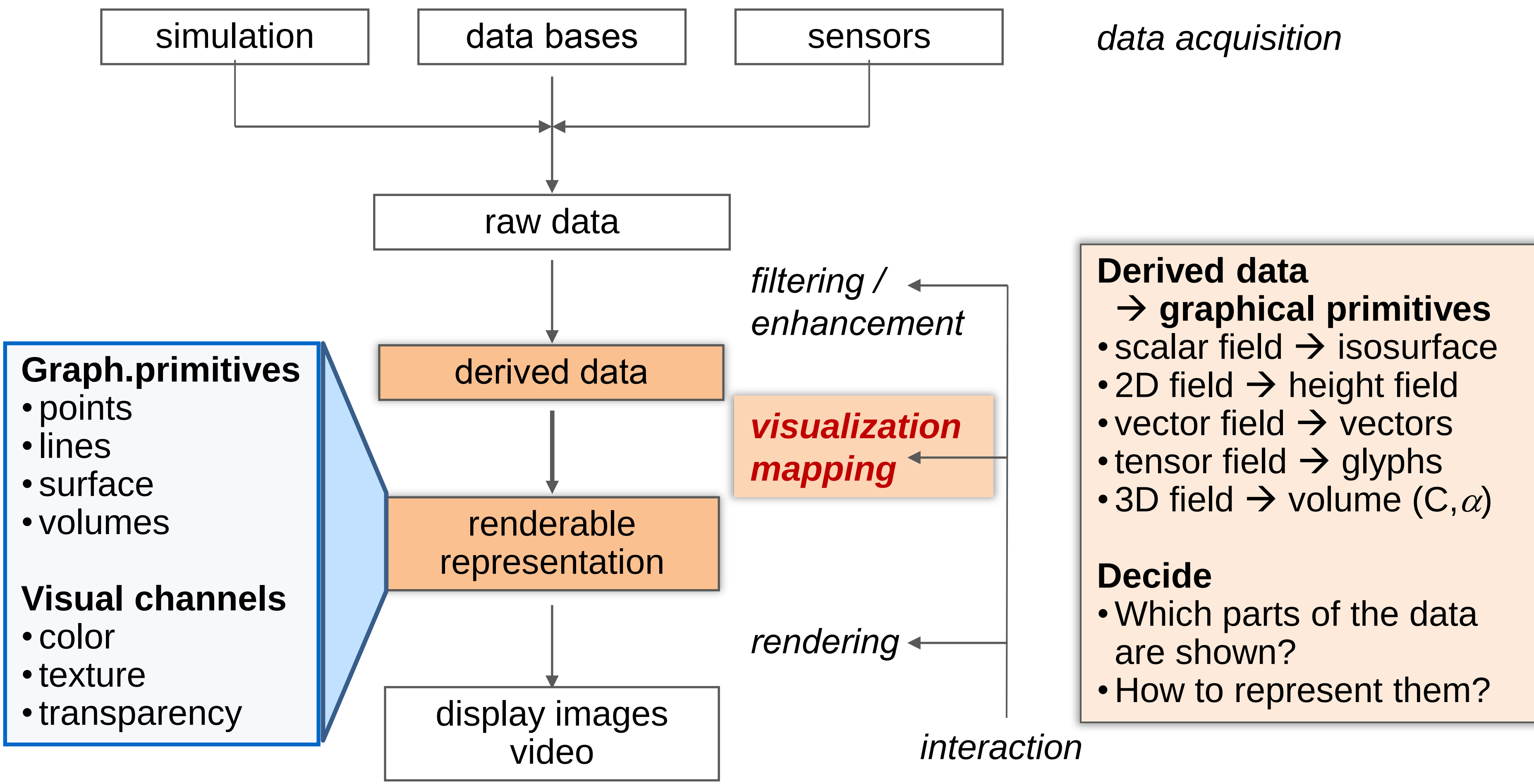
Enter the code

3850 4109

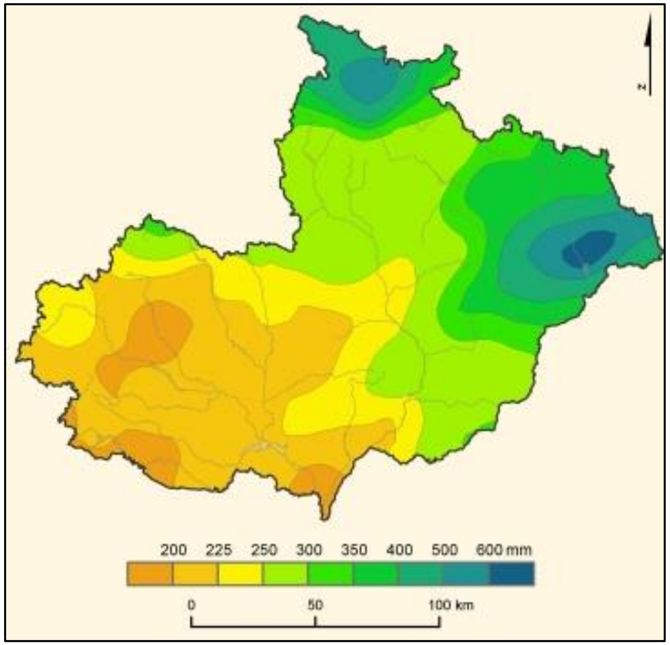
Or use QR code

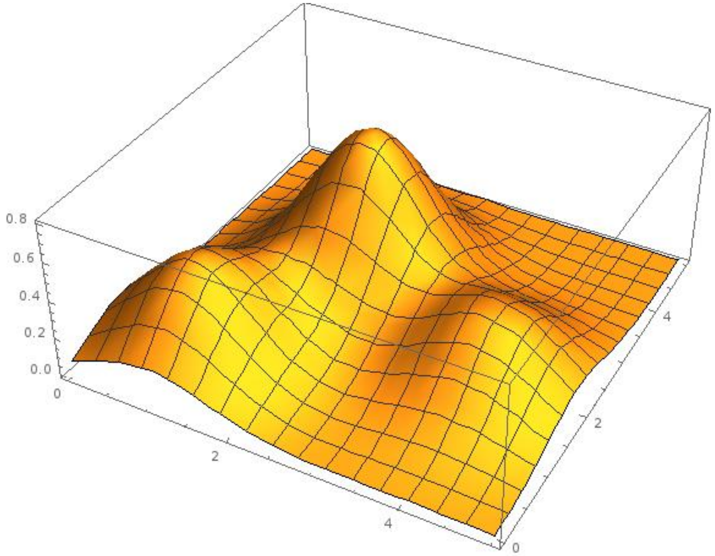
Mapping techniques

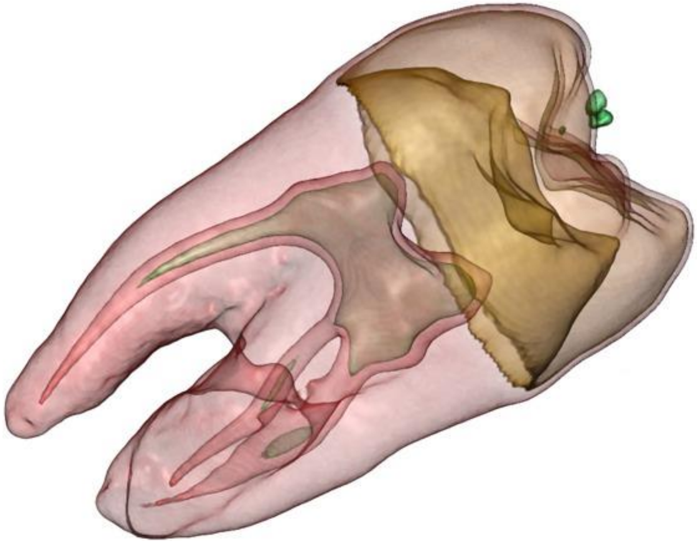
- From derived data to a renderable representation

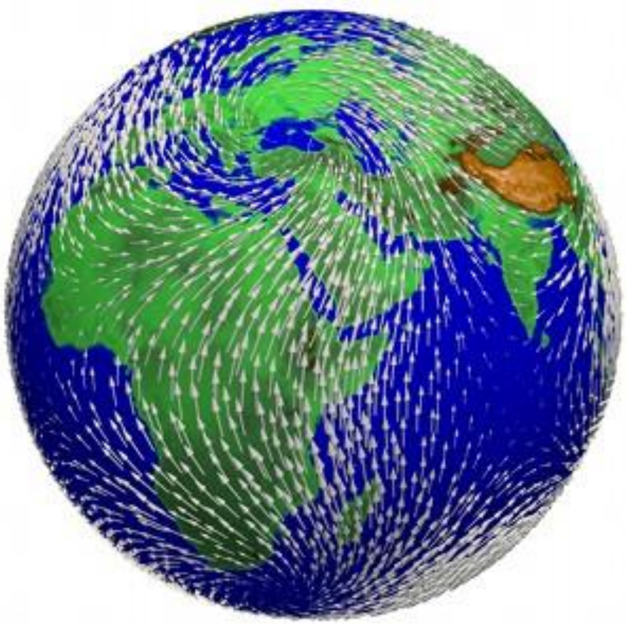


Visualization Mappings – Examples



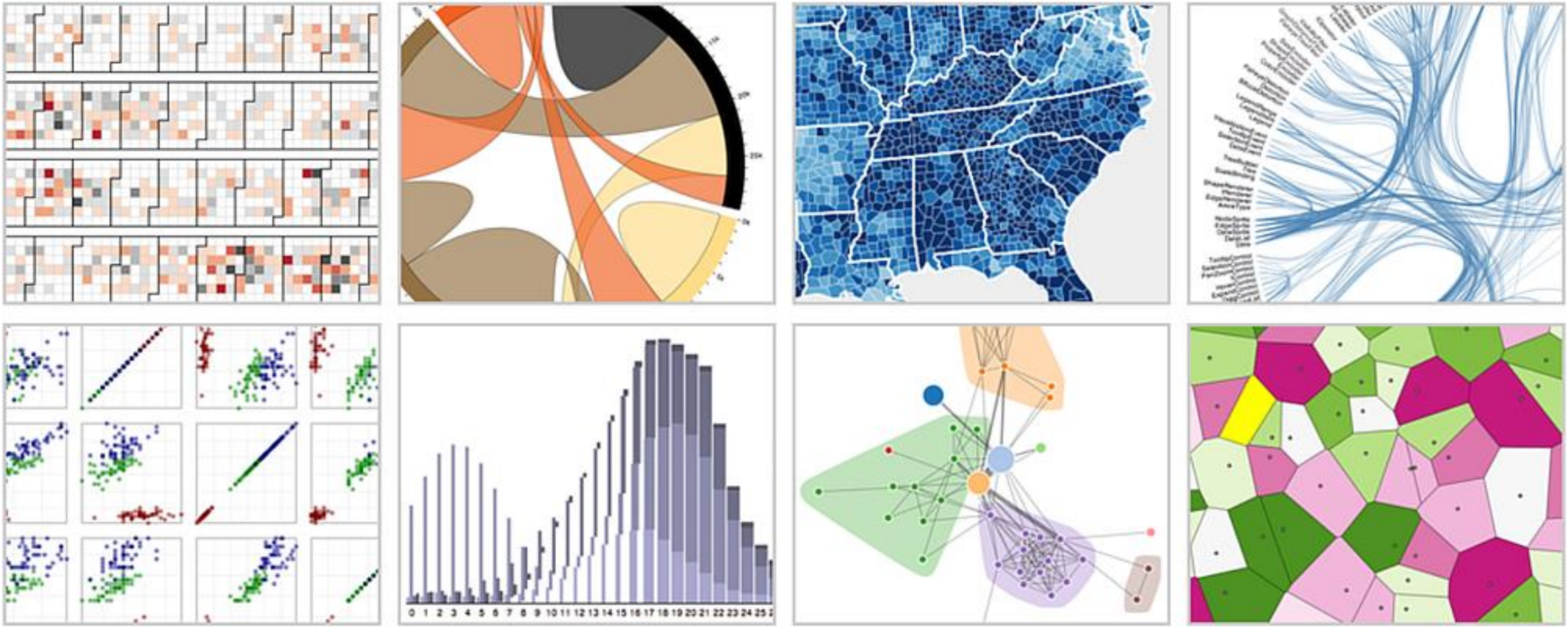






Visualization Mappings – Examples

SIEMENS
Ingenuity for life

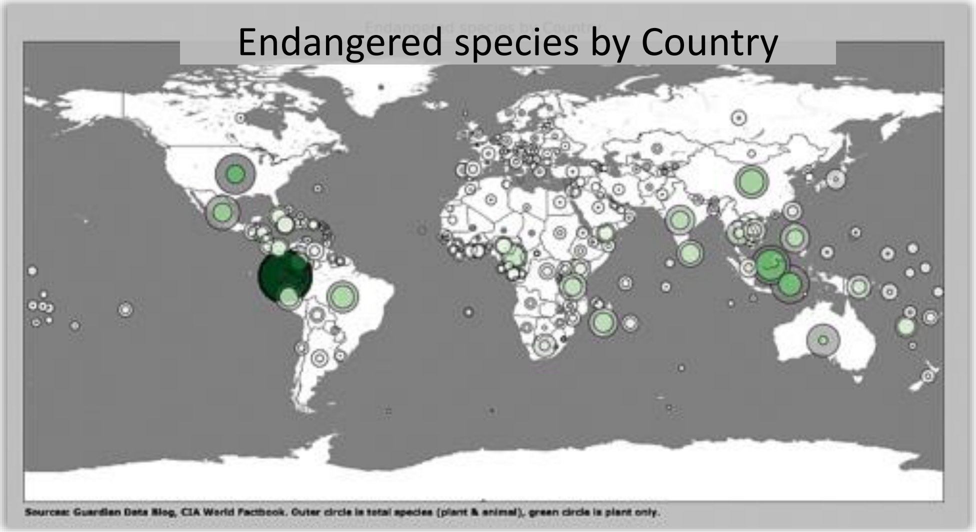


Interactive visualizations built with D3. From left to right: calendar view, chord diagram, choropleth map, hierarchical edge bundling, scatterplot matrix, grouped & stacked bars, force-directed graph clusters, Voronoi tessellation

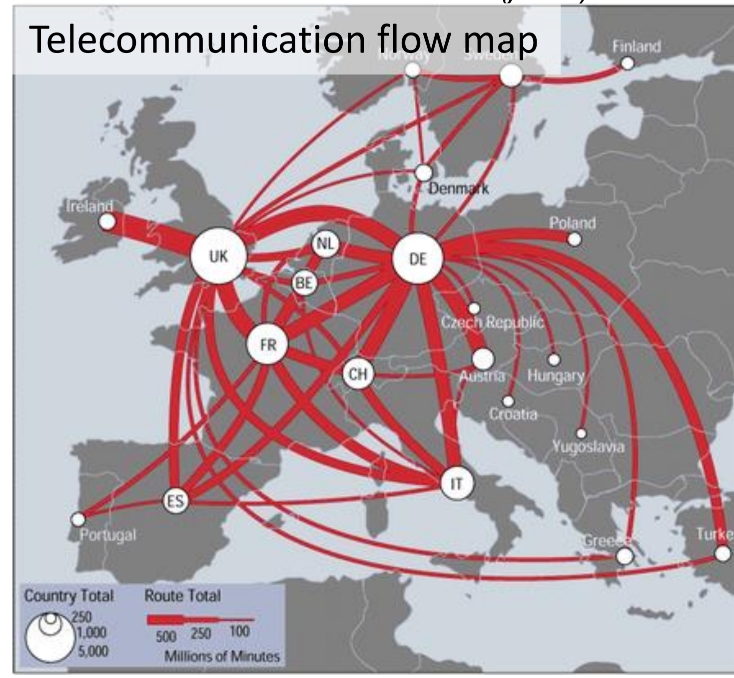
https://d3js.org/

Mapping techniques

- Mapping of data to a visual (renderable) representation



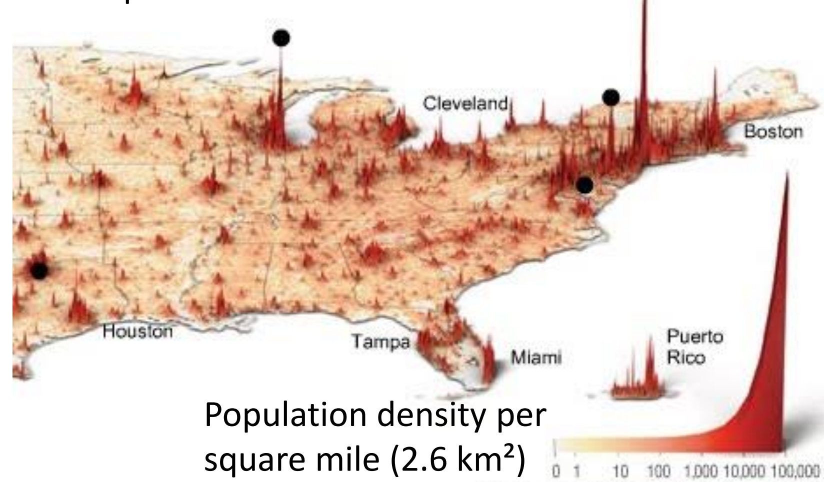
Endangered species by Country



Telecommunication flow map

Military expenses worldwide

80% of US population lives in a metropolitan area.

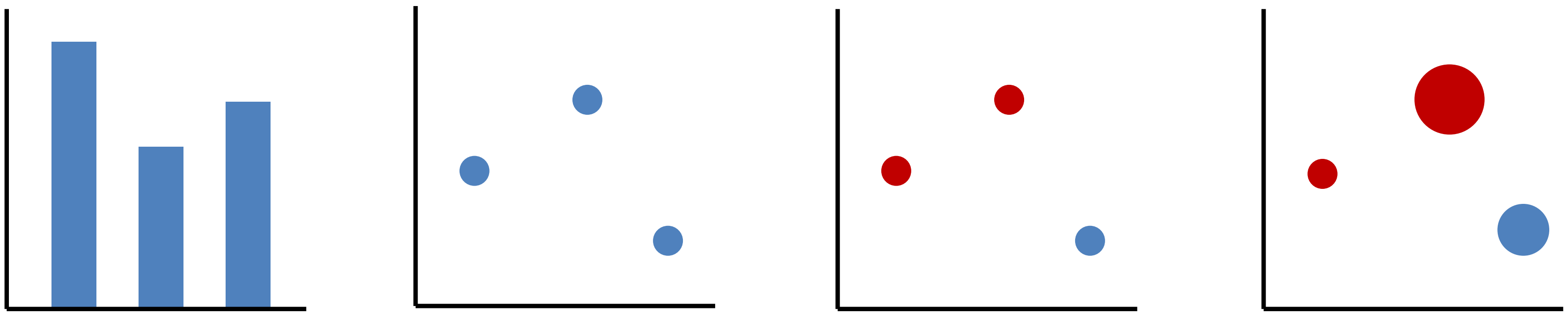


Population density per square mile (2.6 km²)

Mapping techniques

SIEMENS
Ingenuity for life

How to systematically analyze visual mappings?



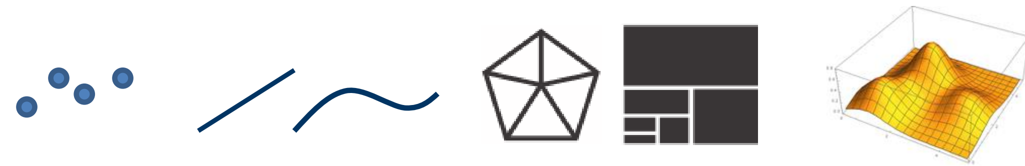
Mapping of data to a visual representation consists of:

– Graphical primitives: represent data items or links

– Visual channels: control appearance of graph. primitives based on data attributes

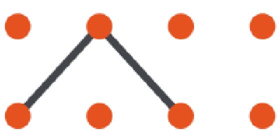
Mapping techniques

- Mapping of data to a visual representation consists of:
  - Graphical primitives
    that represent data items

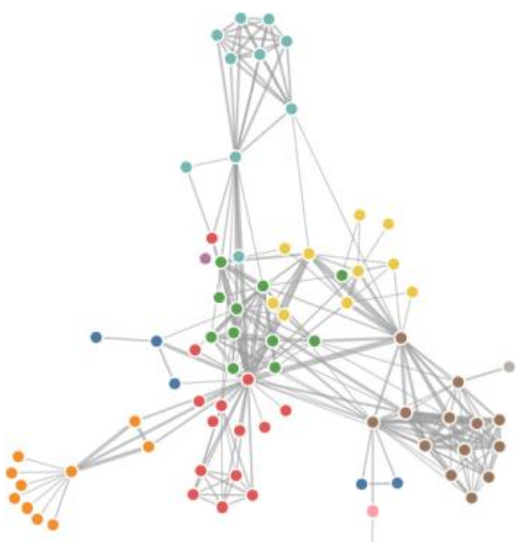


points lines areas surfaces

Representation of links between data items



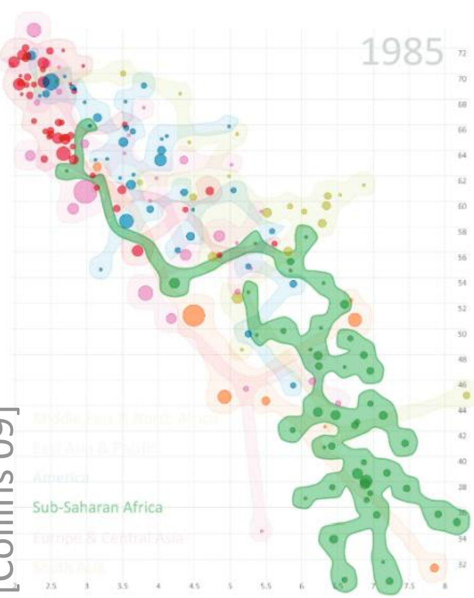
Connection



observablehq.com/@d3/force-directed-graph

Containment

[Collins 09]

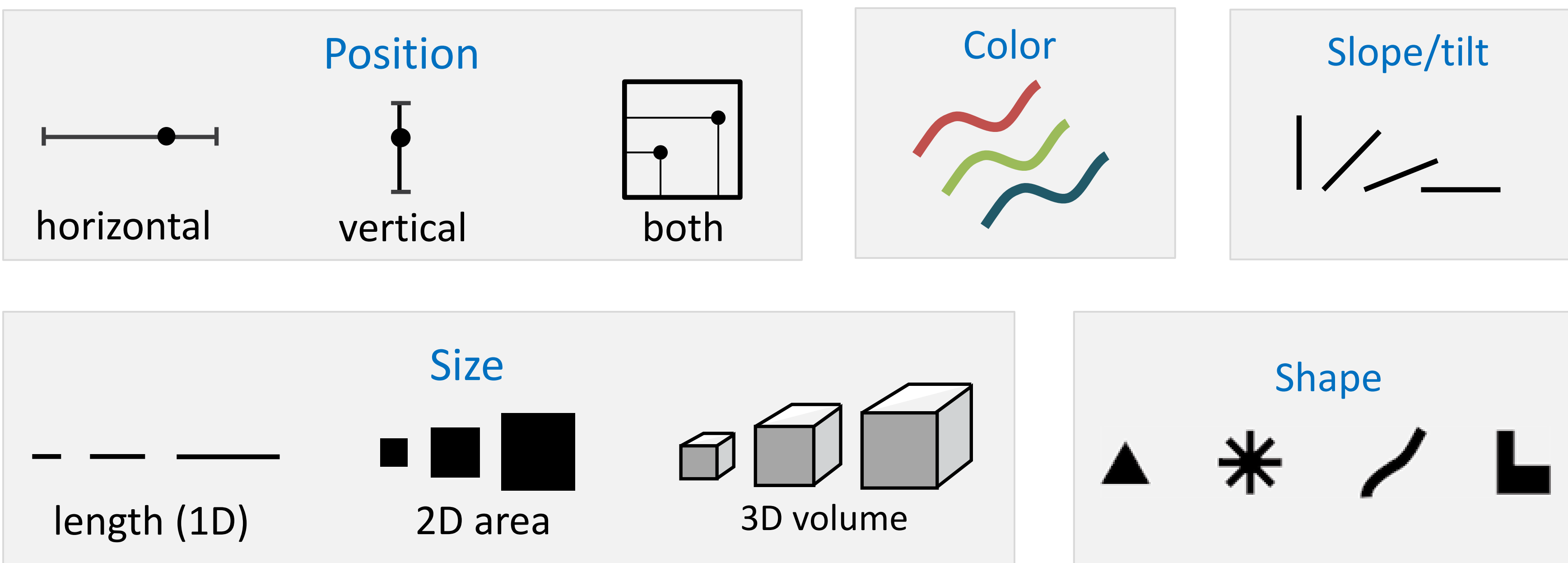


[Munzner 14]

Mapping techniques

SIEMENS
Ingenuity for life

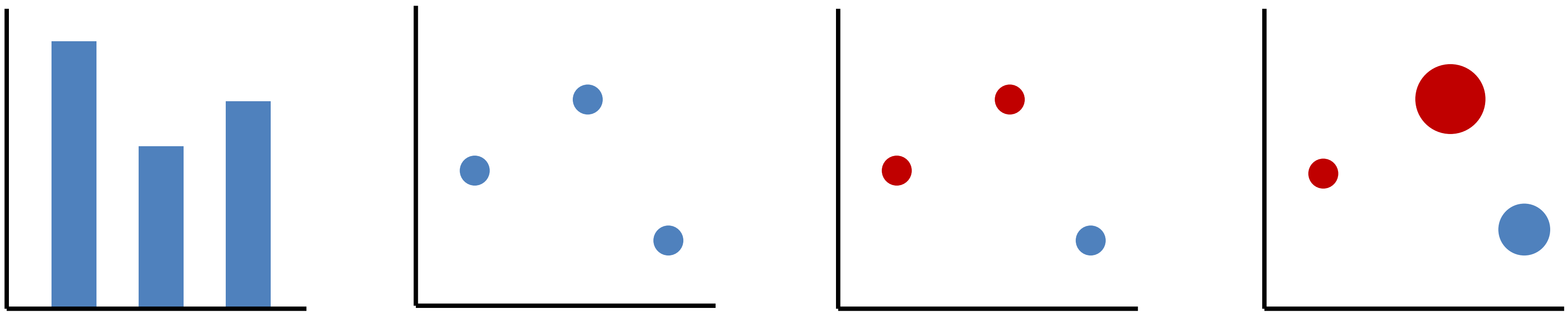
- Mapping of data to a visual representation consists of:
  - Visual channels that control appearance of graph. primitives based on data attributes



[Munzner 14]

Mapping techniques

Combination of graphical primitives and visual channels



Visual channel

1: length/position	2: vertical position horizontal position	3: vertical position horizontal position color hue	4: vertical position horizontal position color hue size (area)

Graphical primitive

line	point	point	point

Mapping techniques

SIEMENS
Ingenuity for life

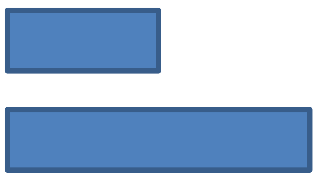
- Effectiveness principle
  - Some visual channels are better than others
  - Encode most important data attributes with most effective/accurate channels

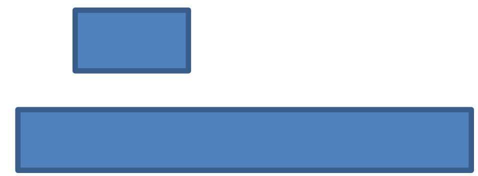
- Properties of visual channels
  - Pop-out (emphasize important information)
  - Discriminability (how many usable steps?)
  - Separability (judge each channel independently)
  - Relative vs. absolute judgement

[Munzner 14]

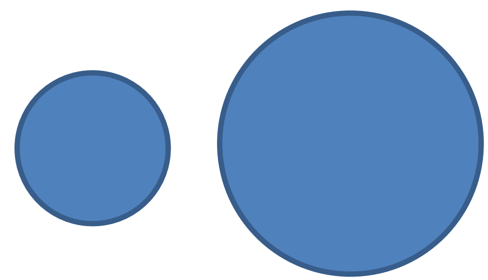
Mapping techniques

- Some visual channels can be compared more accurately





How much longer?



How much larger?

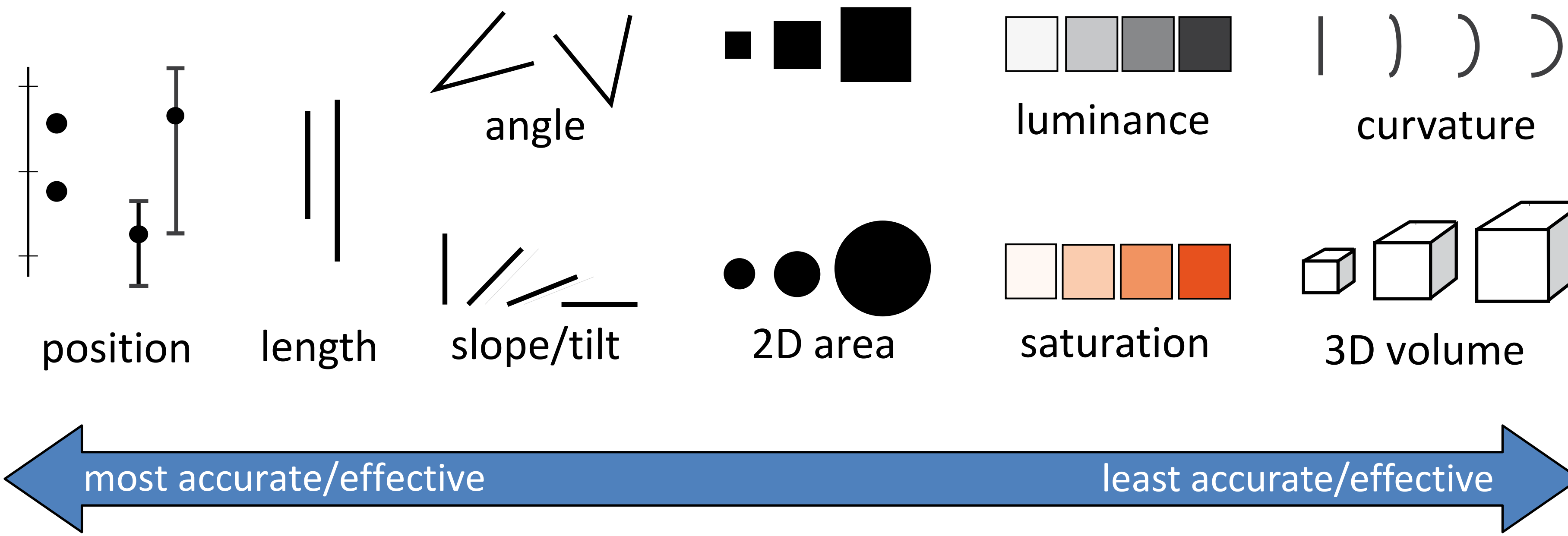
How much darker?

http://woodgears.ca/eyeball/

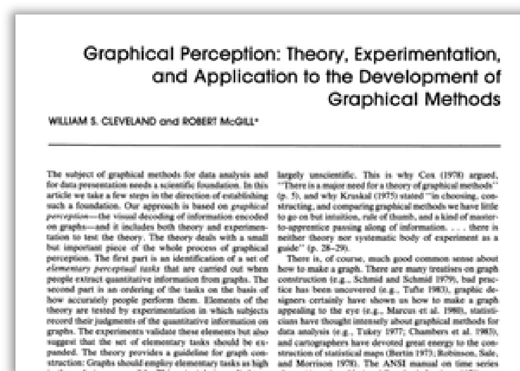
Mapping techniques

SIEMENS
Ingenuity for life

- Accuracy/effectiveness (quantitative data)
  - Ranking of vis. channels derived from empirical studies



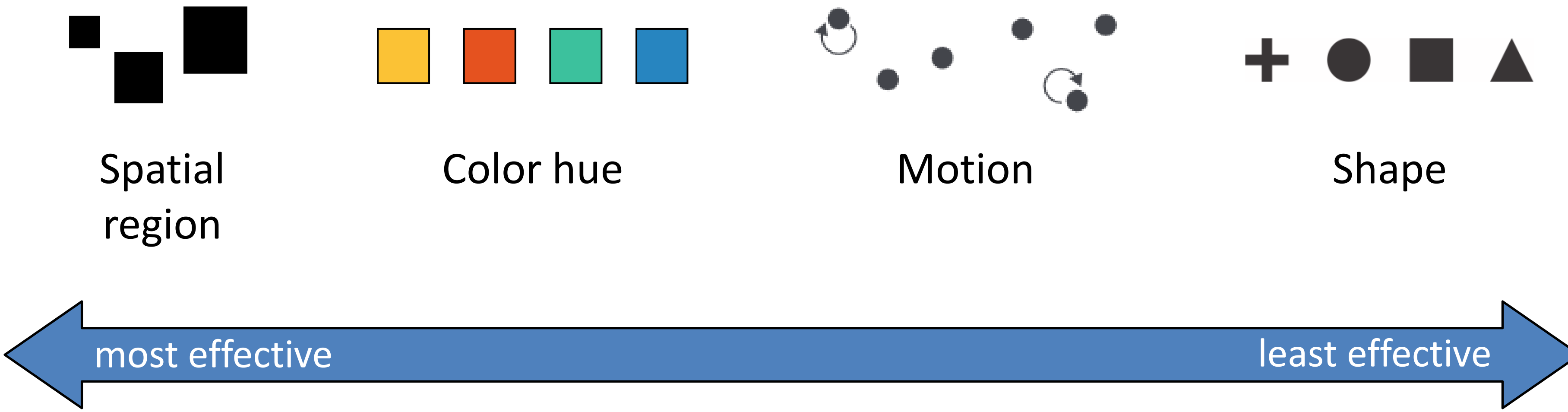
[Cleveland & McGill 84,
Heer and Bostock 10,
Munzner 14]


Graphical Perception: Theory, Experimentation, and Application to the Development of Graphical Methods

WILLIAM S. CLEVELAND and ROBERT McGILL*

Mapping techniques

- Accuracy/effectiveness (categorical data)
  - Ranking of vis. channels derived from empirical studies



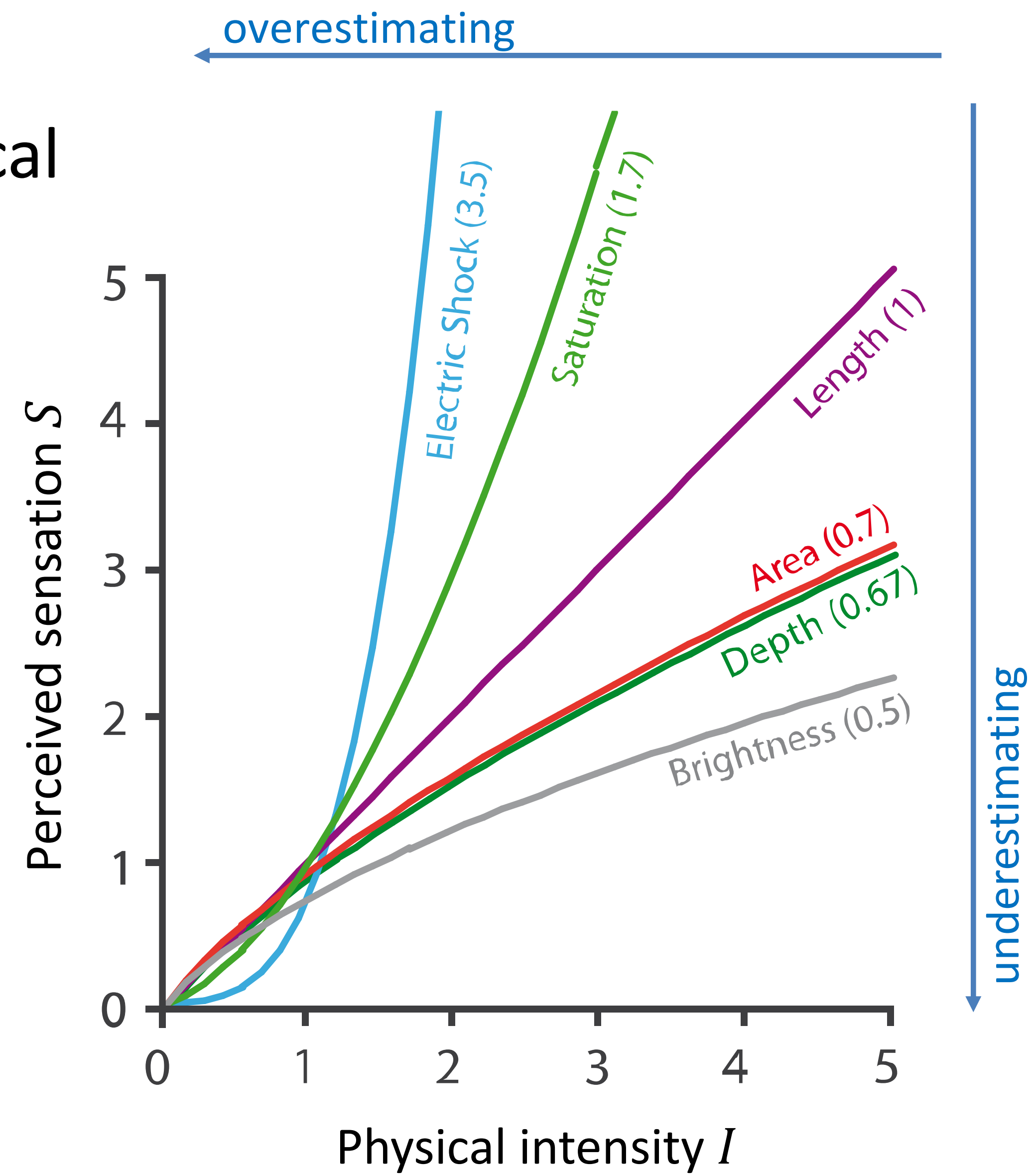
Expressive mapping:
Match type of visual channel to data type

[Munzner 14]

Mapping techniques

SIEMENS
Ingenuity for life

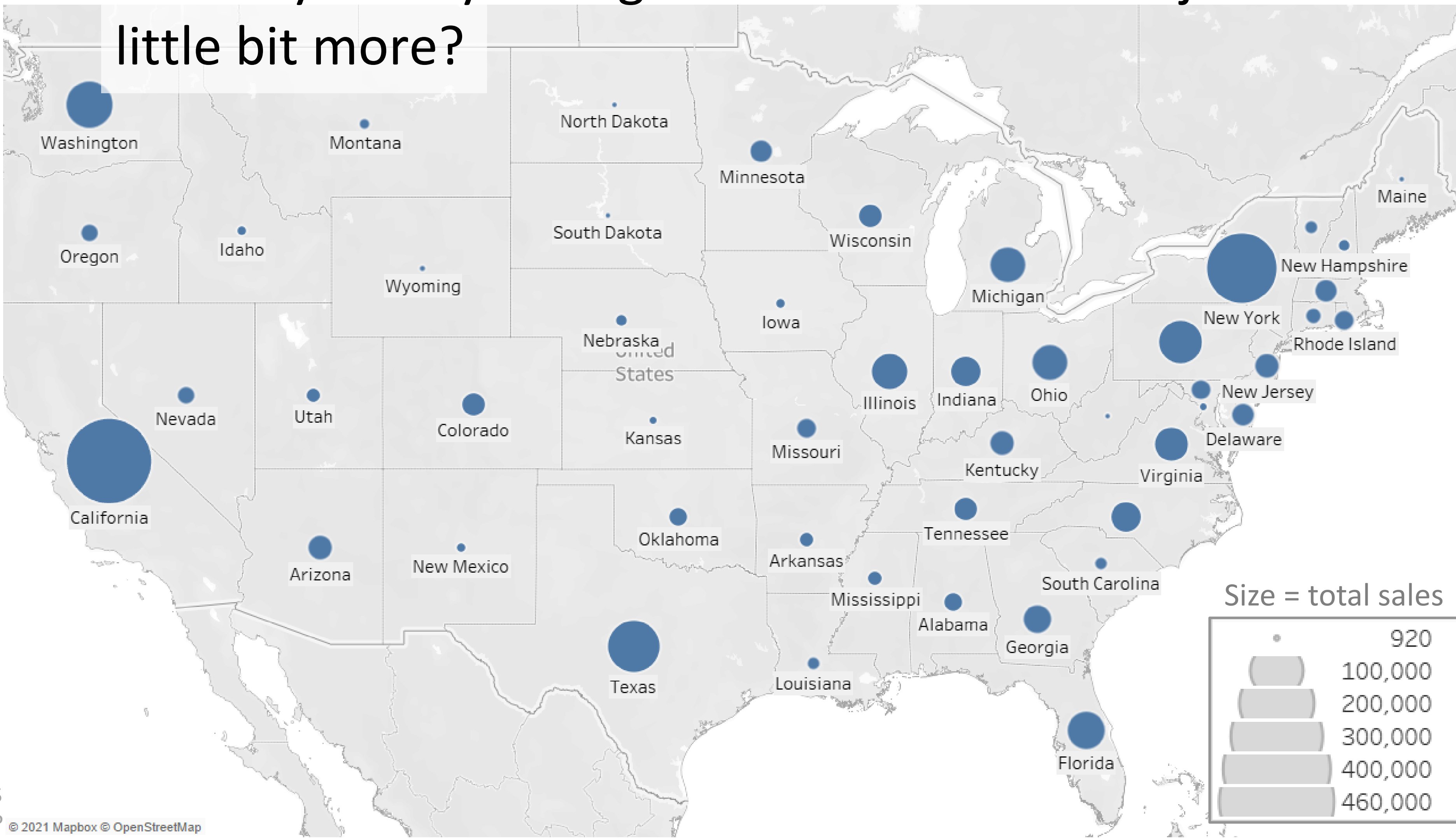
- Accuracy/effectiveness
  – Steven's psychophysical power law: $S = I^{\gamma}$
  - Length is accurate (linear)
  - Other channels are magnified or compressed



[Munzner 14]

Mapping techniques

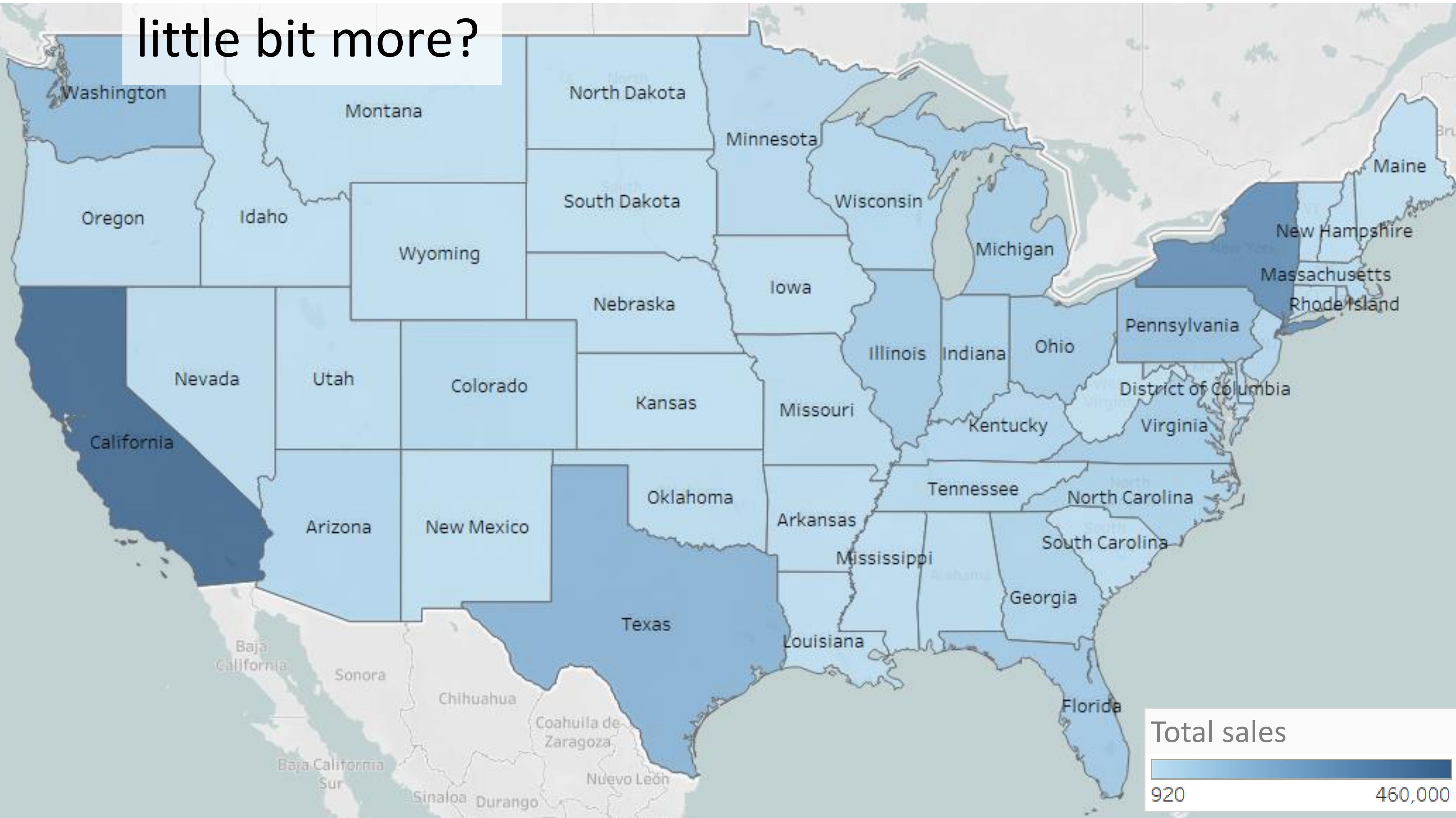
- Which are the top five states in terms of sales?
- Are they clearly selling more than the rest or just a little bit more?



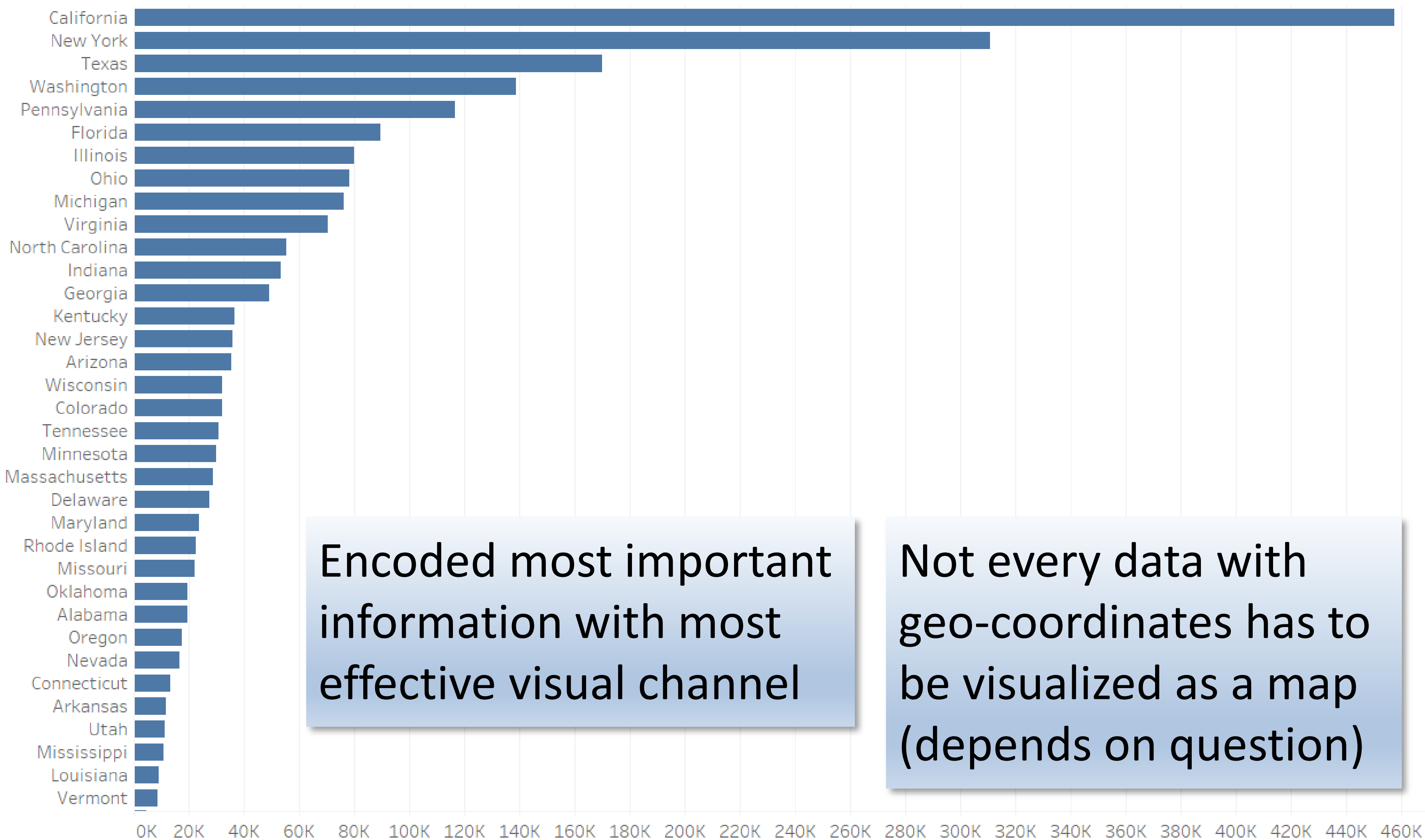
Mapping techniques

SIEMENS
Ingenuity for life

- Which are the top five states in terms of sales?
- Are they clearly selling more than the rest or just a little bit more?



- Which are the top five states in terms of sales?
- Are they clearly selling more than the rest or just a little bit more?



Encoded most important information with most effective visual channel

Not every data with geo-coordinates has to be visualized as a map (depends on question)

Mapping techniques

SIEMENS
Ingenuity for life

- Effectiveness principle
  - Some visual channels are better than others
  - Encode most important data attributes with most effective/accurate channels

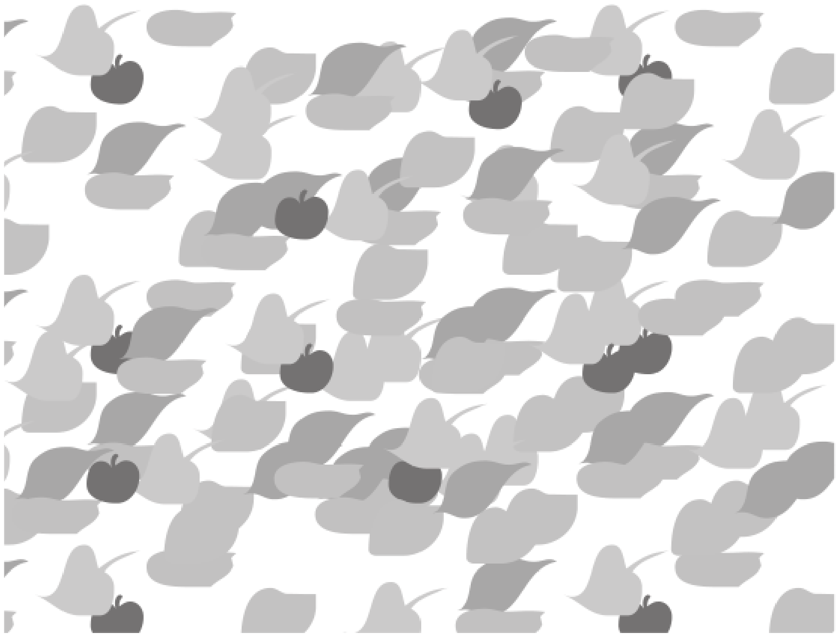
- Properties of visual channels
  - Pop-out (emphasize important information)
  - Discriminability (how many usable steps?)
  - Separability (judge each channel independently)
  - Relative vs. absolute judgement

[Munzner 14]

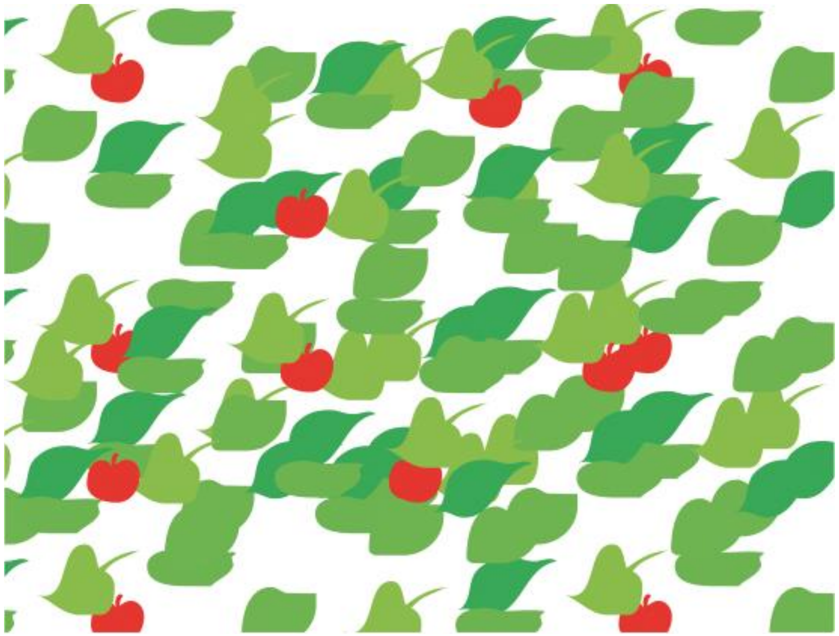
Mapping techniques

SIEMENS
Ingenuity for life

- Some visual channels “pop out”



Where are the cherries?



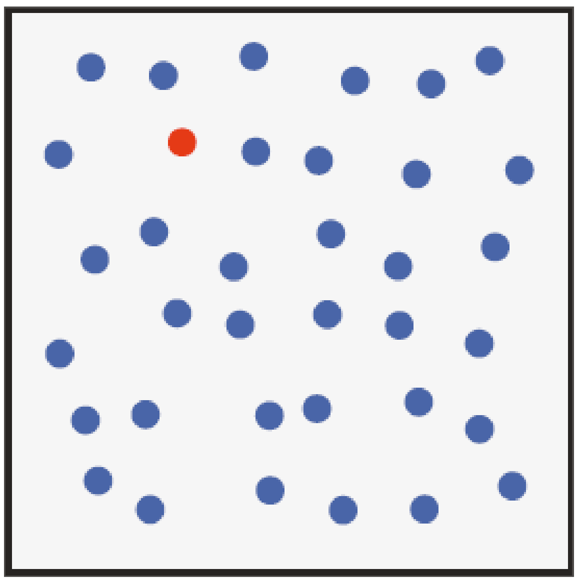
Hunters & gatherers

[Ware 04]

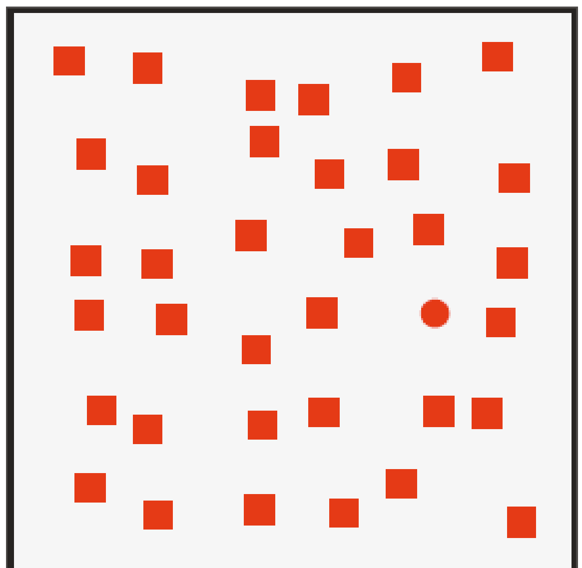
Mapping techniques

SIEMENS
Ingenuity for life

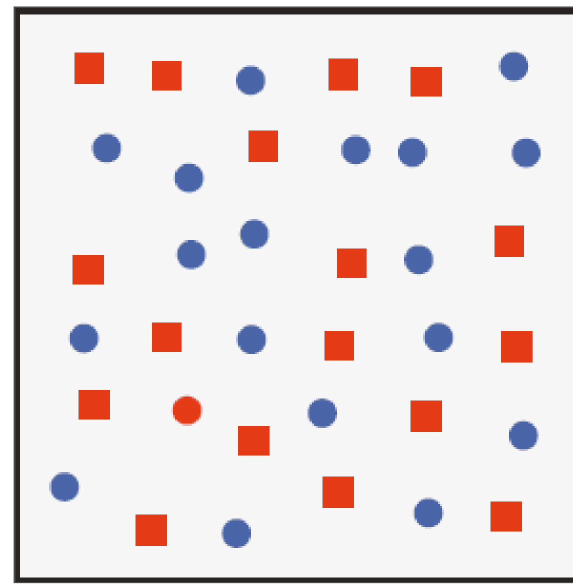
- Pop-out
  - Preattentive processing: automatic and parallel detection of basic features in visual information (200-250 msec)
  - Speed independent of distractor count
  - Works on many individual channels



Color



Shape



Combination of channels usually requires serial search

www.csc2.ncsu.edu/faculty/healey/PP/

The value of visualization

SIEMENS
Ingenuity for life

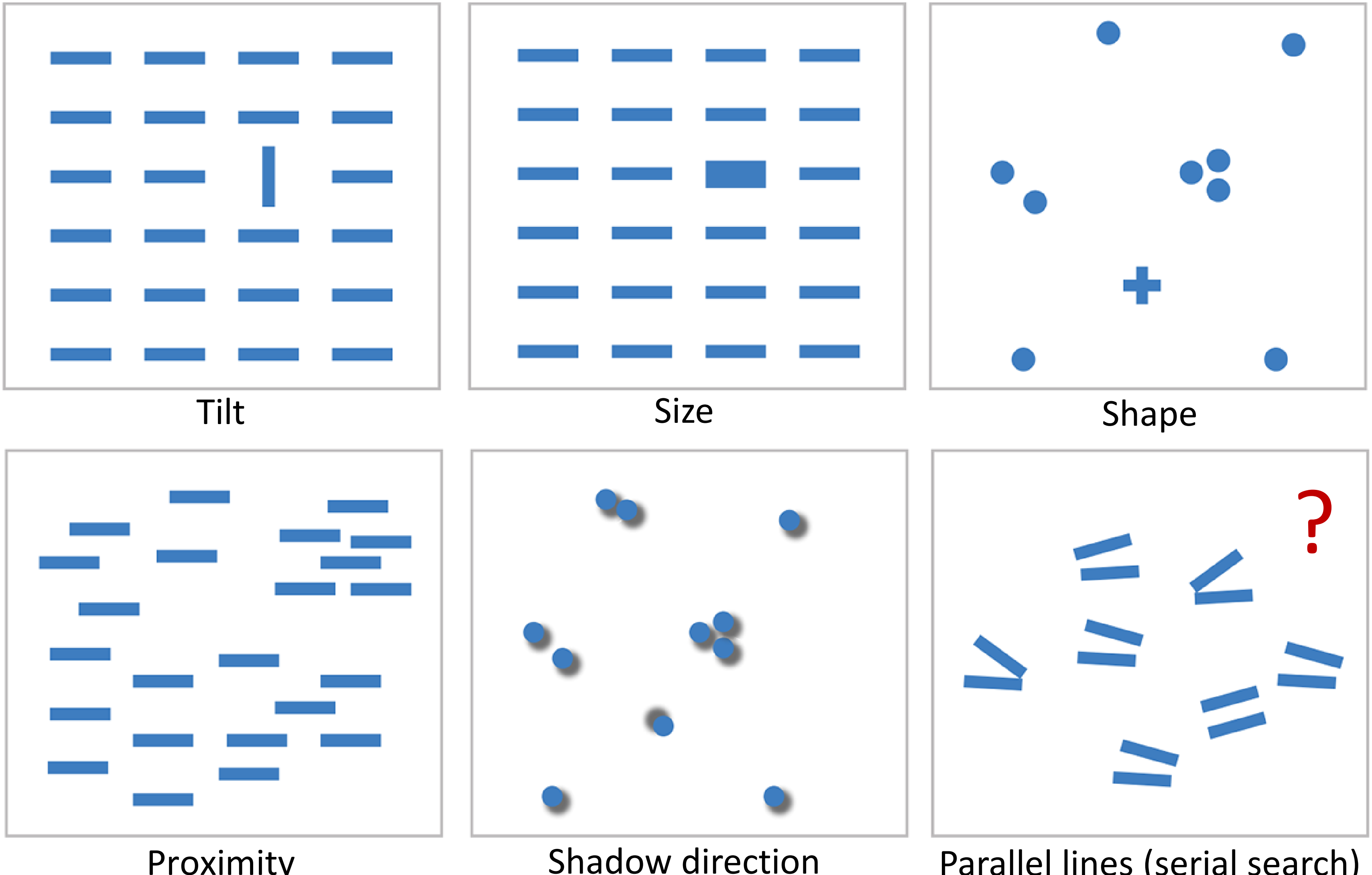
2	1	4	3	9	5	6	7	8	2	3	6	5	9	4	0	1
6	7	9	3	4	9	0	5	6	2	5	8	4	0	5	2	6
9	8	2	6	3	5	9	3	2	9	3	7	2	6	3	4	8
8	1	6	2	3	8	7	9	5	0	2	3	9	2	8	4	3
0	9	1	8	5	4	2	9	4	7	4	6	8	4	0	2	9
3	9	2	7	3	6	6	5	2	9	4	0	4	9	4	8	6
5	2	4	3	6	4	8	1	0	3	9	4	8	4	7	3	2
8	6	2	3	0	8	7	3	6	2	5	4	4	8	3	5	0

https://vimeo.com/29684853

Mapping techniques

SIEMENS
Ingenuity for life

- Pop-out

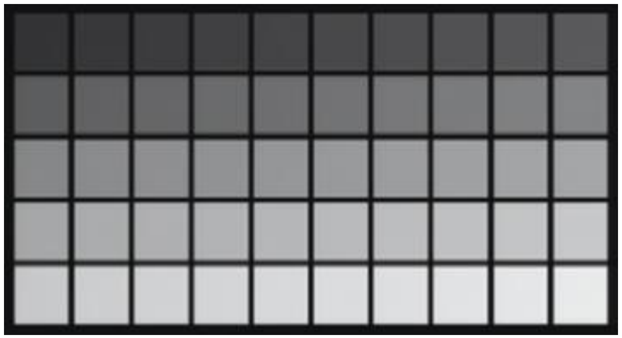


[Munzner 14]

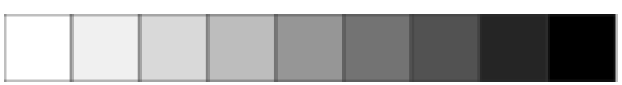
Mapping techniques

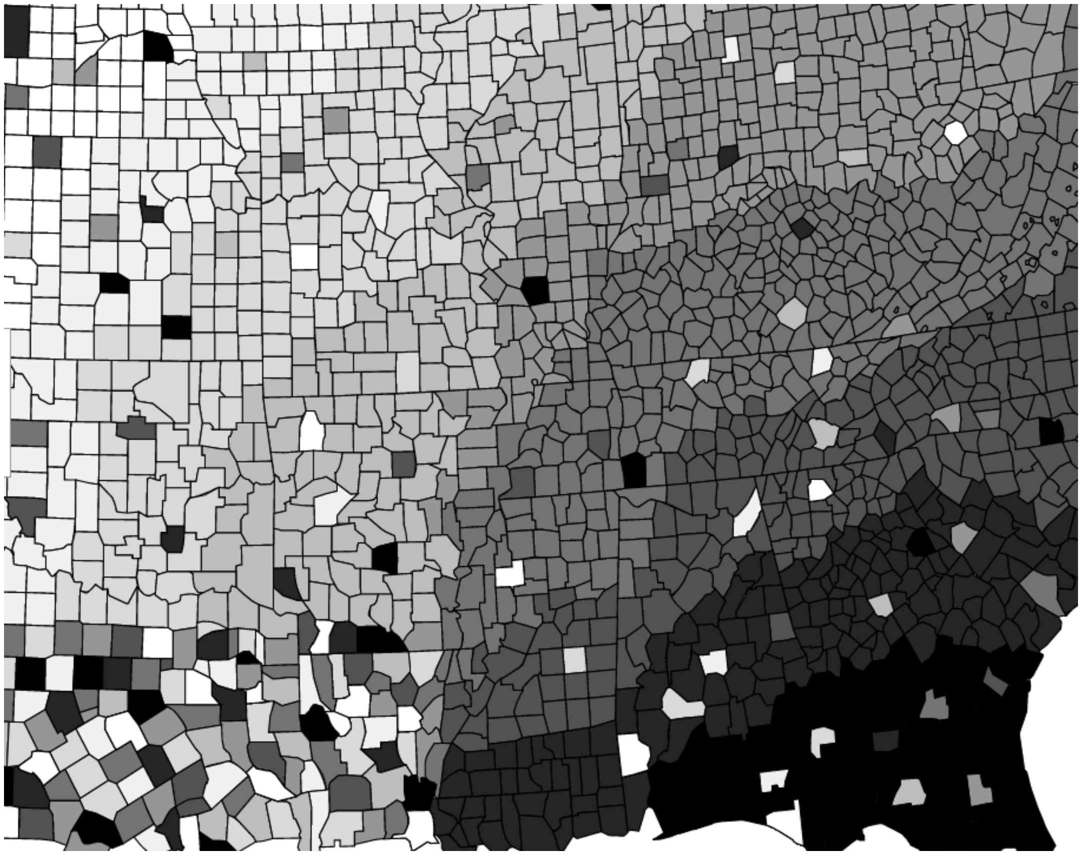
SIEMENS
Ingenuity for life

- Discriminability: How many usable steps?
  - Must be sufficient for number of discriminable bins



We can only distinguish a limited number of colors / brightness levels



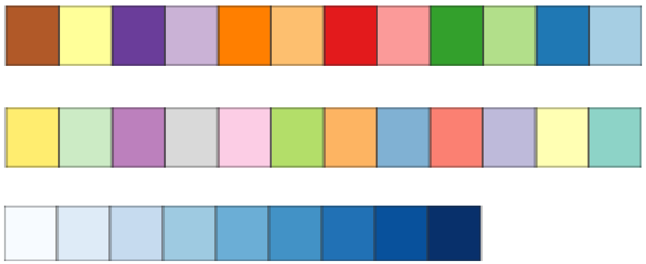


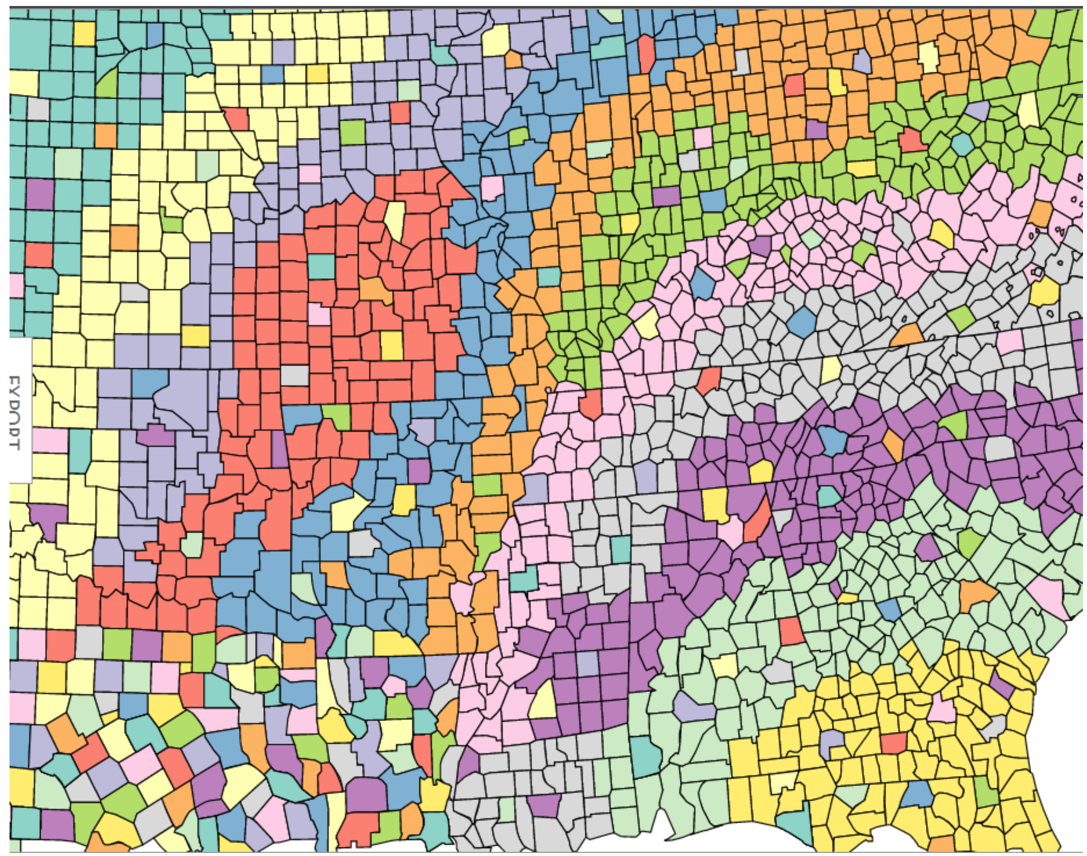
colorbrewer2.org

Mapping techniques

- Discriminability: How many usable steps?
  - Must be sufficient for number of discriminable bins

9-12 color hues



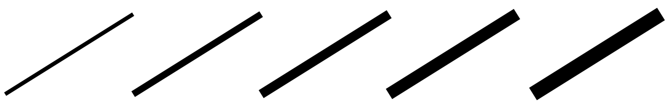


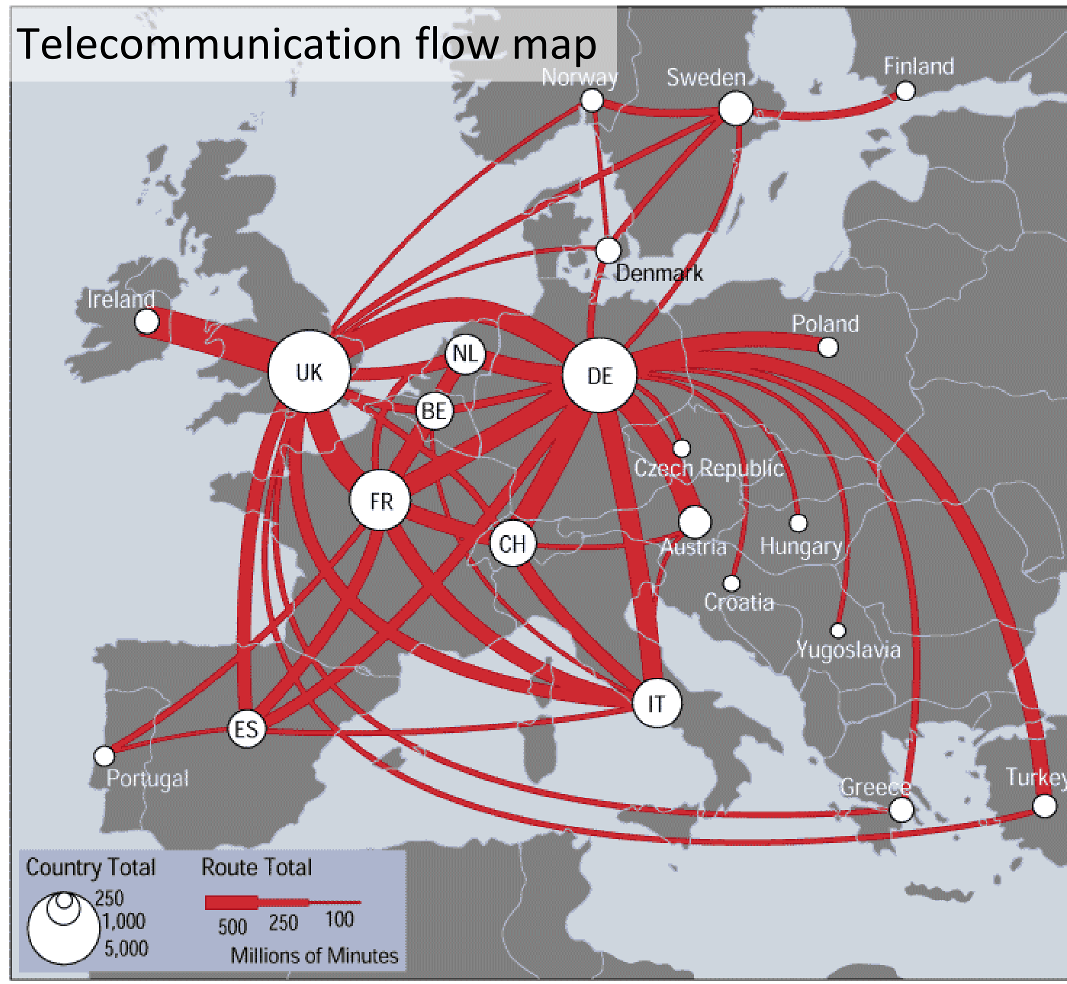
colorbrewer2.org

Mapping techniques

- Discriminability: How many usable steps?
  – Must be sufficient for number of discriminable bins

~5 size bins





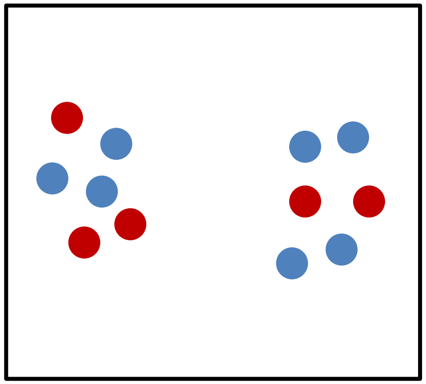
www.mundi.net/maps/maps_014/

Mapping techniques

SIEMENS
Ingenuity for life

- Separable vs. integral visual channels

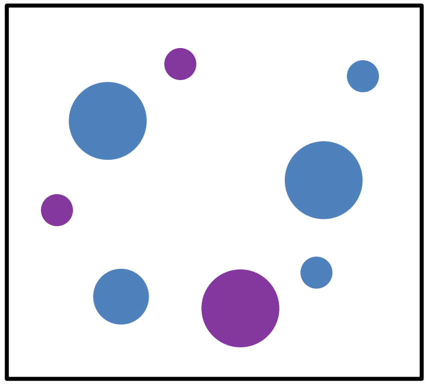
Color + position



Fully separable

2 groups each

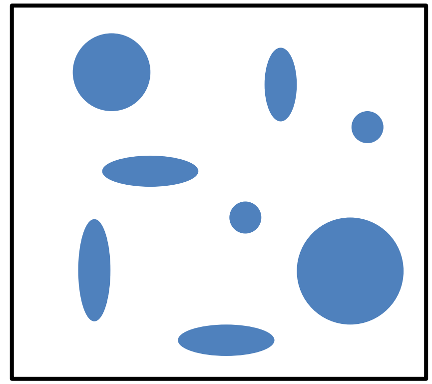
Color + size



Some interference

2 groups each

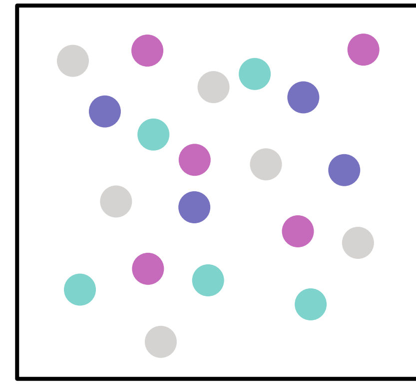
Width + height



Some/significant interference

3 groups total: integral area

Red + green



Major interference

4 groups total: integral color

Separable visual channels ⟷ Integral visual channels

We are able to judge each visual channel independently

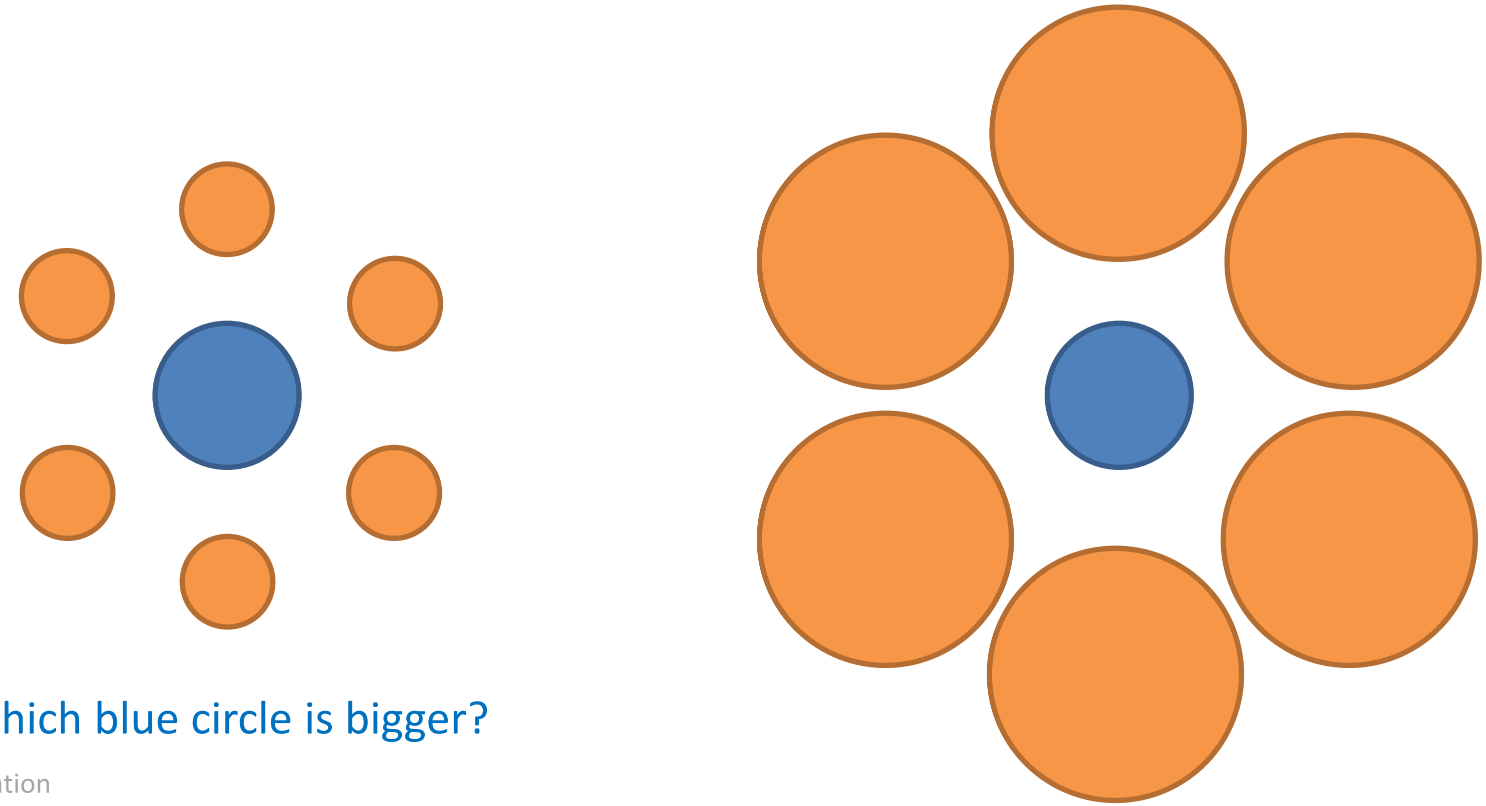
Channels are viewed holistically

[Ware 13, Munzner 14]

Mapping techniques

SIEMENS
Ingenuity for life

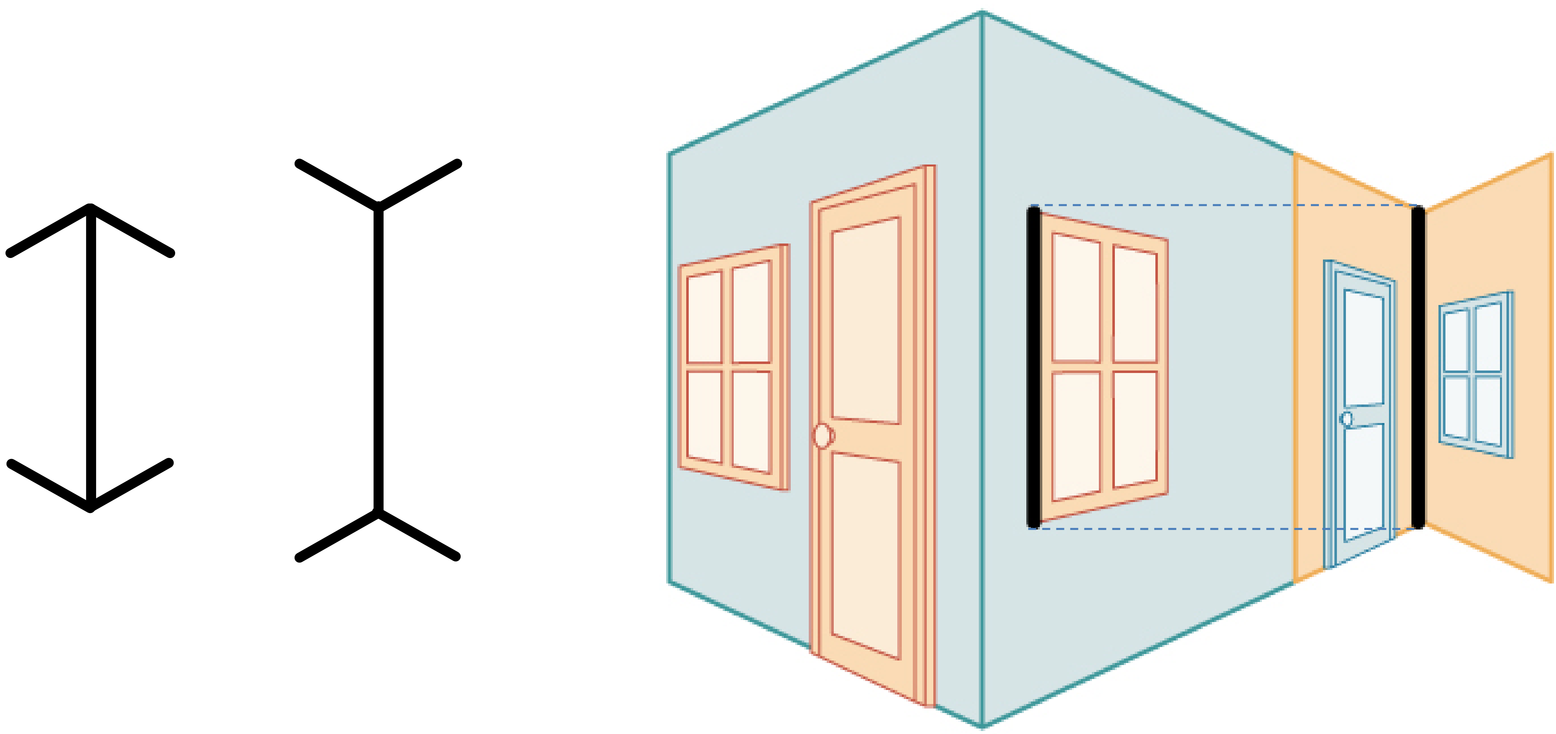
- Relative vs. absolute judgments
  - Perception highly context-dependent
  - Perceptual system mostly operates with relative judgments, not absolute ones



Which blue circle is bigger?

Mapping techniques

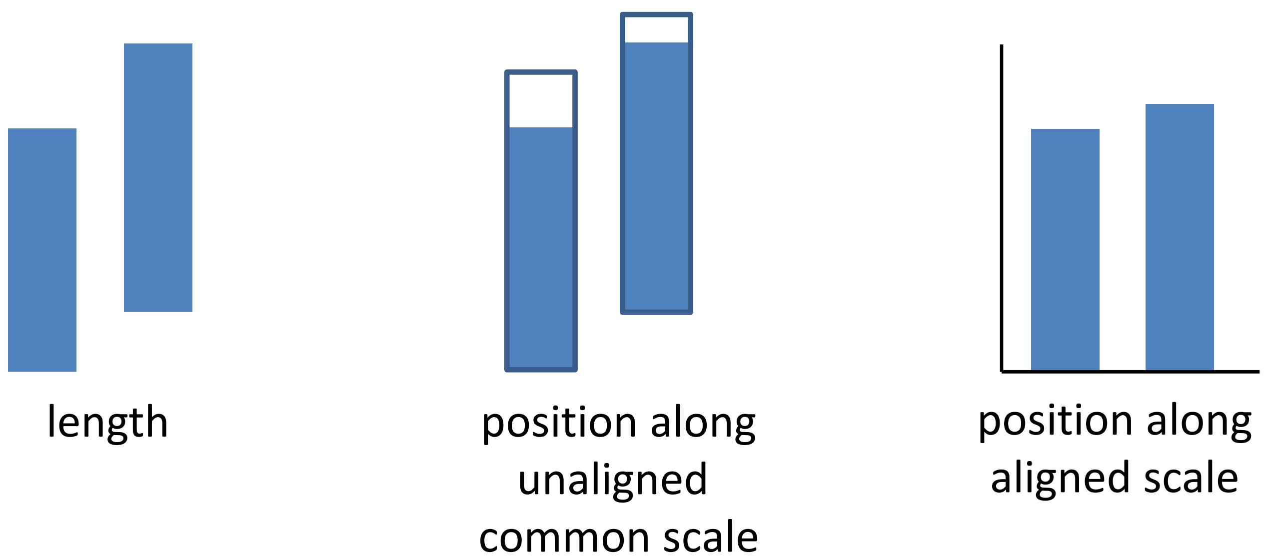
- Relative vs. absolute judgments



Müller-Lyer illusion

Mapping techniques

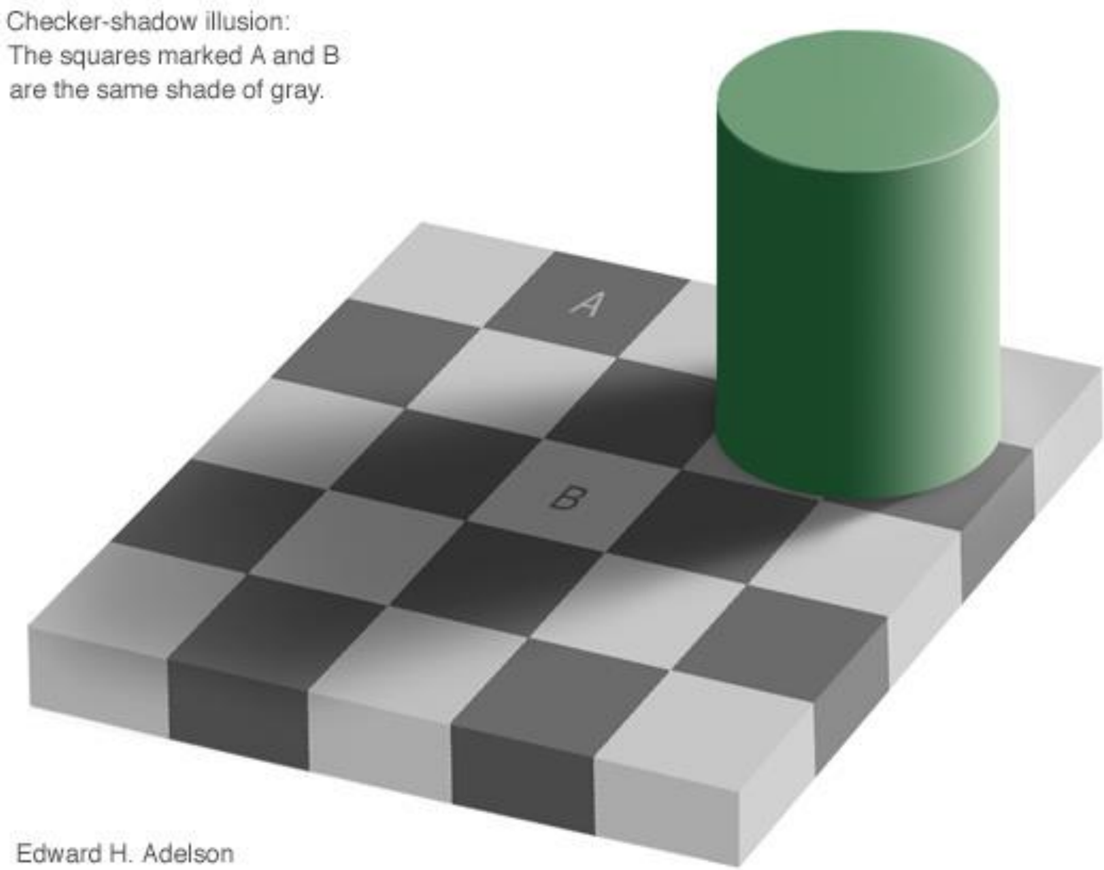
- Relative vs. absolute judgments
  – Perceptual system mostly operates with relative judgments, not absolute ones
  – Weber's Law: just-noticeable difference is a fixed percentage of the magnitude of the stimuli (e.g., bar length)
    - filled rectangles differ in length by 1:9 → difficult judgment
    - white rectangles differ in length by 1:2 → easy judgment



[Munzner 14]

Mapping techniques

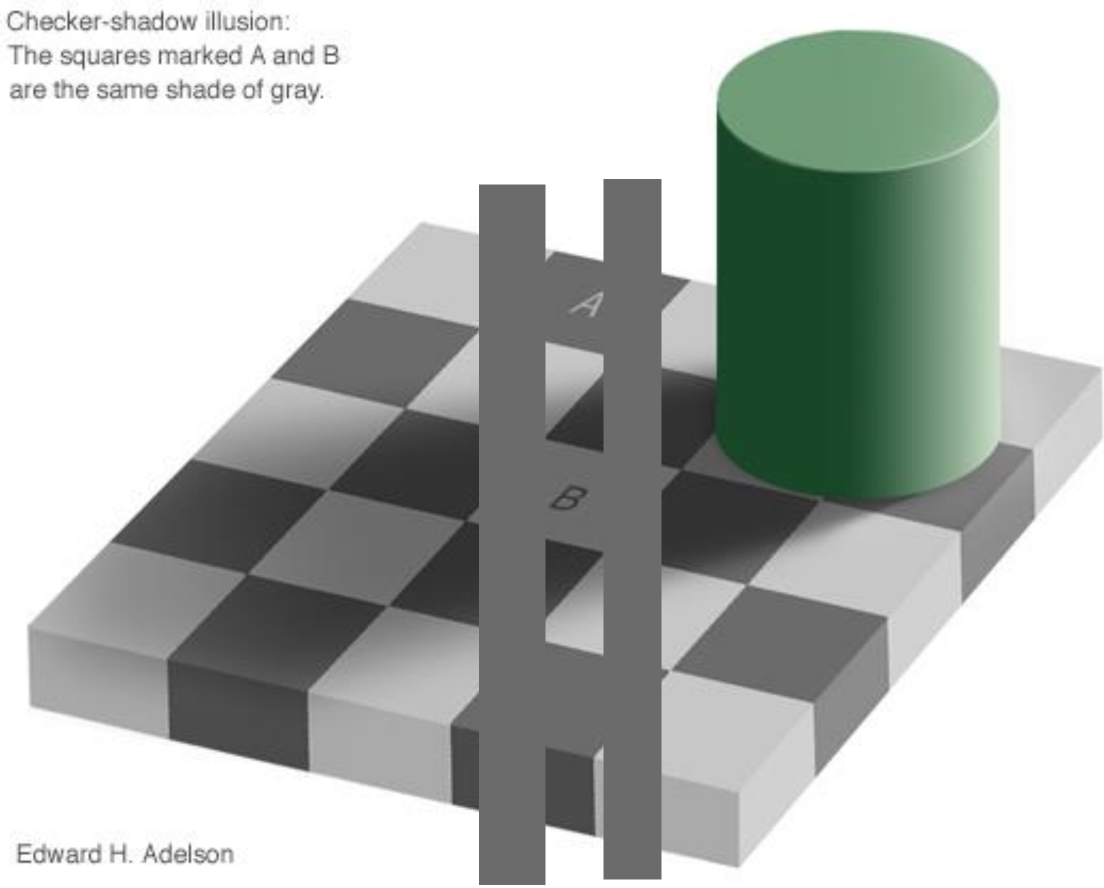
- Perceived color is highly context dependent



http://web.mit.edu/persci/people/adelson/checkershadow_illusion.html

Mapping techniques

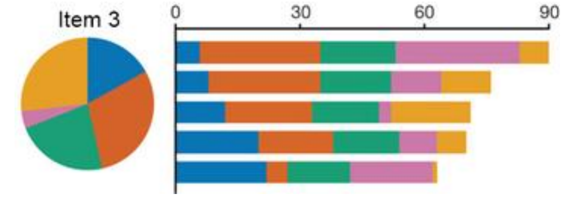
- Perceived color is highly context dependent



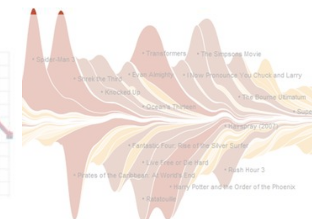
http://web.mit.edu/persci/people/adelson/checkershadow_illusion.html

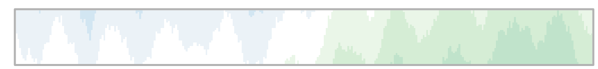
Diagram techniques

- Categorical + quantitative data
  - Bar/pie chart, stacked bars

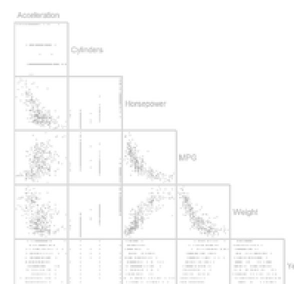


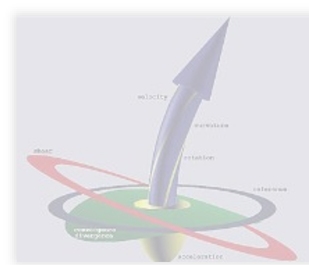
- Time-dependent data
  - Line graph, ThemeRiver, Horizon graph





- Single and multiple variables
  - Histogram, scatterplot, parallel coordinates
  - Glyphs, color mapping





Recap: Attribute types

SIEMENS
Ingenuity for life

- Quantitative (numerical, measurable)
  - Objective data produced through a systematic process, not subject to interpretation (e.g., length, mass, temperature)
  - Metric scale – allows measure of distance
  - Continuous (real) or discrete (distinct & separate values)

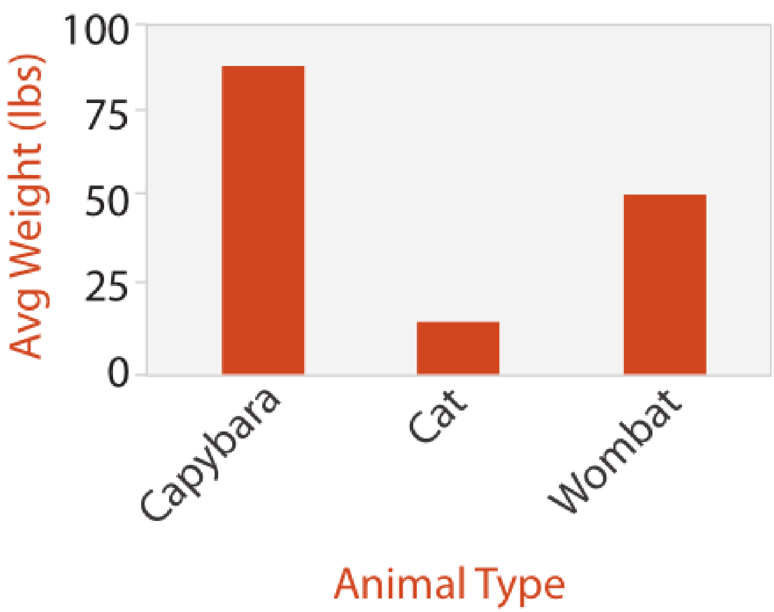
- Qualitative (categorical, not measurable)
  - No metric scale; cannot be measured
  - Requires a subjective decision in order to be categorized
  - Discrete

Expressive mapping:
Match type of visual channel to data type

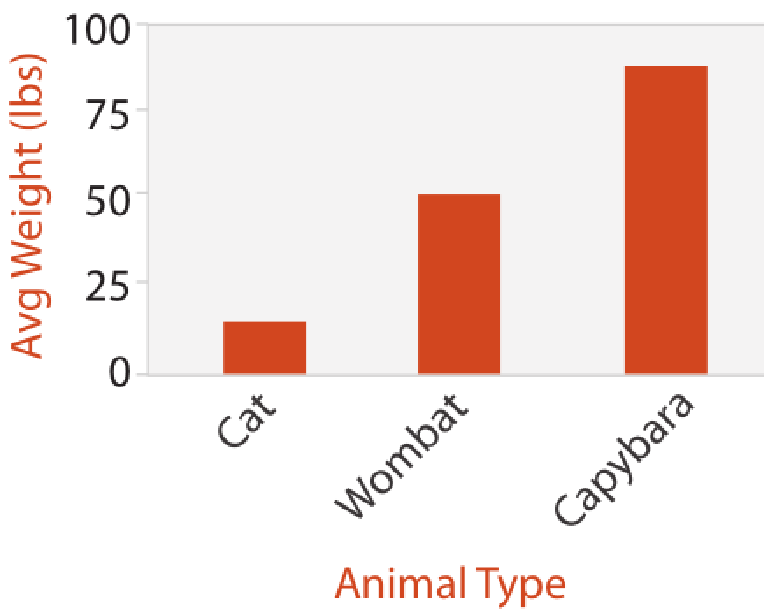
Diagram techniques

SIEMENS
Ingenuity for life

- Bar chart
  - Attrib. 1: categorical → horizontal position
  - Attrib. 2: quantitative (dependent) → length/vertical position



Ordered by name

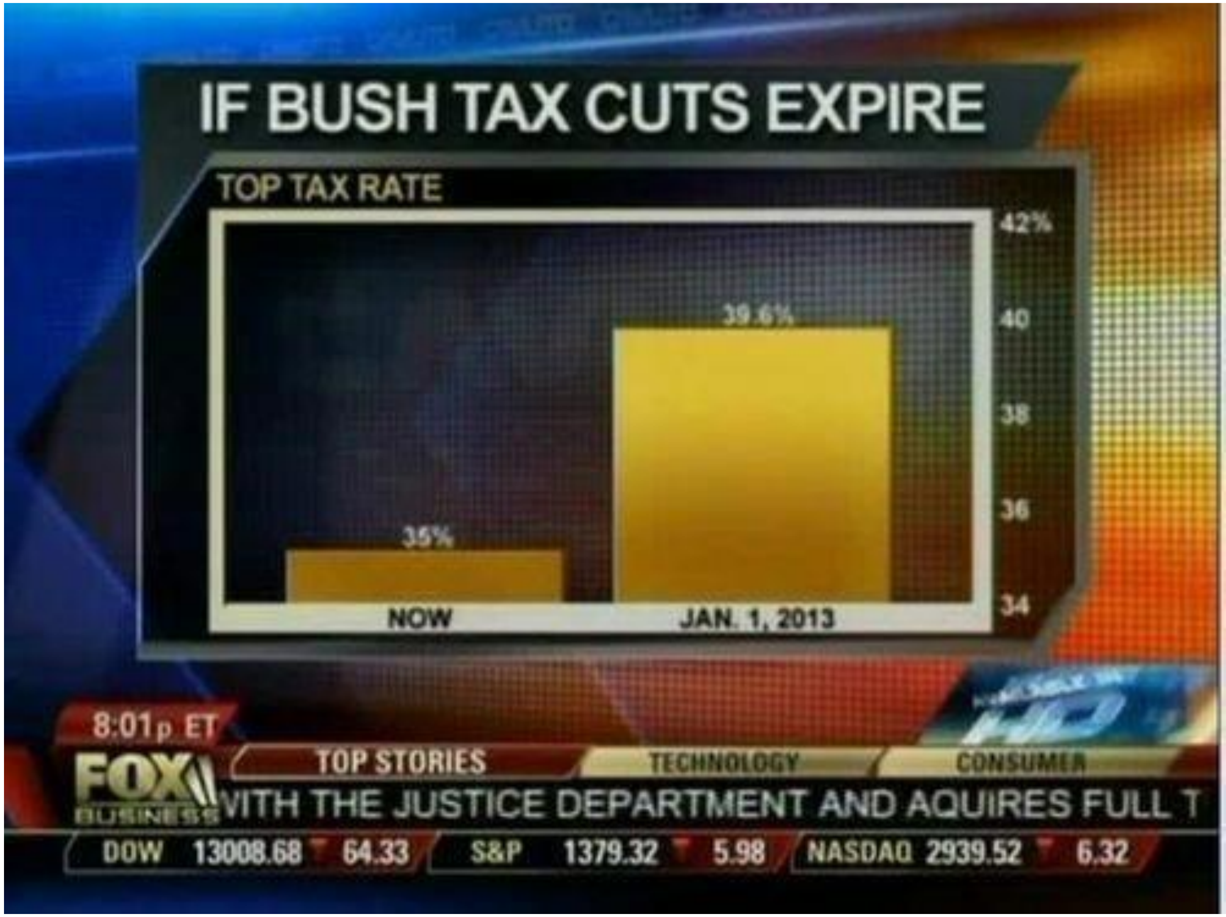


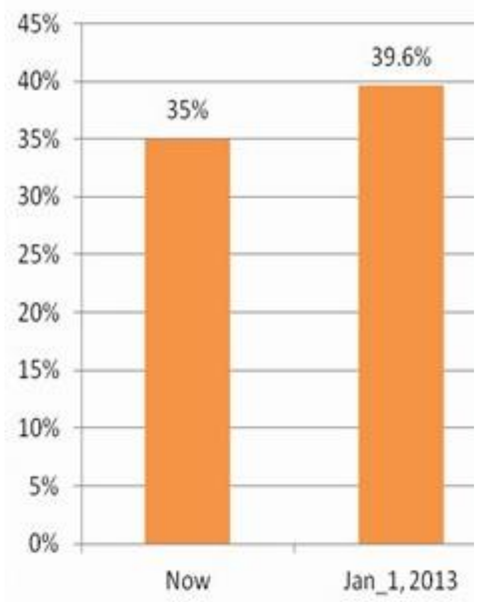
Ordered by weight

Diagram techniques

SIEMENS
Ingenuity for life

- Bars should always start at zero!



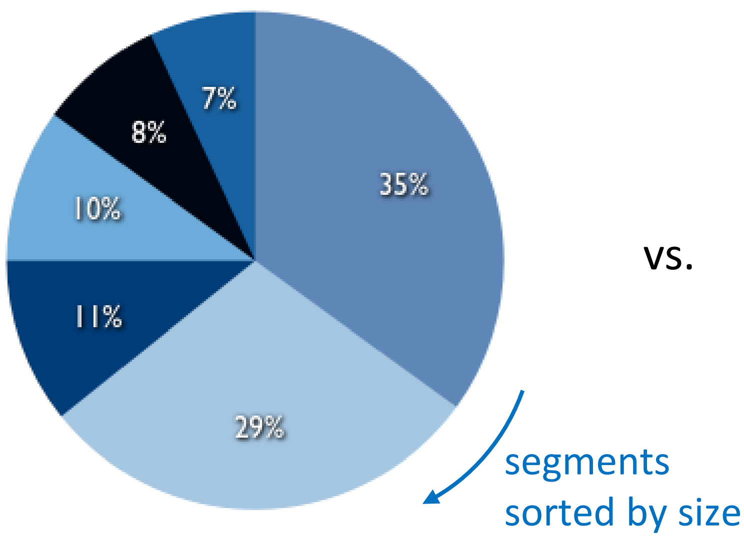


Taking misleading statistics to a new level, KDnuggets, 2012.
http://www.kdnuggets.com/2012/12/taking-misleading-statistics-to-a-new-level.html

Diagram techniques

SIEMENS
Ingenuity for life

- Pie chart
  - Pie chart splits population (100%) into parts
  - Attrib. 1: categorical → color
  - Attrib. 2: quantitative (dependent) → angle
  - However, angle/area less accurate than bar length



vs.

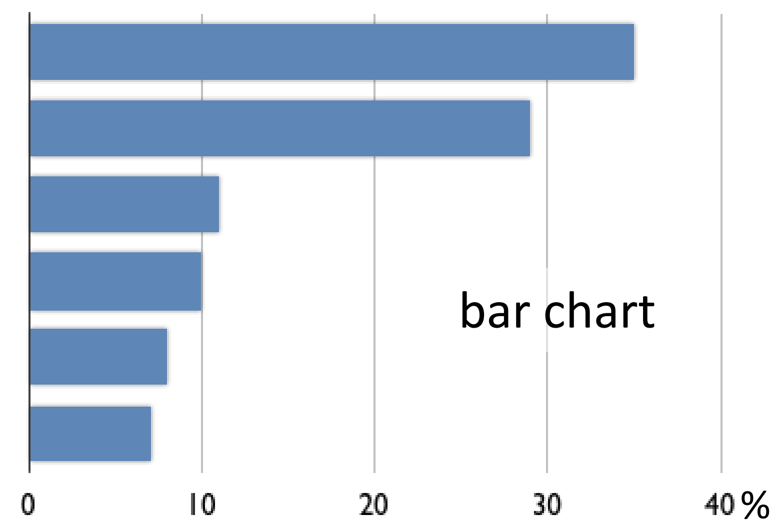


Diagram techniques

- Pie chart vs. Bar chart
  - Angle/area judgment less accurate than bar length

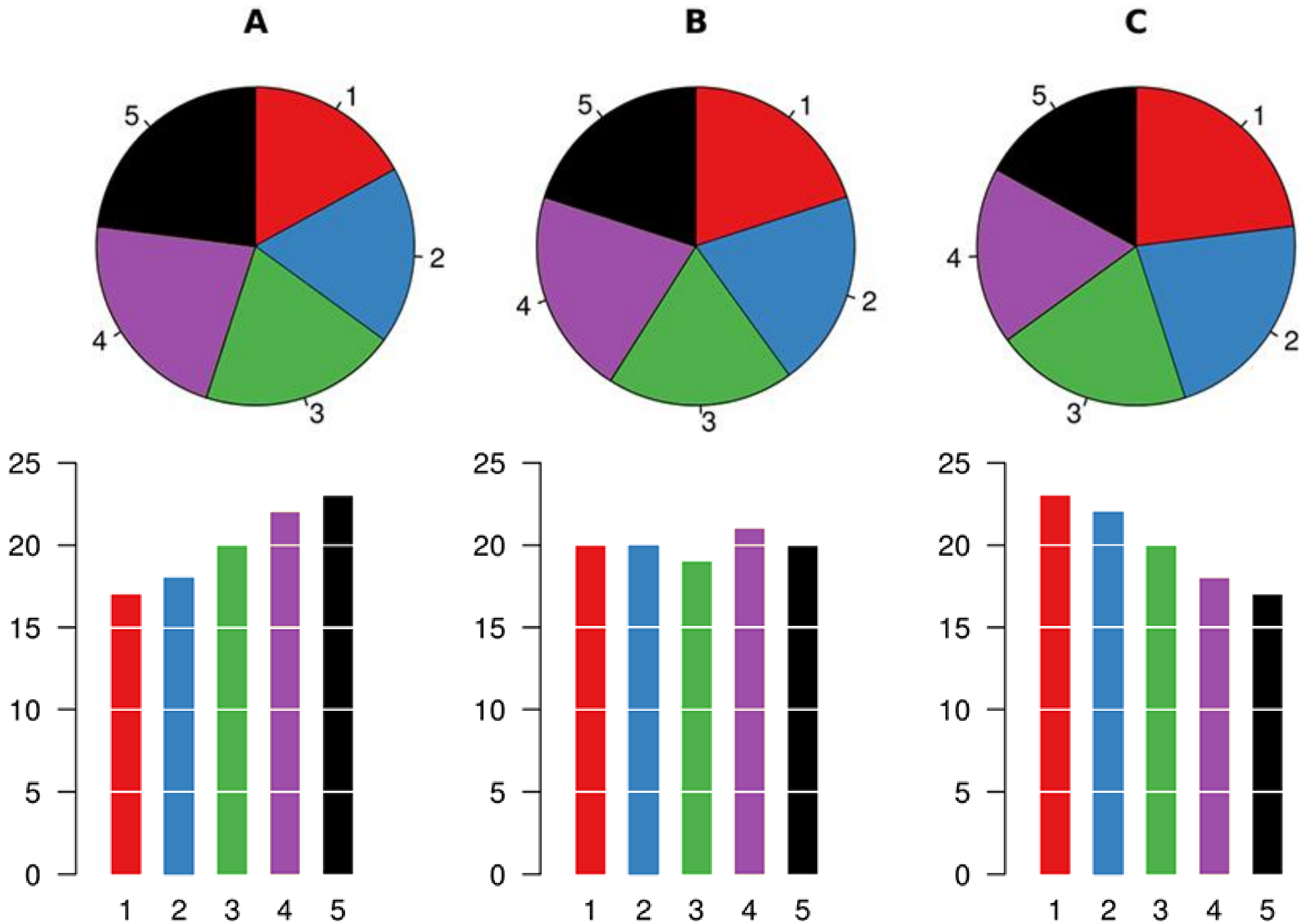
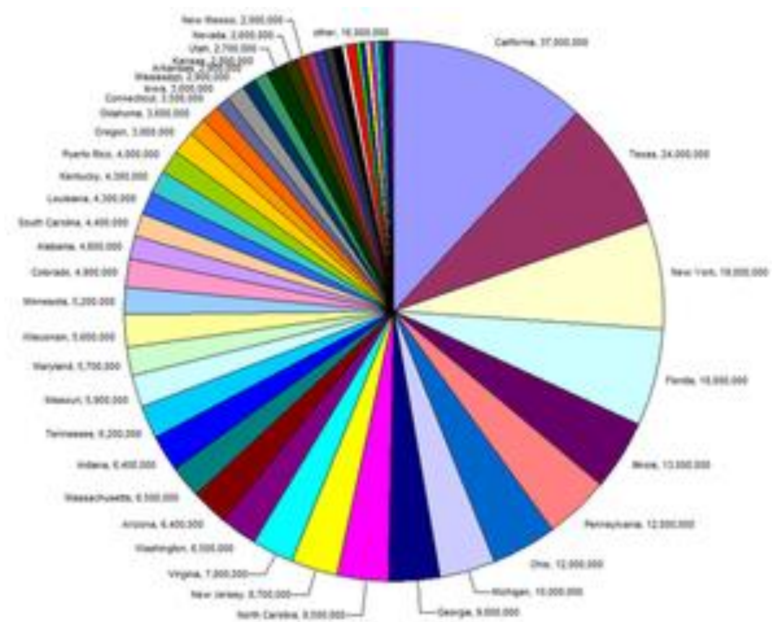


Diagram techniques

- When to use a pie chart?
  - Often bar chart is a better choice!
  - Do the parts make up a meaningful whole?
  - Are the parts mutually exclusive?
  - Do you want to compare the parts to each other or the parts to the whole?
  - How many parts do you have?

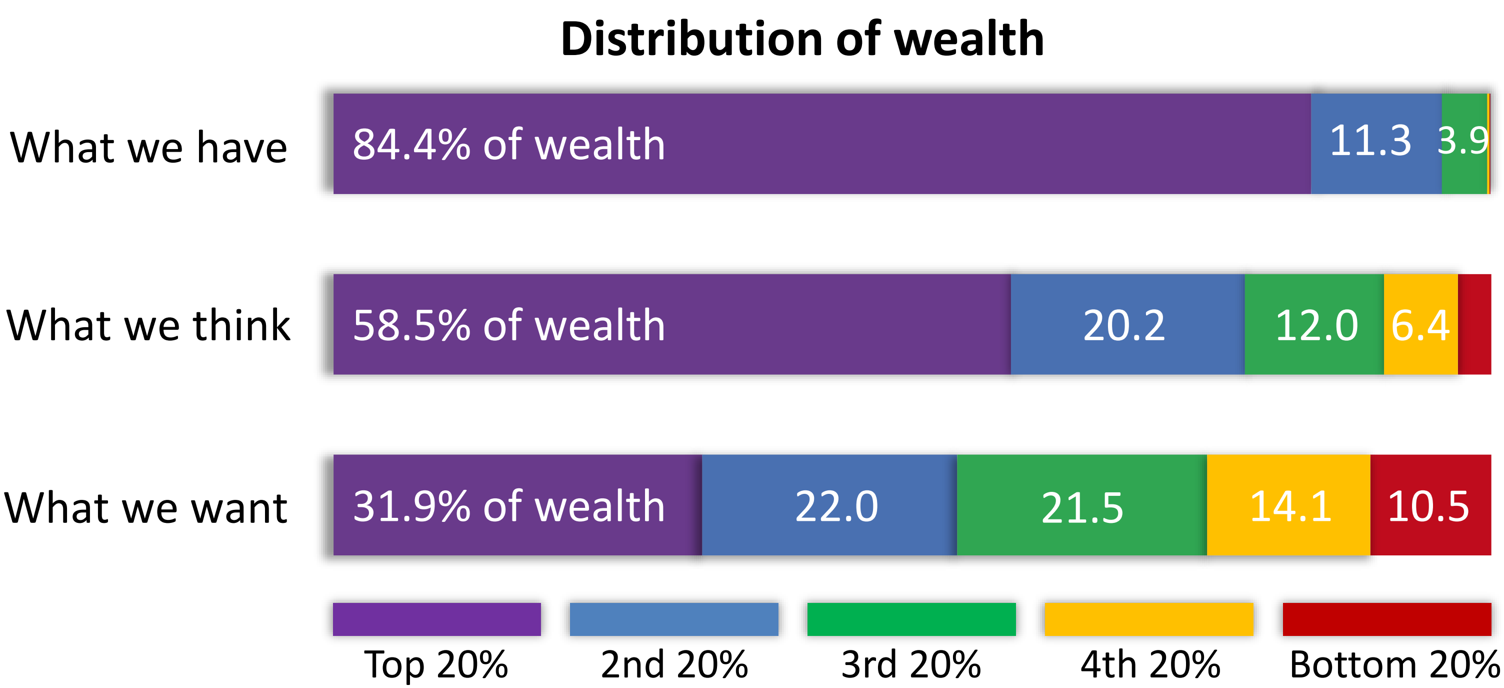


R. Kosara, eagereyes.org/techniques/pie-charts

Diagram techniques

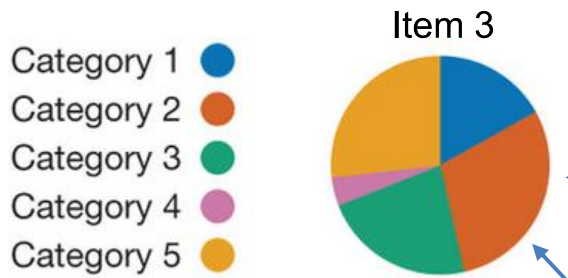
SIEMENS
Ingenuity for life

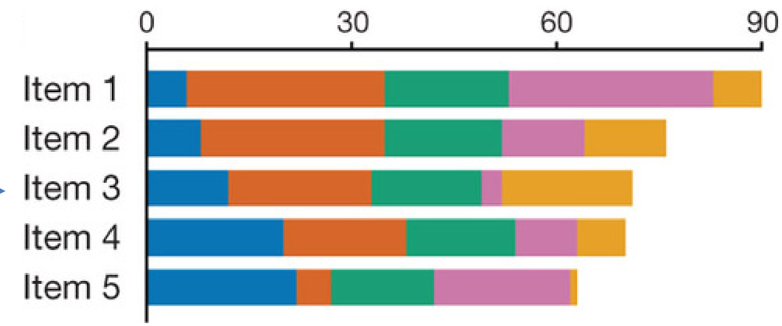
- Stacked bar chart
  - Quantitative data wrt 2 categorical vars (horizontal & vertical)
  - Investigate part-to-whole relationship (100%)
  - Length and color hue



Dan Ariely at [TED.com](TED.com)

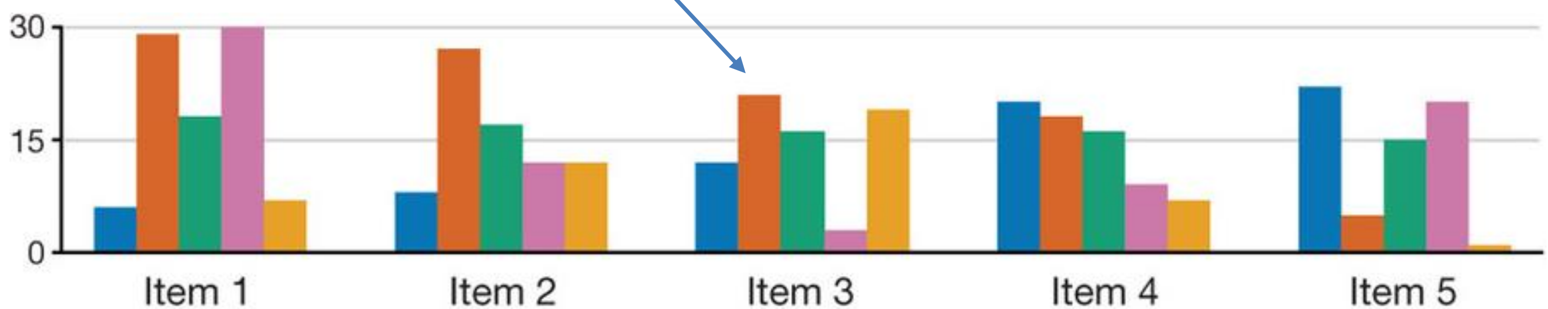
Variations of pie & bar charts



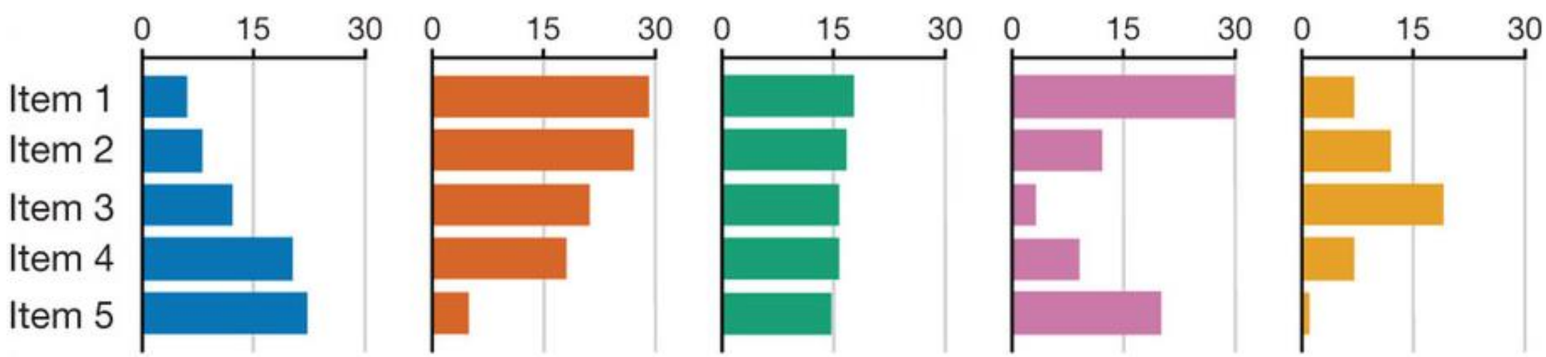


Pie chart: compare values in different categories

Stacked bar charts: compare overall values across items, but also show contribution per category



Grouped bar charts: compare values across categories within each item



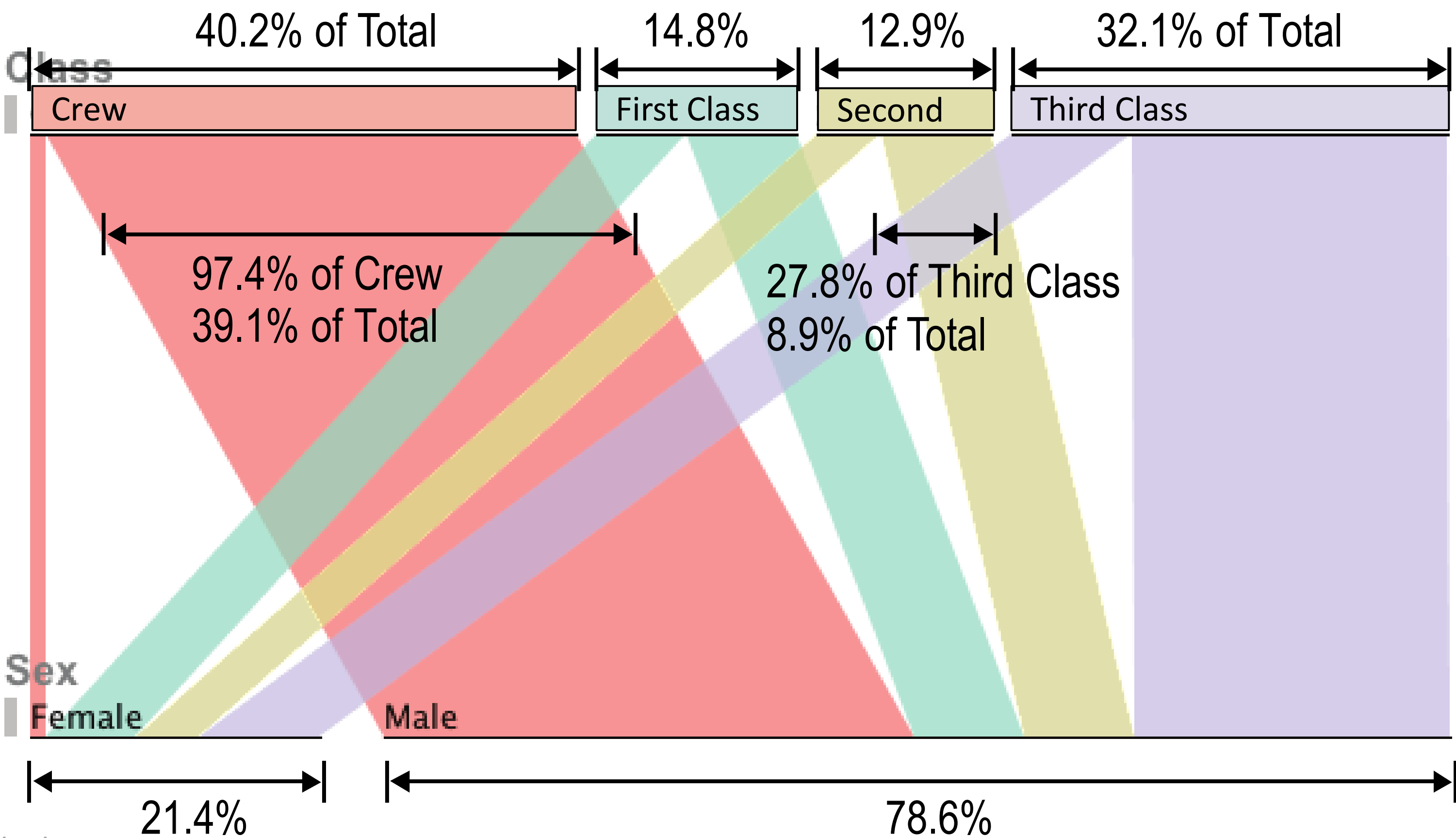
Layered bar charts: compare values within categories

[Streit & Gehlenborg 14]

Diagram techniques

SIEMENS
Ingenuity for life

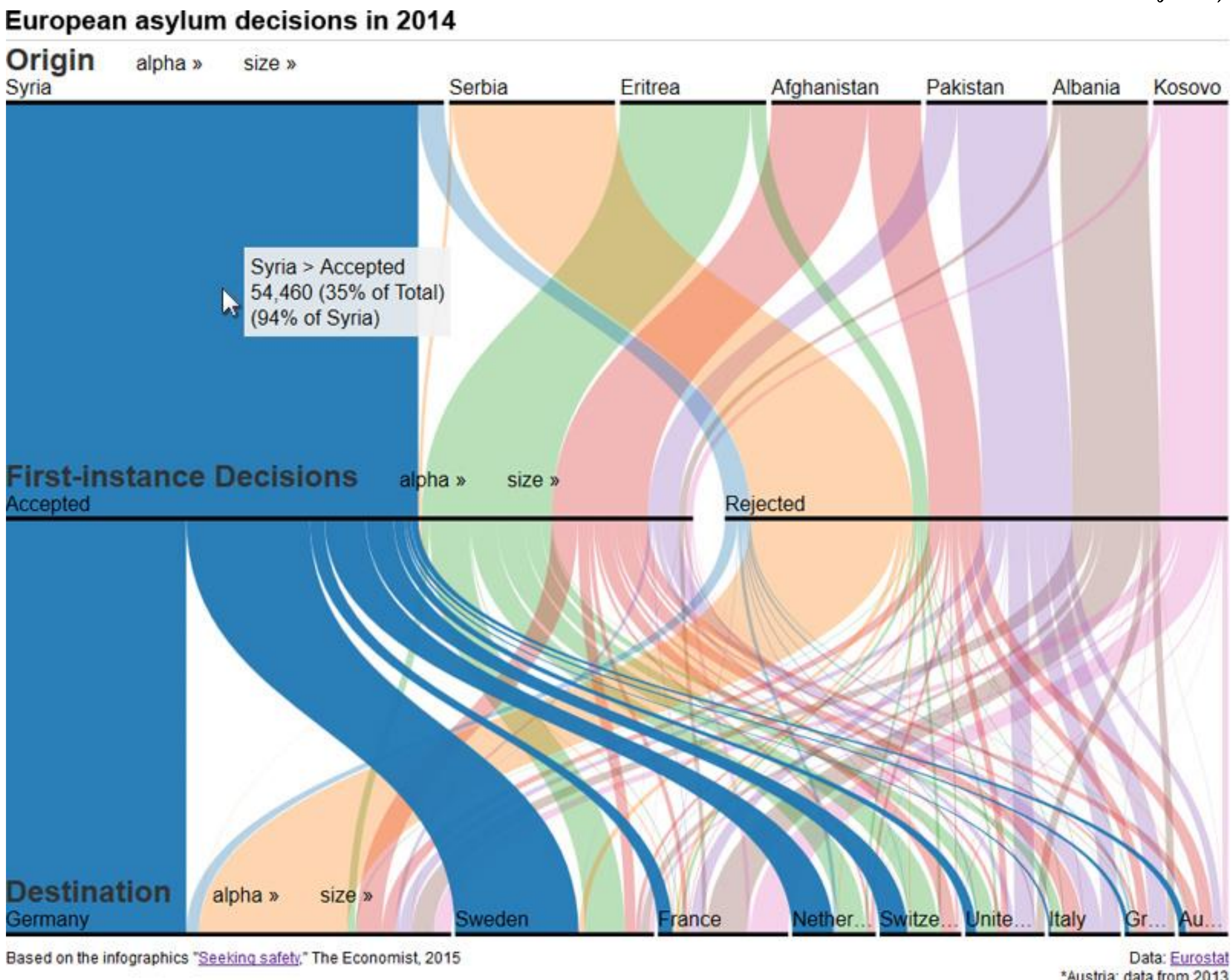
- Parallel sets
  - Quantitative data wrt. multiple categorical attributes
  - Shows connections and proportions



[Bendix, Kosara, Hauser 2005]

Diagram techniques

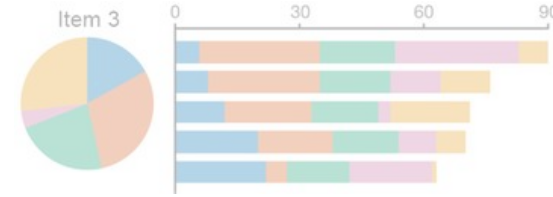
SIEMENS
Ingenuity for life



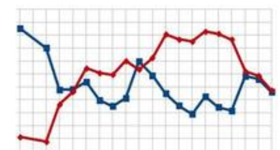
multivis.net/lecture/parallel-sets.htm

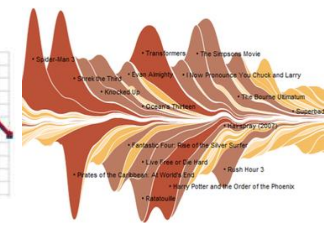
Diagram techniques

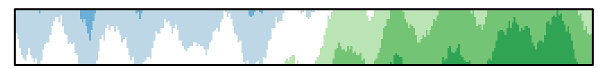
- Categorical + quantitative data
  - Bar/pie chart, stacked bars



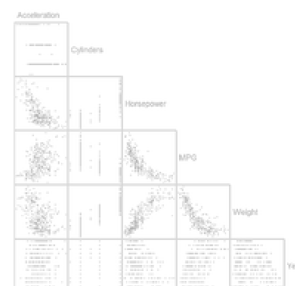
- Time-dependent data
  - Line graph, ThemeRiver, Horizon graph



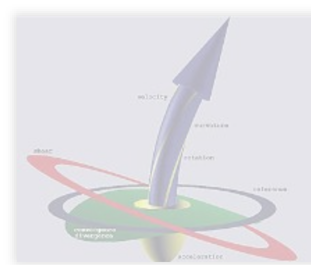




- Single and multiple variables
  - Histogram, scatterplot, parallel coordinates



  - Glyphs, color mapping



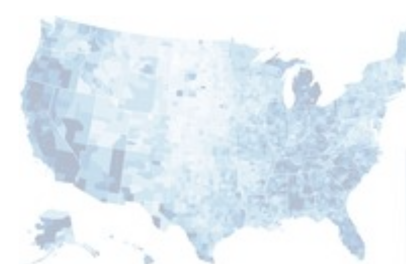
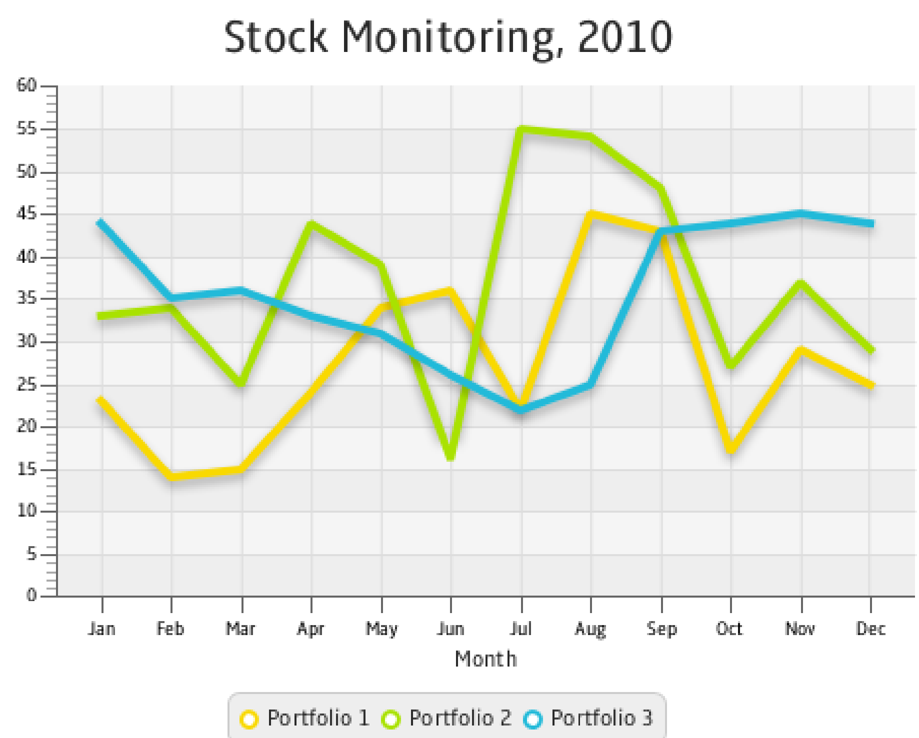
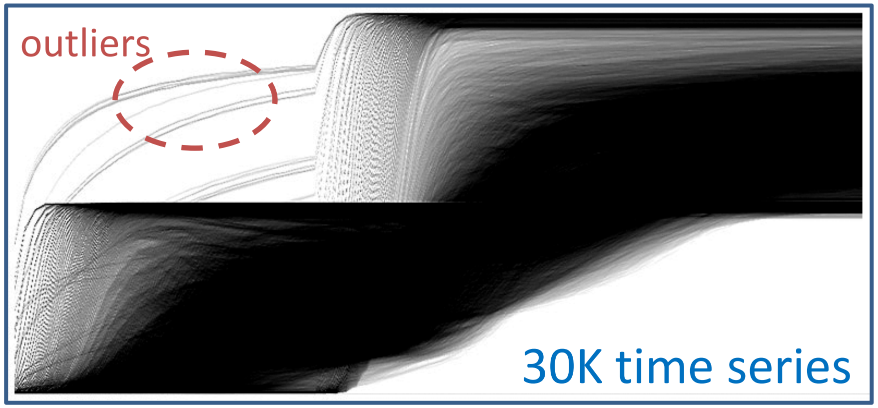


Diagram techniques

- Line graph
  - Quantitative data on common scale(s) wrt. time
  - Connection between points – trends, structures, groups





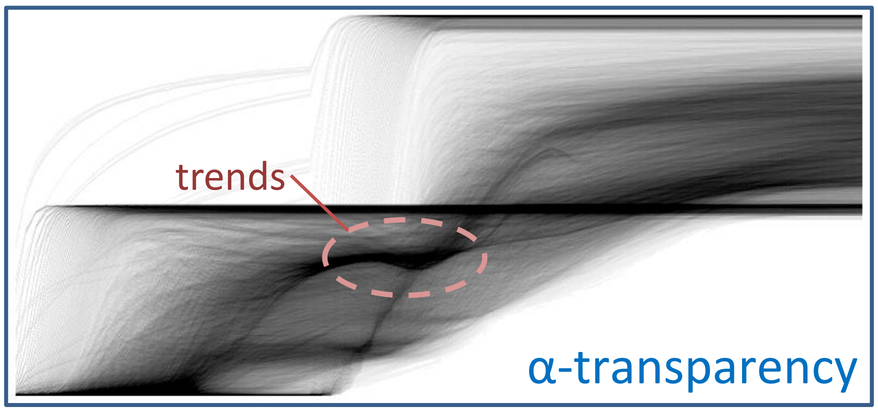
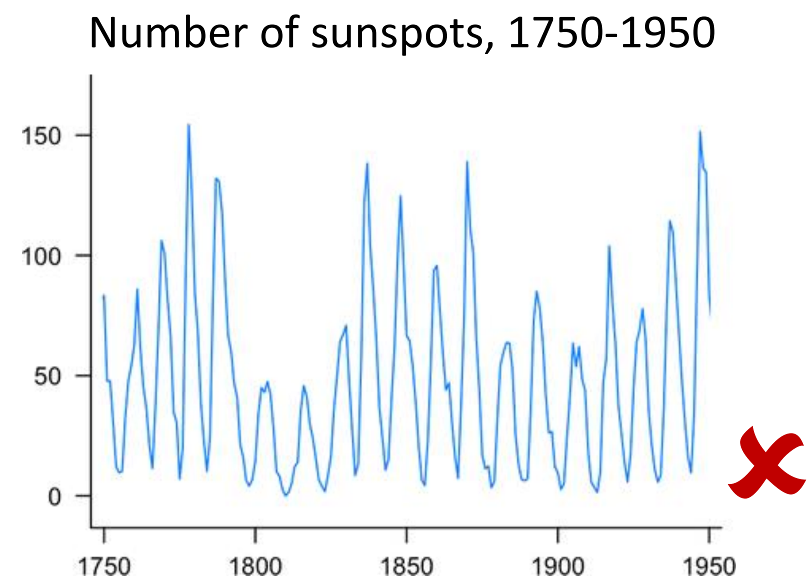
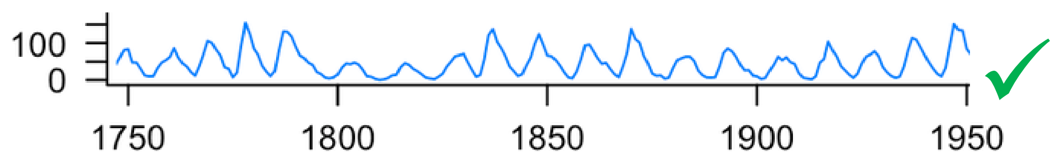


Diagram techniques

- Line graph
  - Banking to 45 degrees
    - Perceptual principle: most accurate angle judgment around 45°
    - Pick aspect ratio (height/width) accordingly





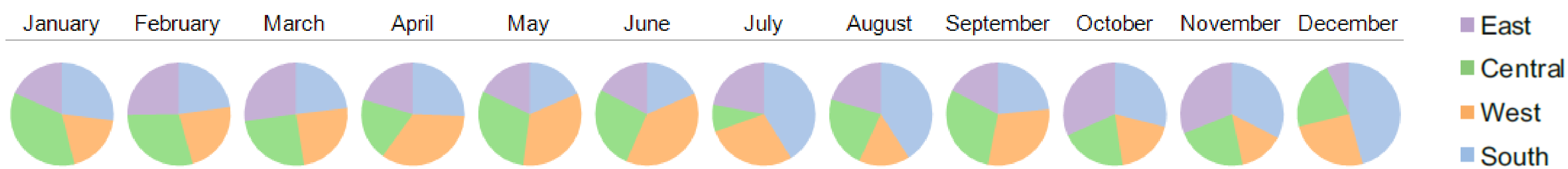
[Cleveland, Tufte]

Example: Analysis of Sales Data

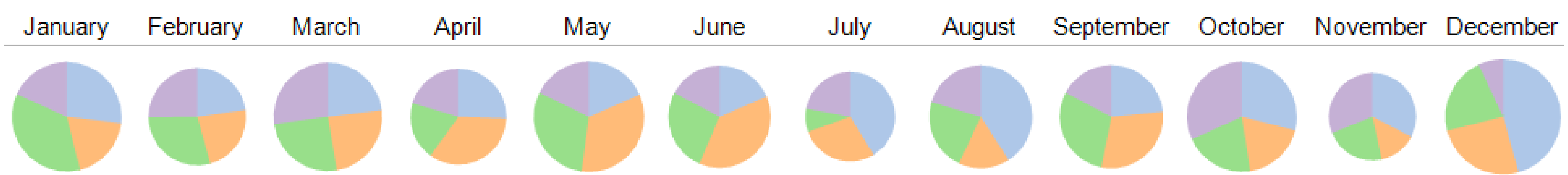
SIEMENS
Ingenuity for life

Region	January	Februa..	March	April	May	June	July	August	Septe..	October	Novem..	Decem..
East	64,162	70,172	93,657	55,056	63,631	53,040	51,974	63,272	54,110	112,284	69,697	27,437
Central	125,392	80,851	87,645	52,996	107,159	80,349	19,907	70,431	92,568	72,961	49,881	85,084
West	67,364	63,742	83,856	92,412	120,284	116,618	66,692	49,671	92,920	65,971	31,516	99,000
South	94,572	63,234	79,491	68,963	65,868	56,659	97,101	126,879	73,240	102,589	73,044	177,943
Total ($)	351,490	277,999	344,649	269,427	356,942	306,666	235,674	310,253	312,838	353,805	224,138	389,464

Pie chart: compare sales in different regions, but total sales are not shown



Pie chart: Total sales → Size



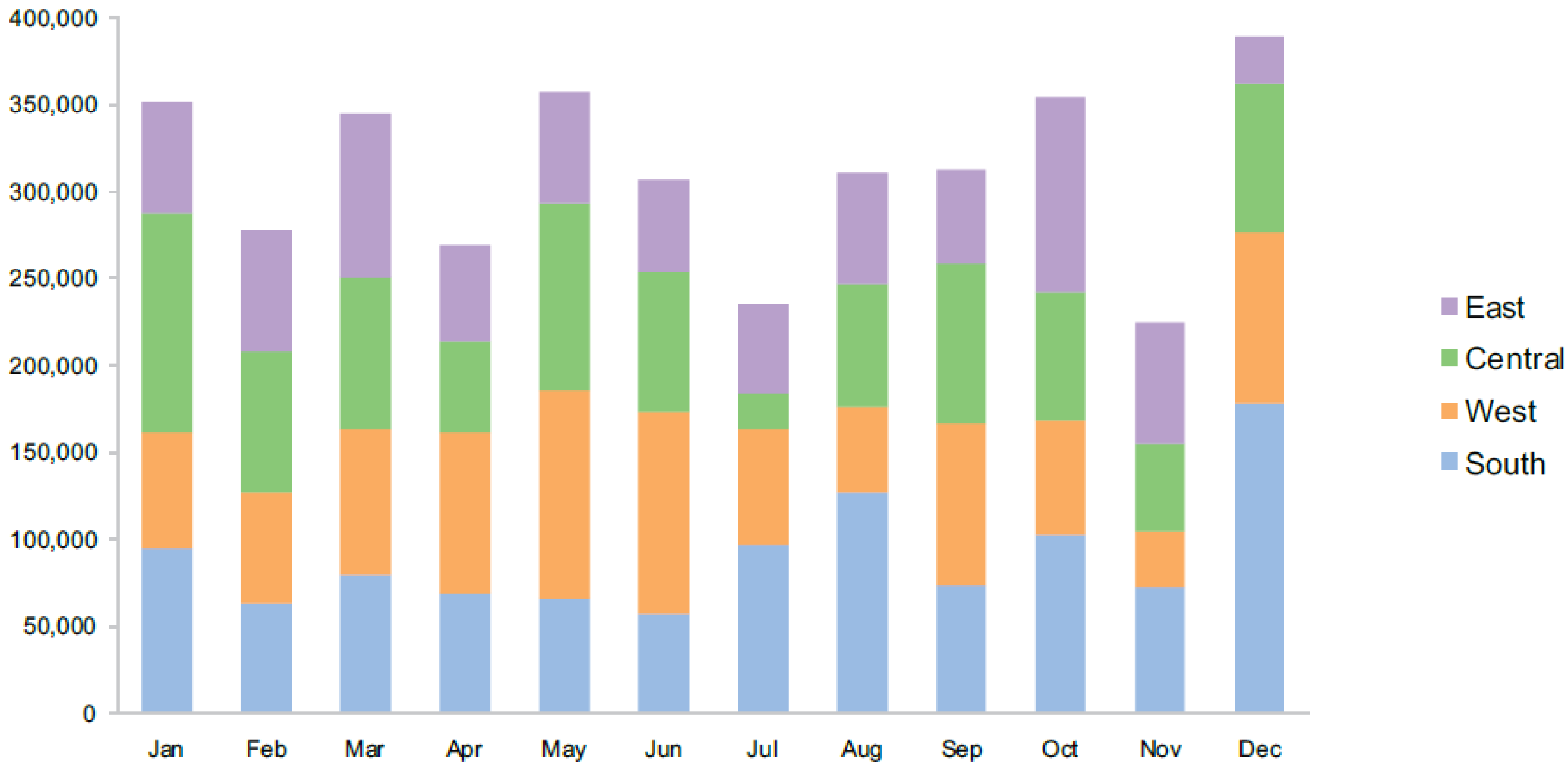
Hard to compare angles and/or sizes

[Few 2011]

Example: Analysis of Sales Data

SIEMENS
Ingenuity for life

Stacked bar chart



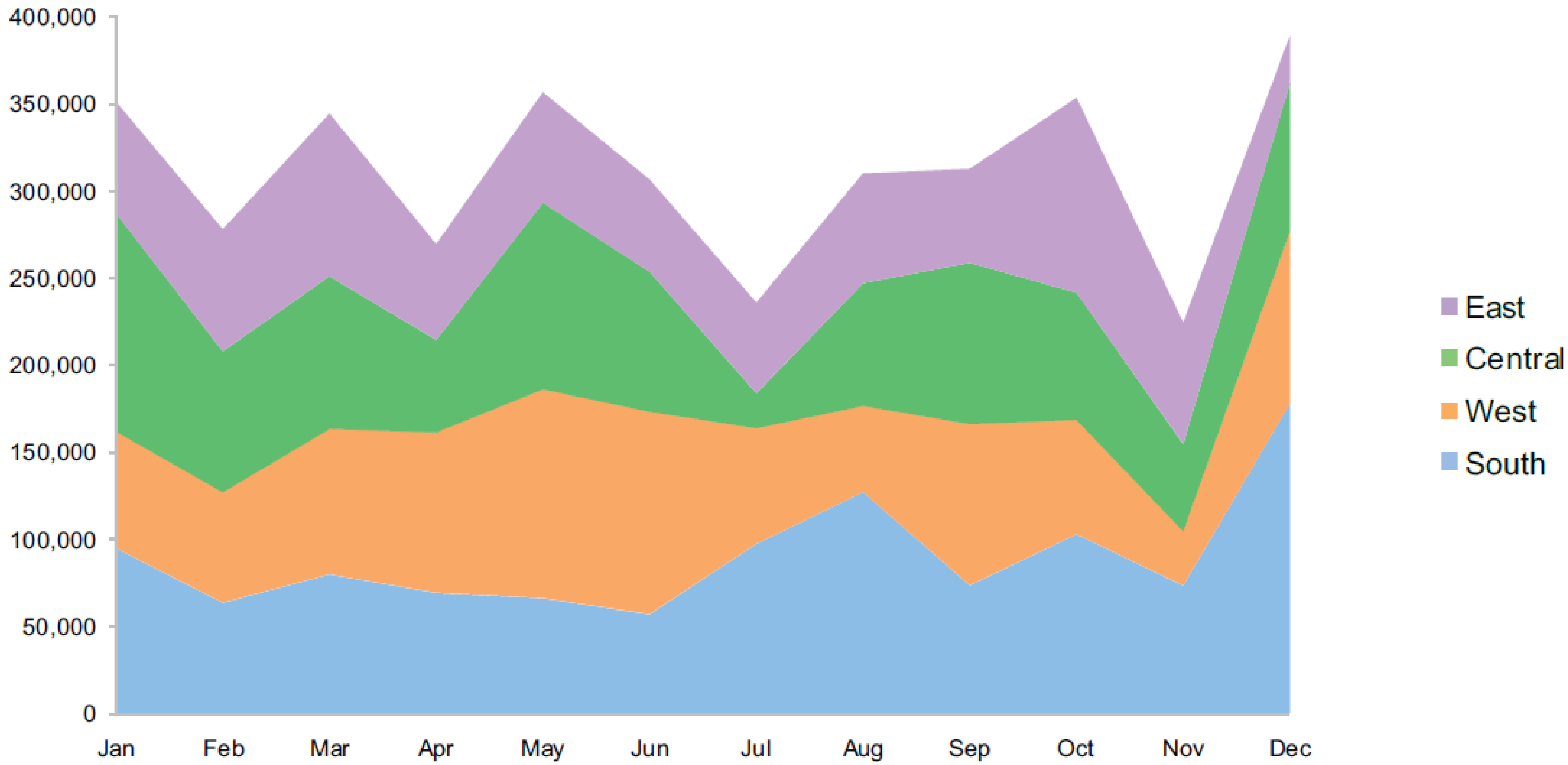
Difficult to see trends over time

[Few 2011]

Example: Analysis of Sales Data

SIEMENS
Ingenuity for life

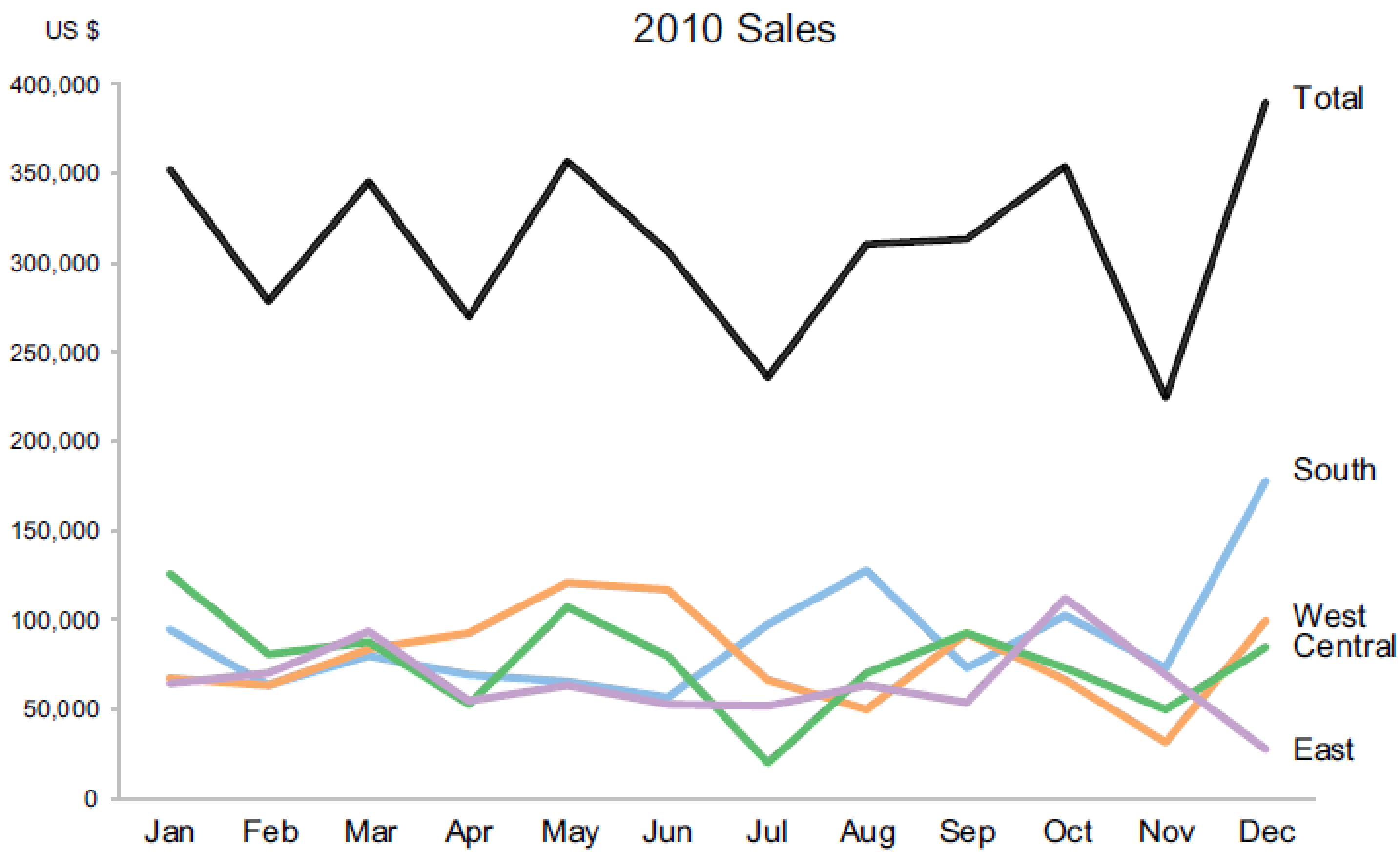
Stacked area chart



Exercise: Sketch development of orange curve

[Few 2011]

Example: Analysis of Sales Data

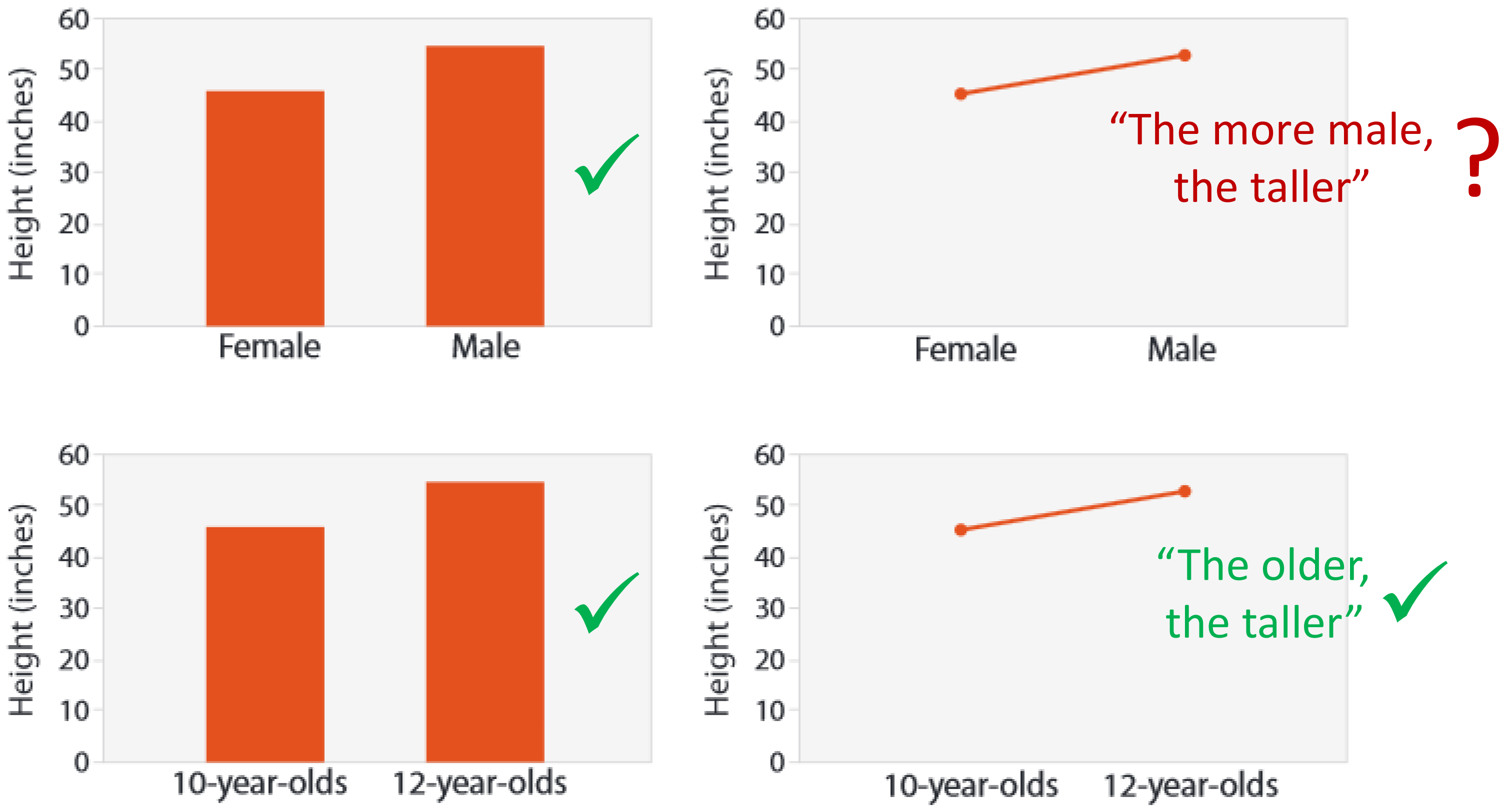


[Few 2011]

Diagram techniques

SIEMENS
Ingenuity for life

- Bar vs. line chart
  - Bars support comparison
  - Lines imply trends

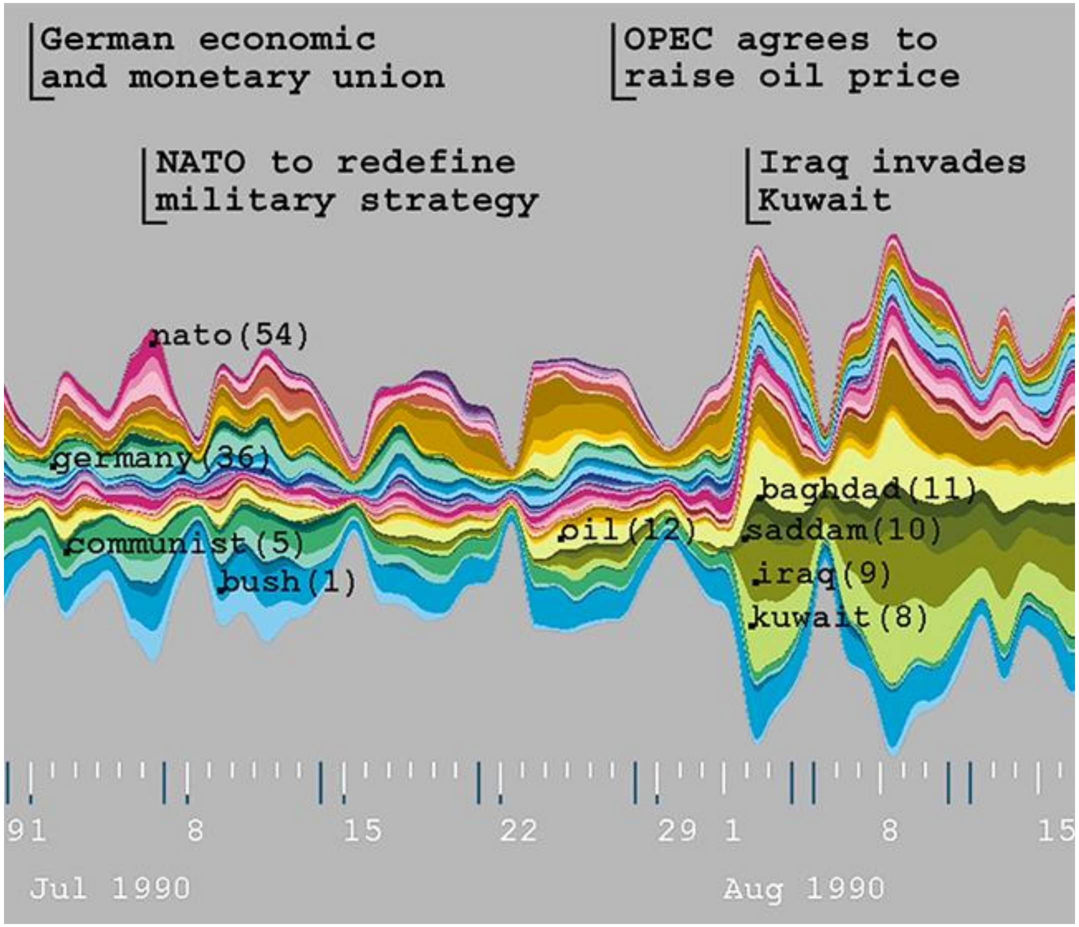


[Munzner 14]

Diagram techniques

SIEMENS
Ingenuity for life

- ThemeRiver
  - Thematic changes in documents
  - Occurrence per topic/category mapped to width of river band
  - Less distorted around center
  - Rearranging bands

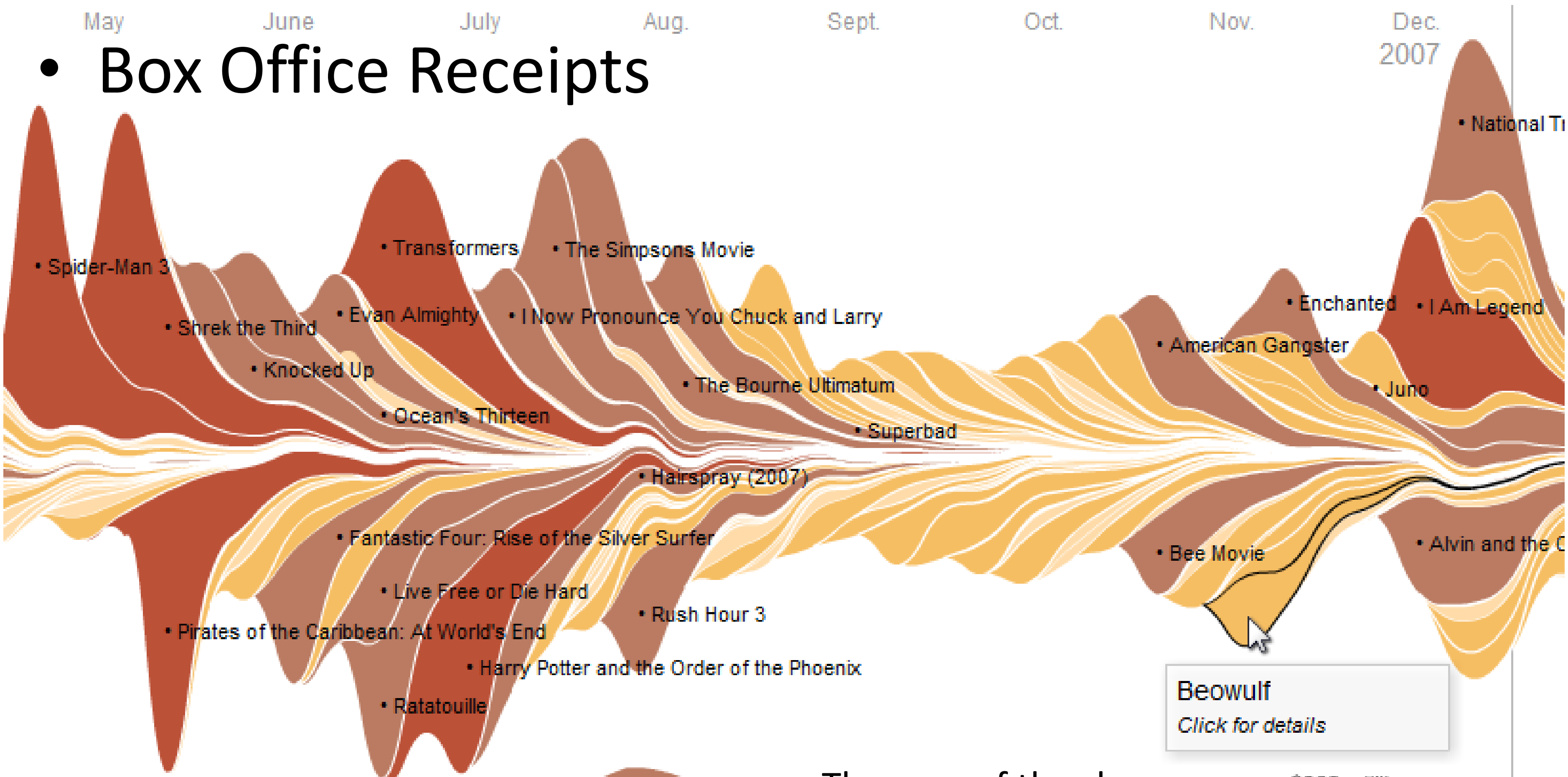


[Havre et al. 01]

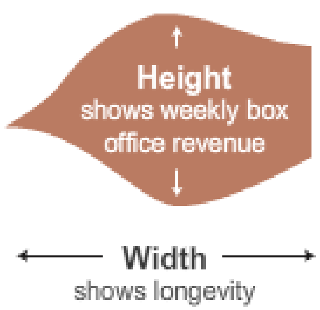
Diagram techniques

SIEMENS
Ingenuity for life

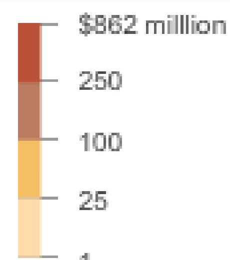
- Box Office Receipts



Each shape shows how one film did at the box office



The **area** of the shape (and its **color**) correspond to the film's total domestic gross, through Feb. 21.

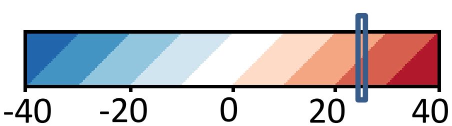


www.nytimes.com/interactive/2008/02/23/movies/20080223_REVENUE_GRAPHIC.html

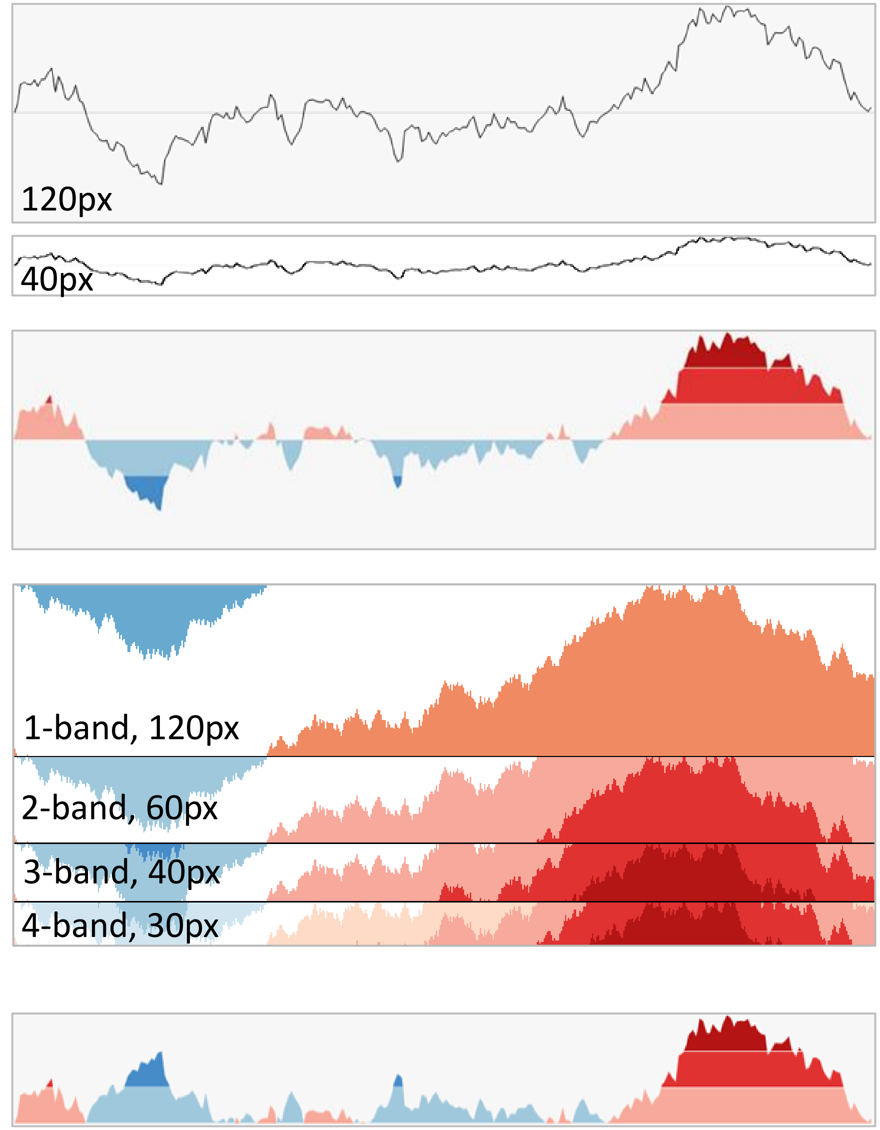
Diagram techniques

SIEMENS
Ingenuity for life

- Horizon graph
  - Reduces vertical space without losing precision
  - Split vertically into layered bands
  - Collapse color bands to show values in less vertical space



  - Optional mirroring of negative values

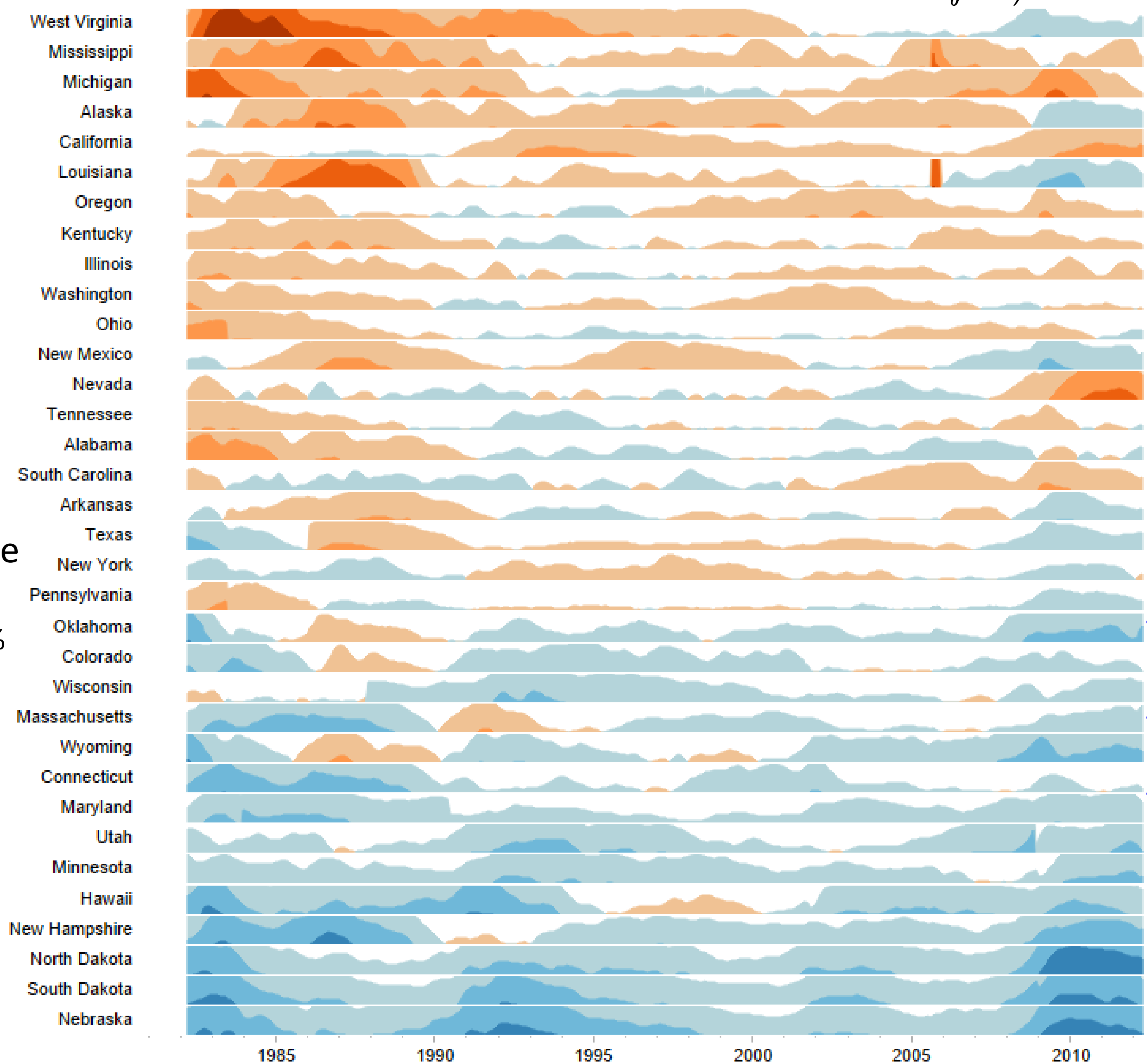


http://square.github.io/cubism/ [Saito 05, Heer 09]

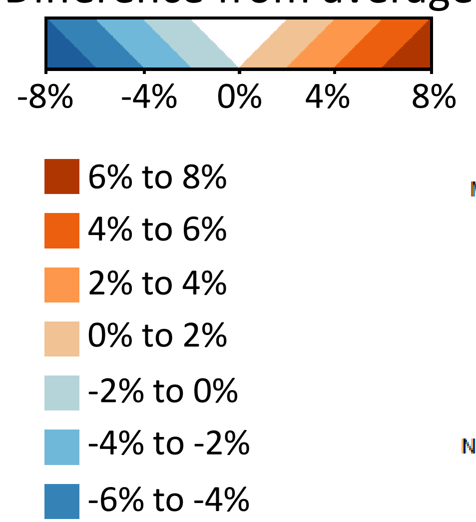
Diagram techniques

SIEMENS
Ingenuity for life

Unemployment rate (1982-2012)



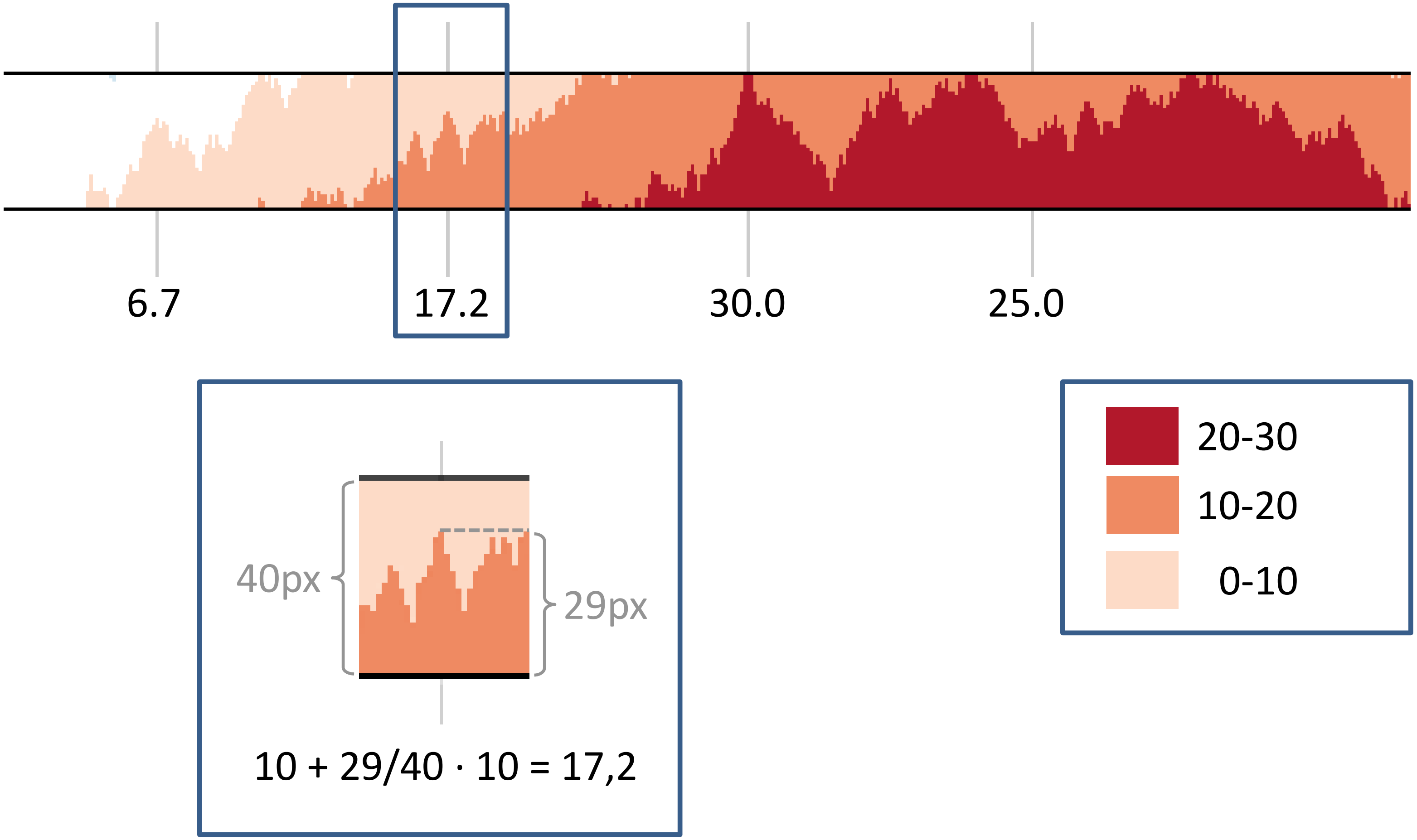
Difference from average



www.tableau.com/learn/gallery/unemployment-horizon-chart

Diagram techniques

- Example



Contact information

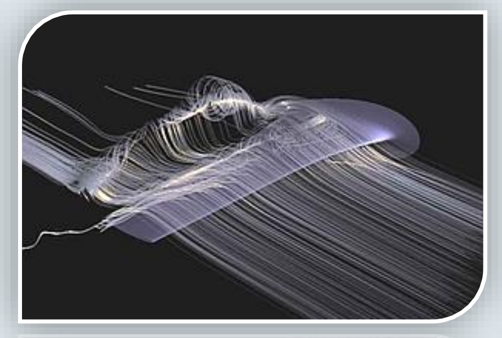
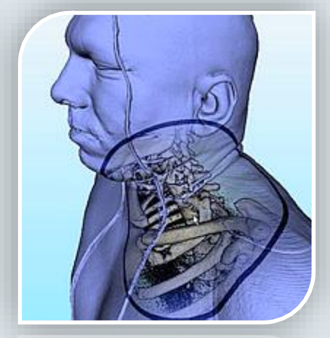
Dr. Johannes Kehrer

Siemens Technology
T RDA BAM IBI-DE
Otto-Hahn-Ring 6
81739 München, Deutschland

E-mail:
kehrer.johannes@siemens.com

Internet
siemens.com/innovation

Intranet
intranet.ct.siemens.com


Visual Data Analysis

Visualization Mapping II

Dr. Johannes Kehrer

Disclaimer

These lecture slides and materials are for personal use by students and are intended for educational purposes only. Distribution or reproduction of the material is prohibited.

The views and opinions expressed by any individual during the course of these lecturers are their own and do not necessarily represent the views, opinions, or positions of Siemens or the Technical University of Munich.

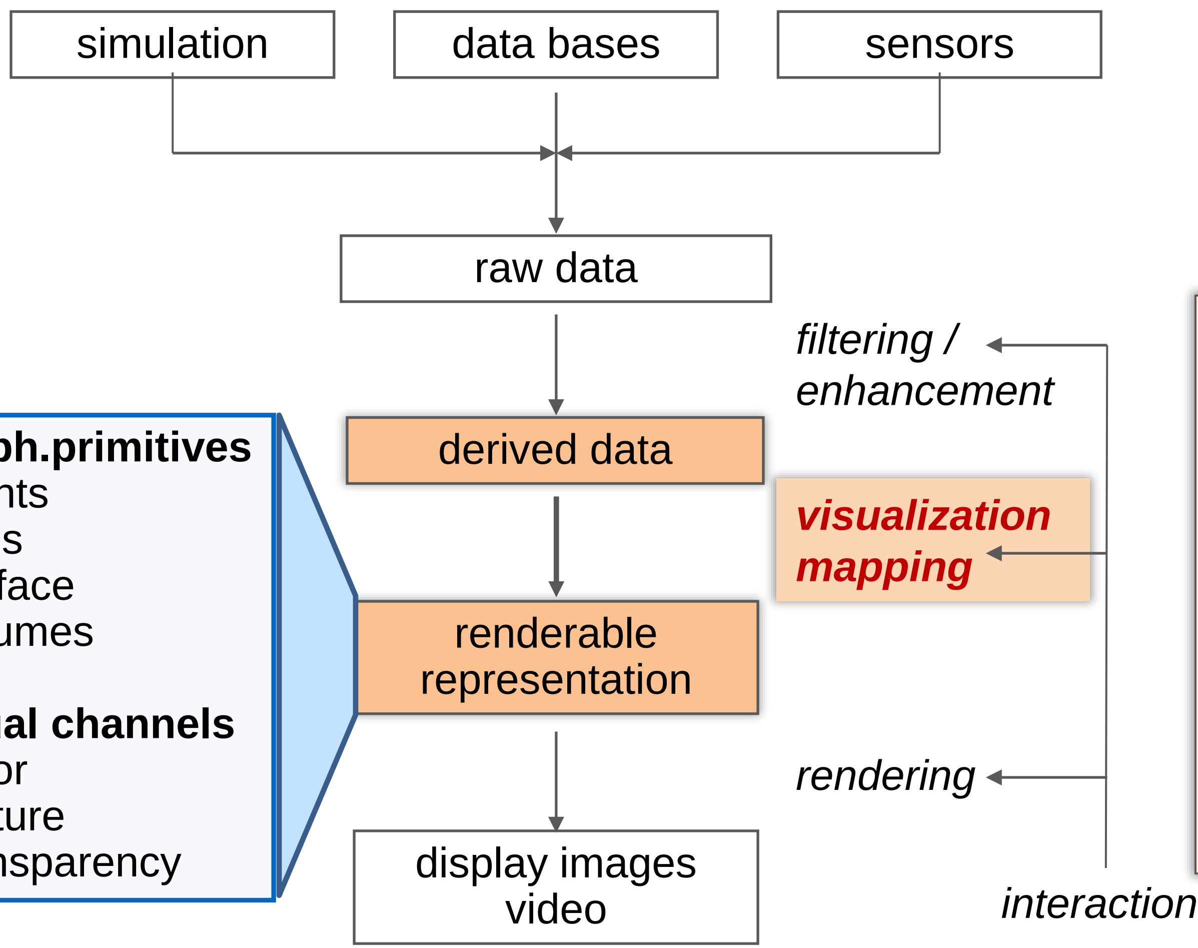
Although reasonable efforts have been made to ensure the information in these lecture slides is current and accurate, we do not assume any responsibility for the content of the posted material.

These slides also contain links to websites of third parties. As the content of these websites is not under our control, we cannot assume any liability for such external content.

Mapping techniques

SIEMENS
Ingenuity for life

- From derived data to a renderable representation



Graph.primitives
- points
- lines
- surface
- volumes

Visual channels
- color
- texture
- transparency

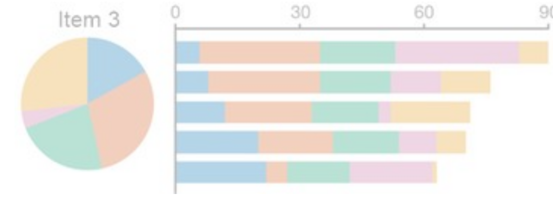
Derived data
→ graphical primitives
- scalar field → isosurface
- 2D field → height field
- vector field → vectors
- tensor field → glyphs
- 3D field → volume (C,α)

Decide
- Which parts of the data are shown?
- How to represent them?

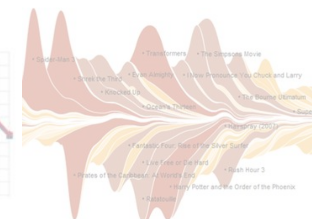
Diagram techniques

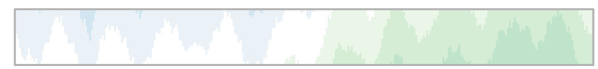
SIEMENS
Ingenuity for life

- Categorical + quantitative data
  - Bar/pie chart, stacked bars

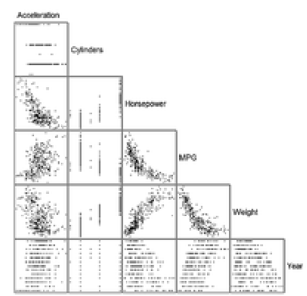


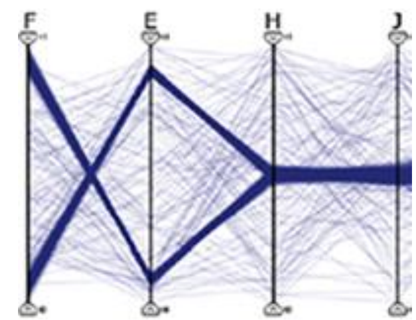
- Time-dependent data
  - Line graph, ThemeRiver, Horizon graph



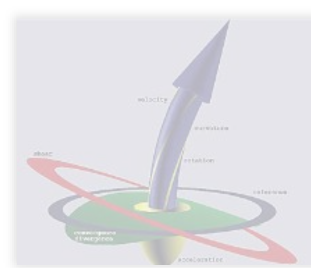


- Single and multiple variables
  - Histogram, scatterplot, parallel coordinates





  - Glyphs, color mapping



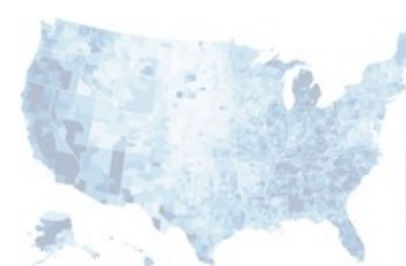


Diagram techniques

SIEMENS
Ingenuity for life

- Histogram (graphical display of statistics)
  - Special kind of bar chart
  - Show frequency of occurrence of values for 1 variable

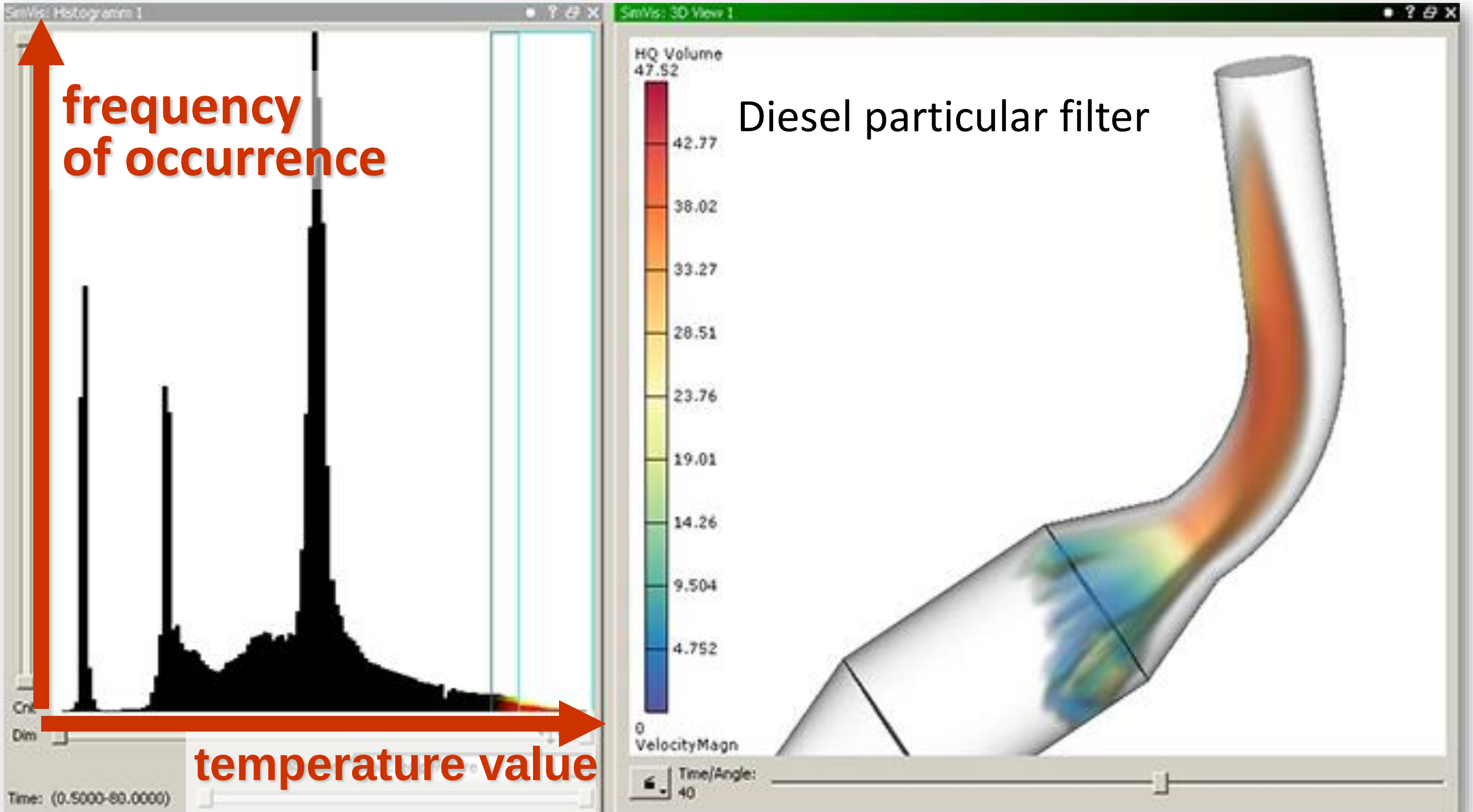
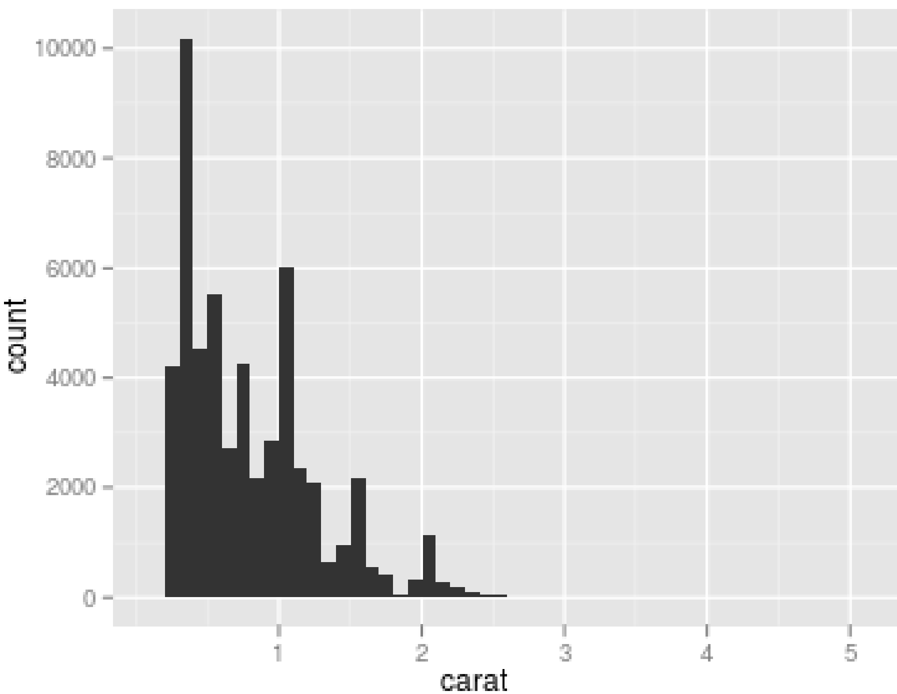


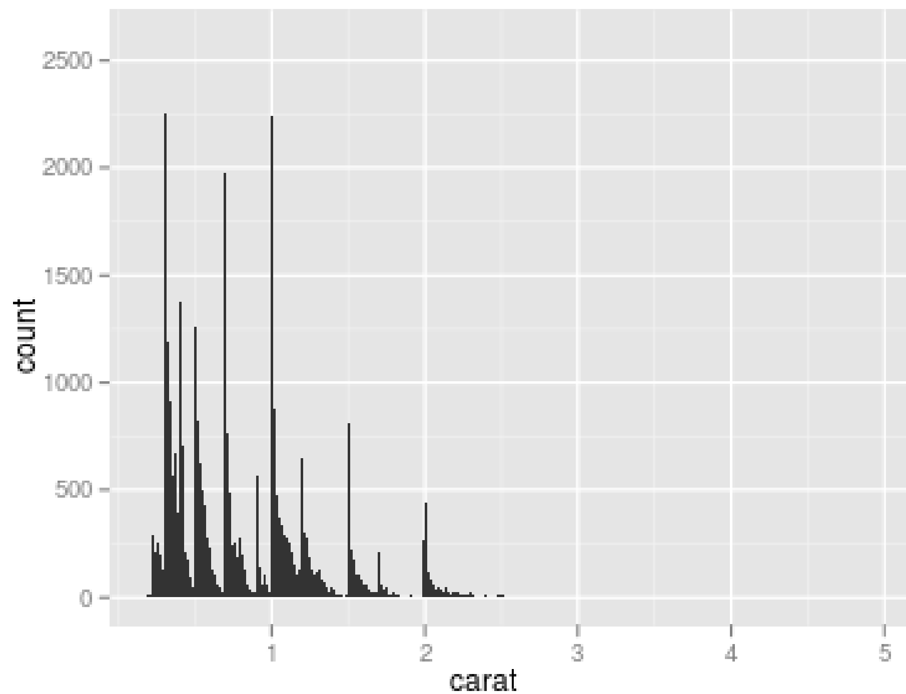
Diagram techniques

SIEMENS
Ingenuity for life

- Histogram
  - Binning: group values into equally spaced intervals (bins)
  - Bin width affects representation



bin width = 0.1

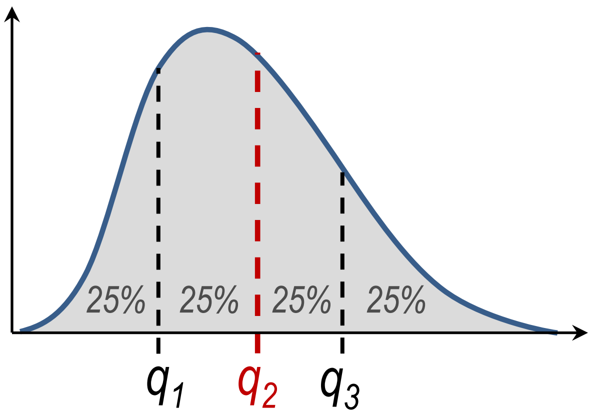


bin width = 0.01

Diagram techniques

SIEMENS
Ingenuity for life

- Box plot variations
  - Shows summary statistics of a distribution (1 variable)

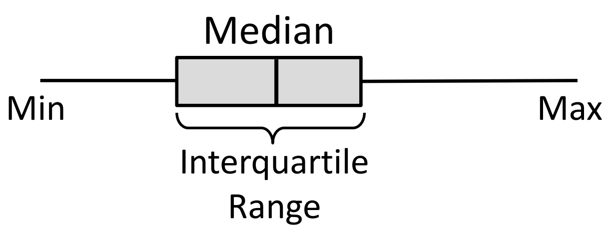


Probability density function (PDF)

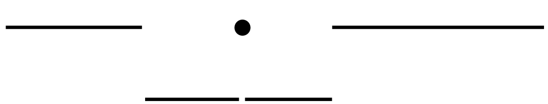
q_1 … lower quartile

q_2 … median

q_3 … upper quartile



Tukey's box plot

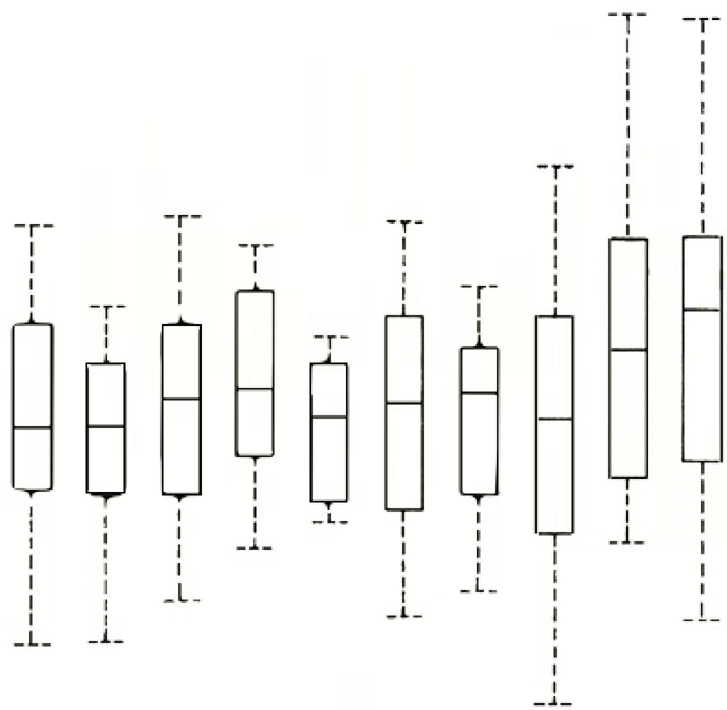


Tufte's quartile plot variations

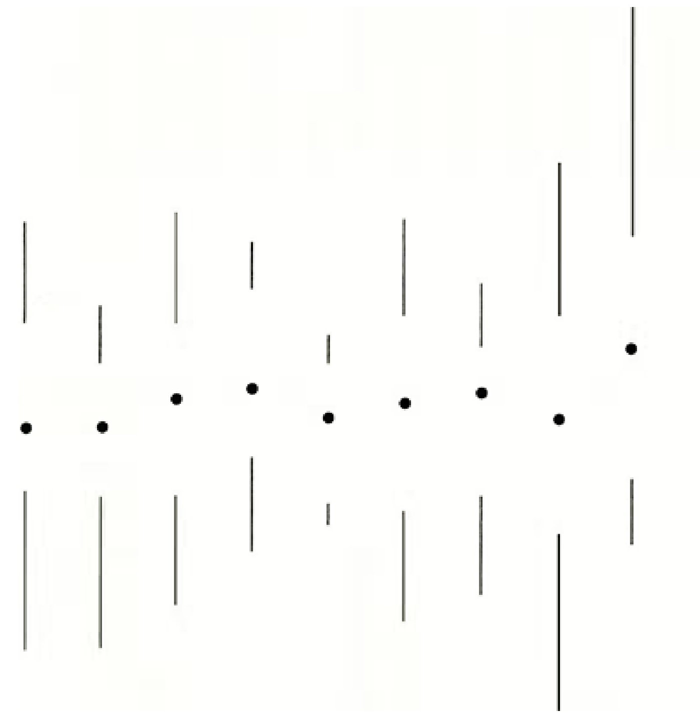
Diagram techniques

SIEMENS
Ingenuity for life

- Box plot variations



Tukey's box plot



Tufte's quartile plot

Diagram techniques

SIEMENS
Ingenuity for life

- Scatterplots
  - Show correlations between 2 dependent variables
  - Typically quantitative (measurable) data attrib.
  - Find trends, outliers, distributions, correlations, clusters, …

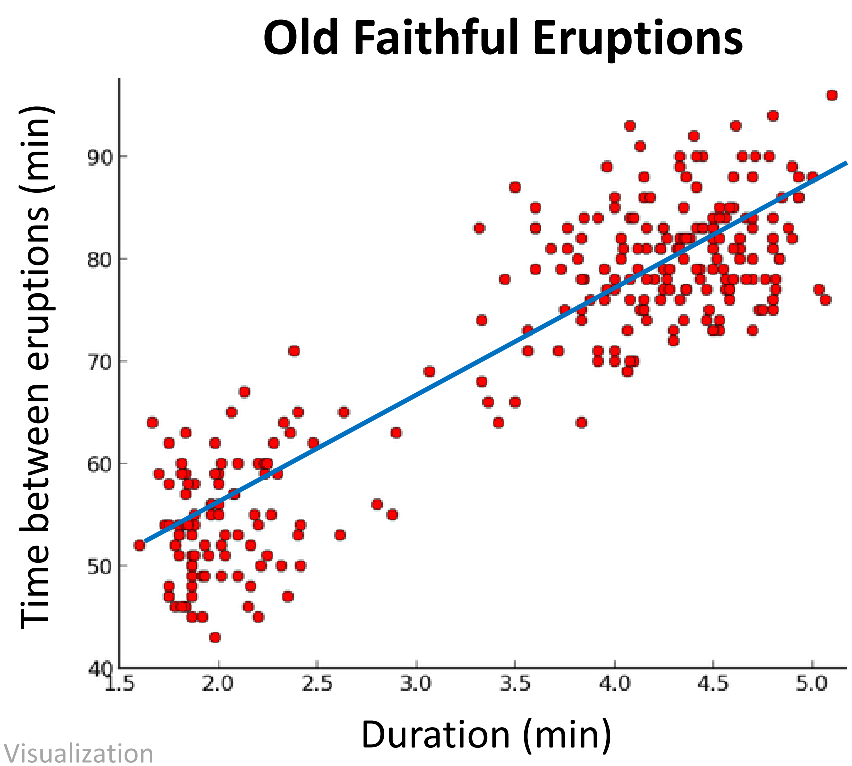
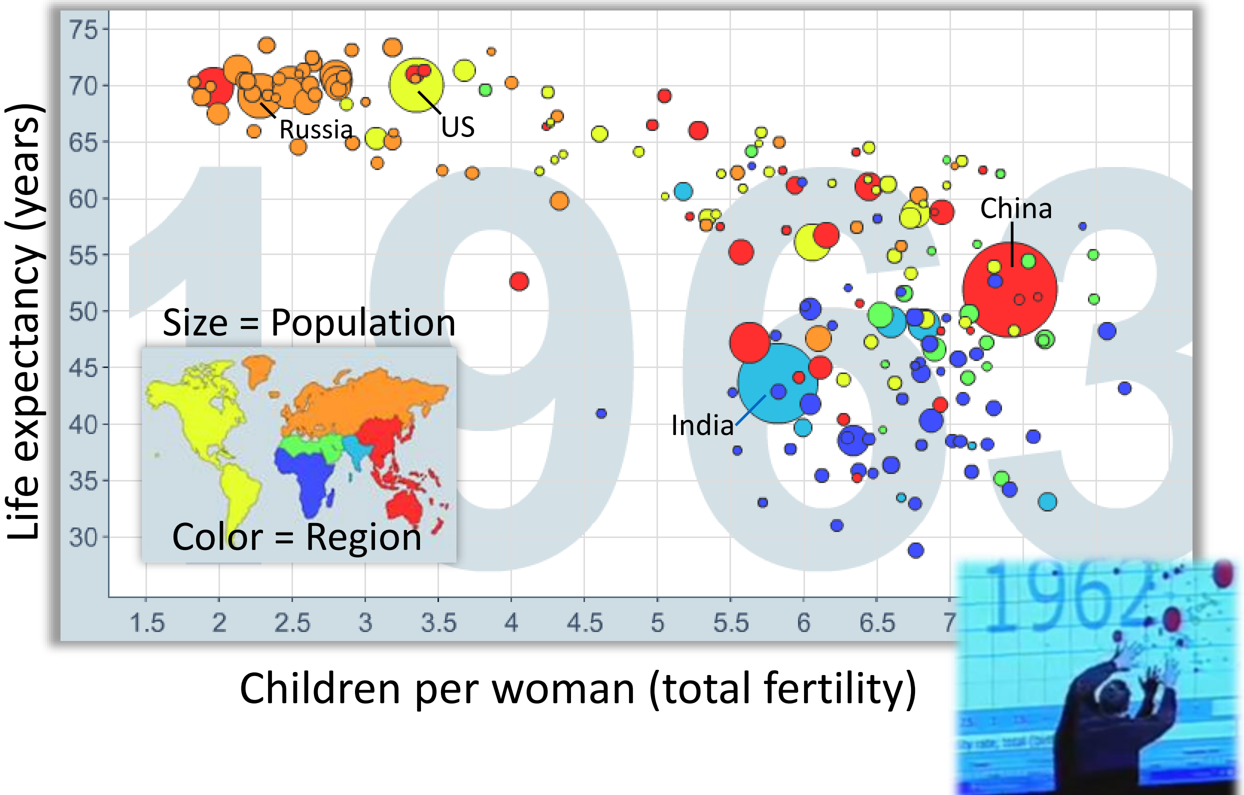
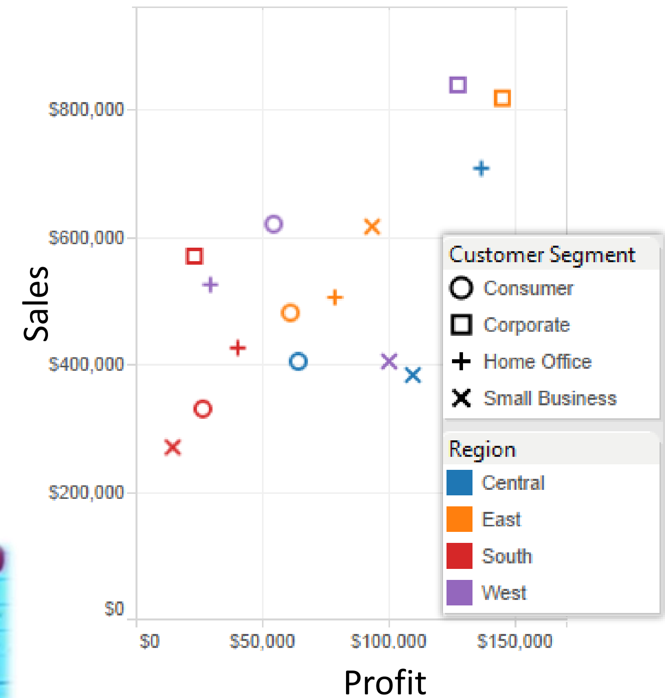


Diagram techniques

- Scatterplot variations
  - Encode additional attributes by size, color, shape, …





www.gapminder.org

www.tableau.com

Diagram techniques

- Scatterplot matrix
  - Show (all possible) combinations of attributes in a scatterplot matrix
  - Each row/column is one attribute
  - Overview of correlation and patterns between data attributes

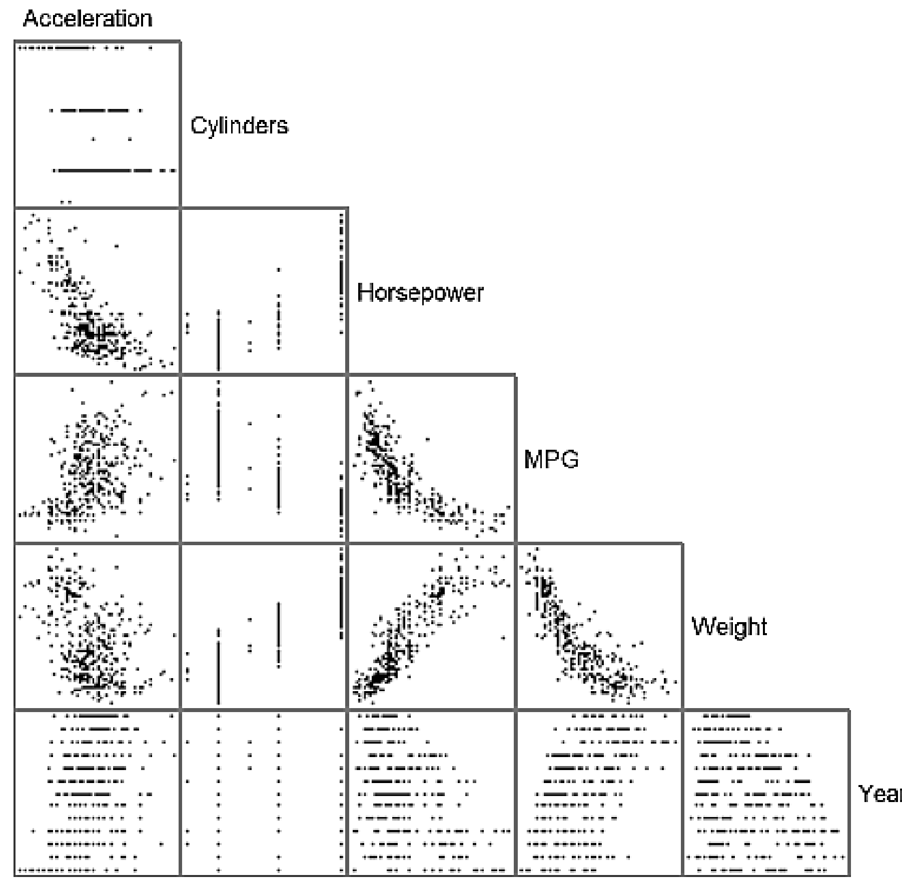
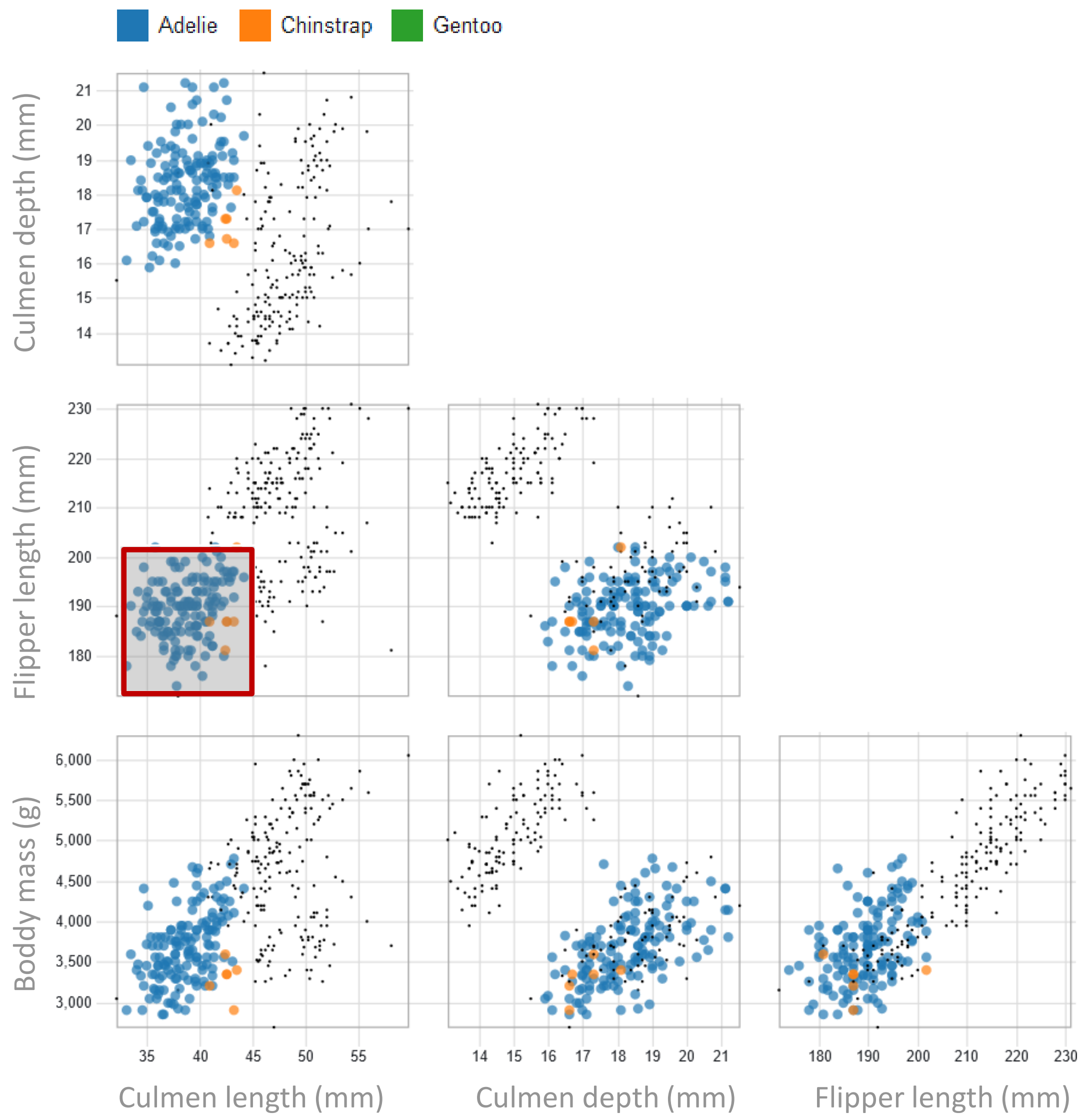


Diagram techniques

- Scatterplot matrix
  – **Brushing:** mark data subset
  – **Linking:** highlight brushed data in linked views
  – Move/alter/extend brush

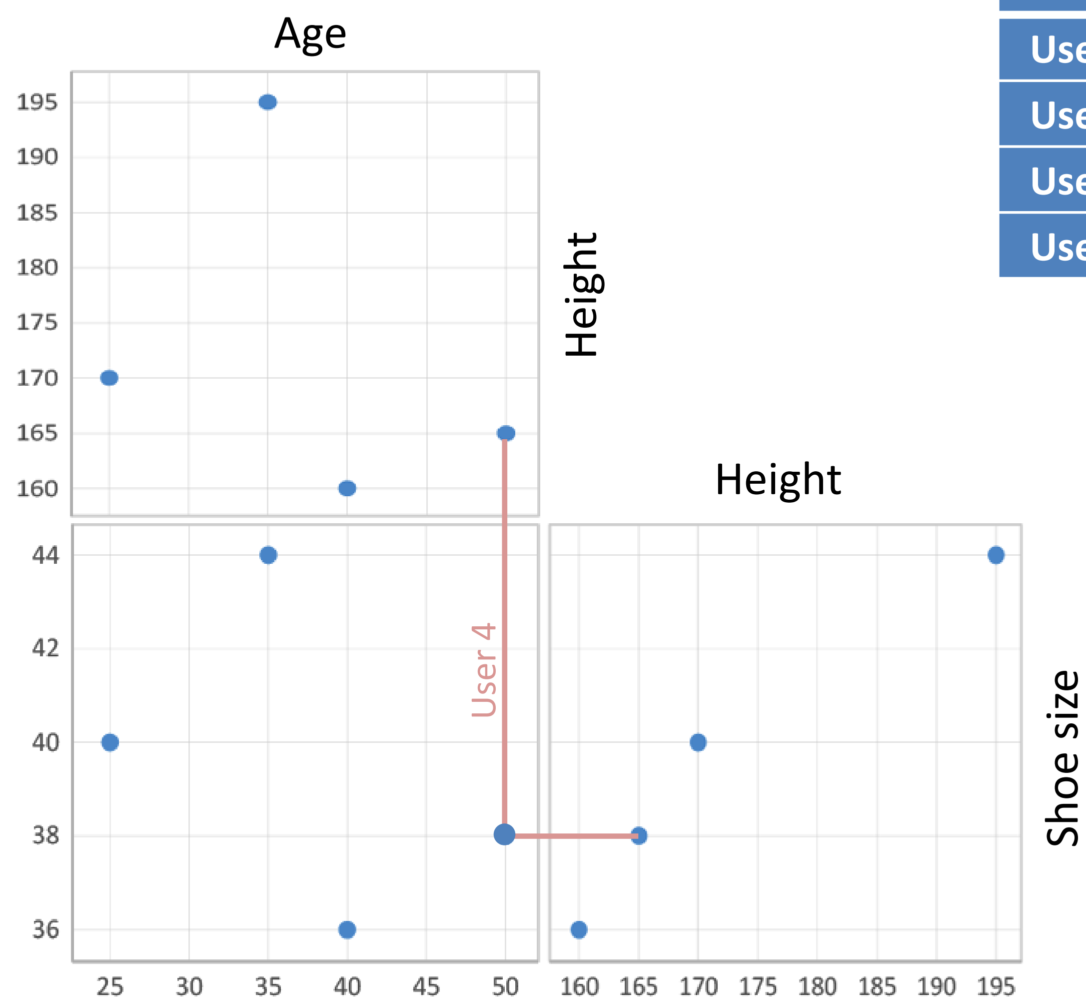


https://observablehq.com/@d3/brushable-scatterplot-matrix

Diagram techniques

SIEMENS
Ingenuity for life

- Scatterplot matrix



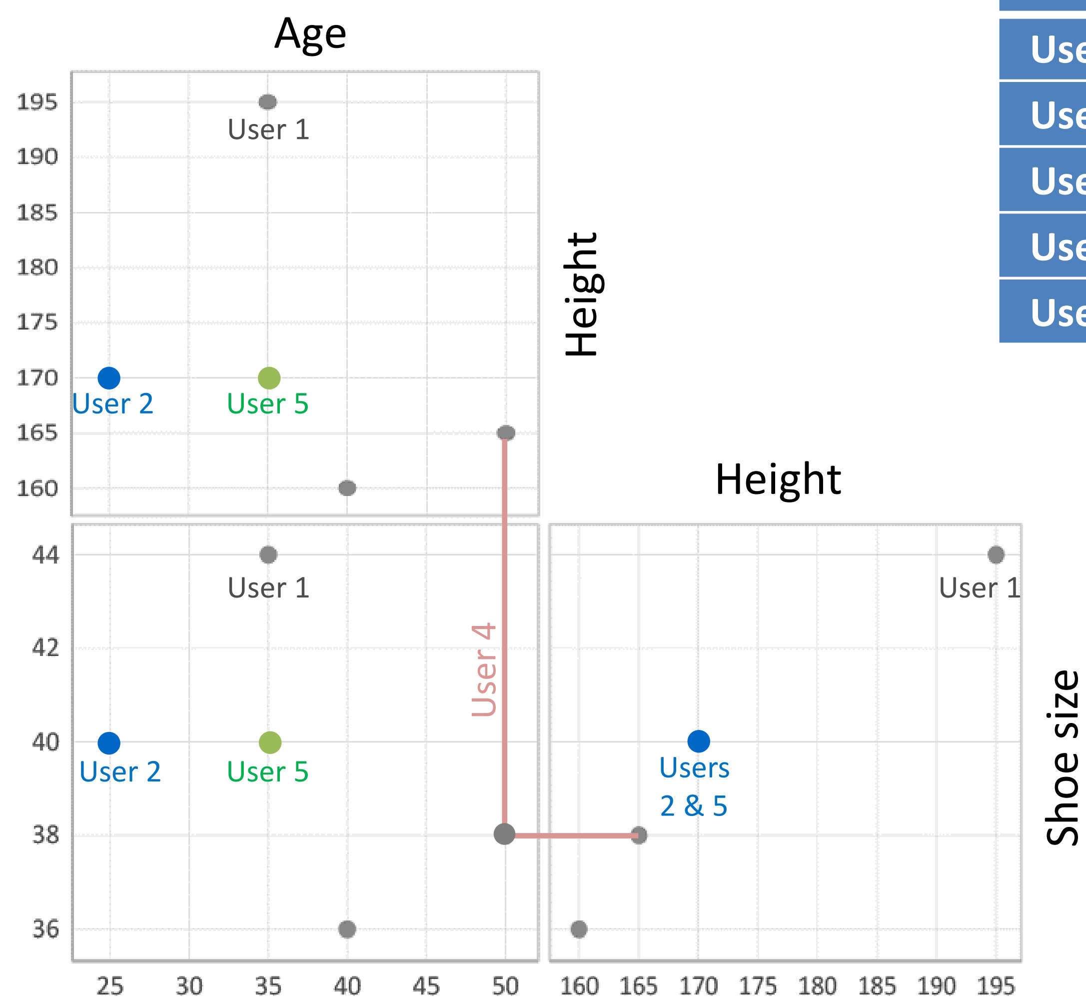
	Age	Height	Shoe size
User 1	35	$\boxed{195}$	$\boxed{44}$
User 2	$\boxed{25}$	170	40
User 3	40	$\boxed{160}$	$\boxed{36}$
User 4	$\boxed{50}$	165	38

Normalize to min/max

https://observablehq.com/@d3/brushable-scatterplot-matrix

Diagram techniques

- Scatterplot matrix



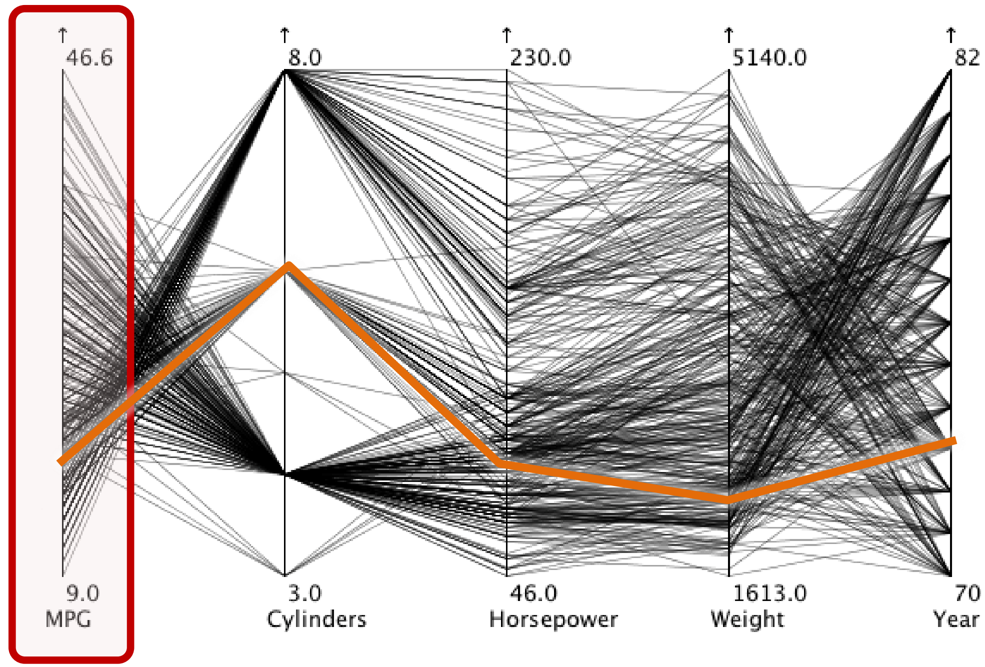
	Age	Height	Shoe size
User 1	35	195	44
User 2	25	**170**	**40**
User 3	40	160	36
User 4	50	165	38
User 5	35	**170**	**40**

Normalize to min/max

https://observablehq.com/@d3/brushable-scatterplot-matrix

Diagram techniques

- Parallel coordinates
  - Represent multiple data variables
  - Each variable is represented by a vertical axis, which are organized as evenly spaced parallel lines
  - Data on each axis is normalized to min/max
  - One data sample is represented by a connected set of points, one on each axis

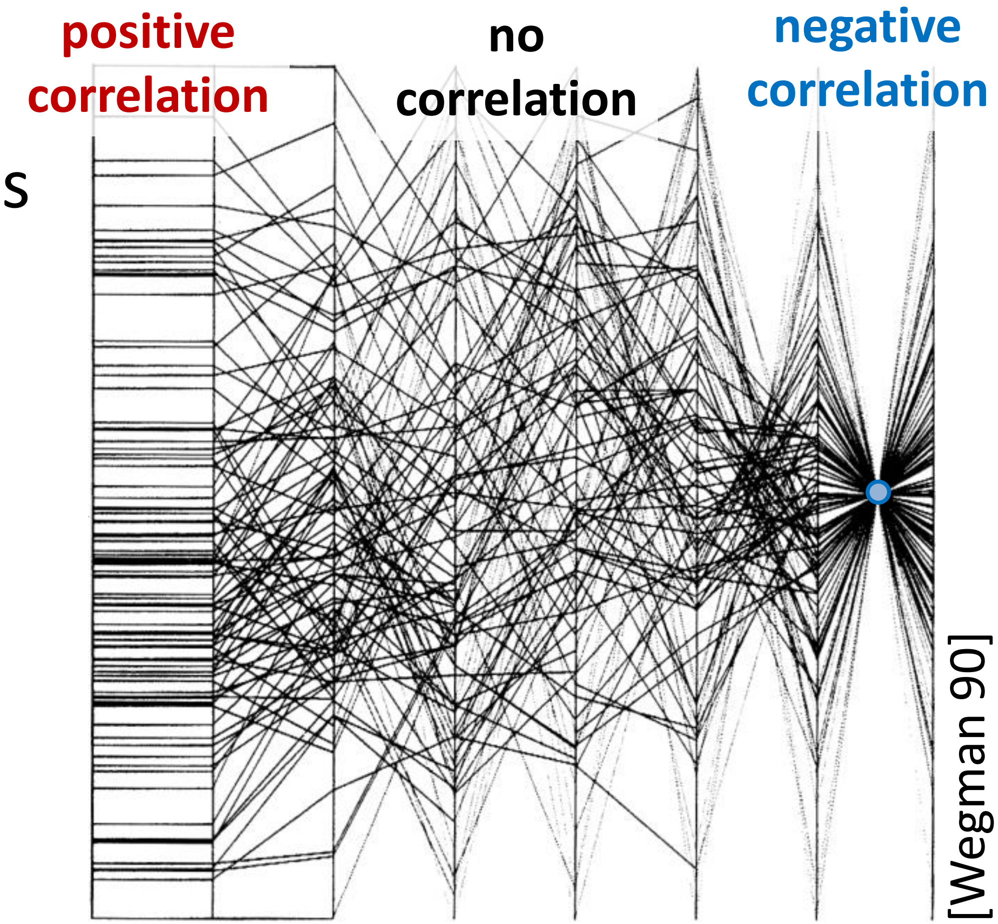


Attribute / Dimension

Diagram techniques

SIEMENS
Ingenuity for life

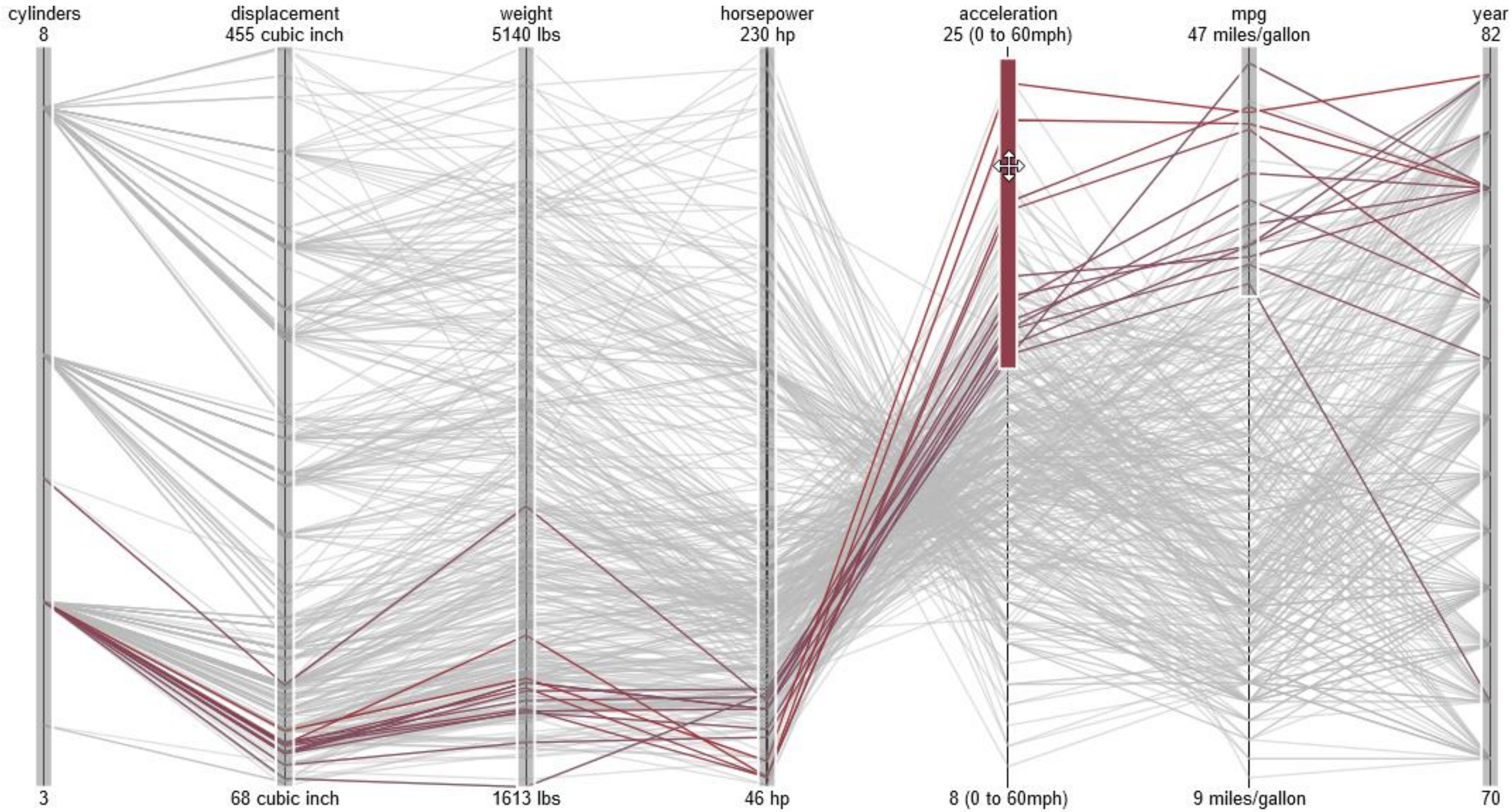
- Parallel coordinates
  - Recognize patterns between adjacent axes
  - Steep learning curve for novices
  - Brushing (mark interesting data subset)



Parallel coordinates illustrating correlation of ρ = *1, 0.8, 0.2, 0, -0.2, -0.8, -1*

Parallel Coordinates of Automobile Data

SIEMENS
Ingenuity for life



A database of cars is plotted in seven coordinate dimensions; each path represents one car. Drag and resize the coordinate selection sliders to filter the cars in any dimension.

Source: GGobi

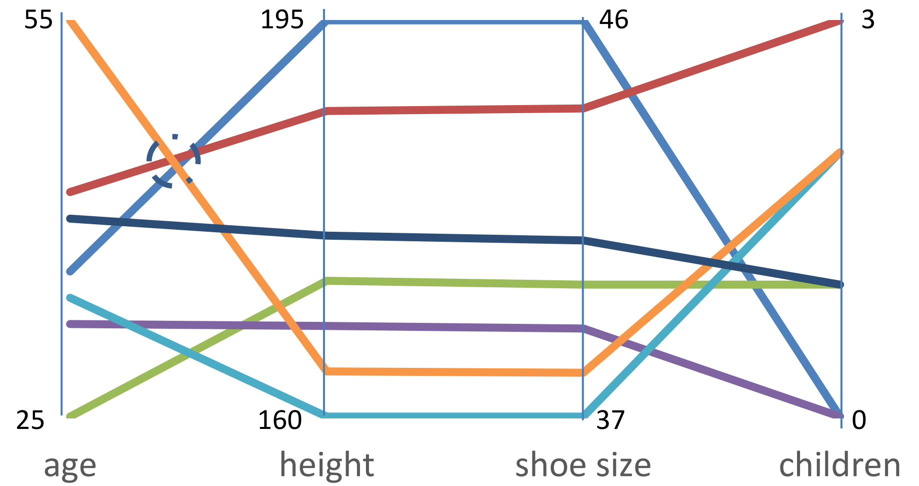
https://homes.cs.washington.edu/~jheer/files/zoo/ex/stats/parallel.html

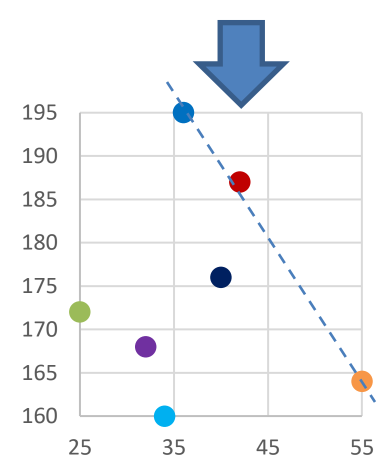
Diagram techniques

- Example: Parallel coordinates

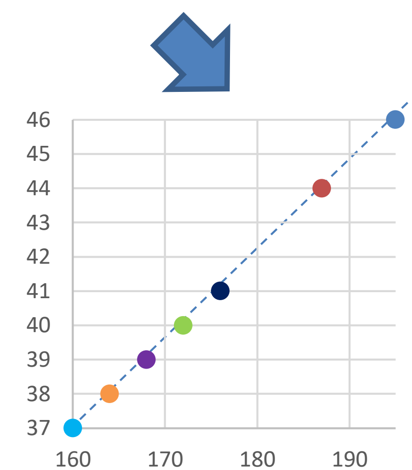
	Age	Height	Shoe size	No. of children
User 1	36	195	46	0
User 2	42	187	44	3
User 3	25	172	40	1
User 4	32	168	39	0
User 5	34	160	37	2
User 6	55	164	38	2
User 7	40	176	41	1

Normalize each attribute to min/max





age vs. height



height vs. shoe size

Diagram techniques

- Parallel coordinates
  - Line point duality
    - Points in scatterplot map to lines in parallel coordinates
    - Points in parallel coordinates map to lines in scatterplot

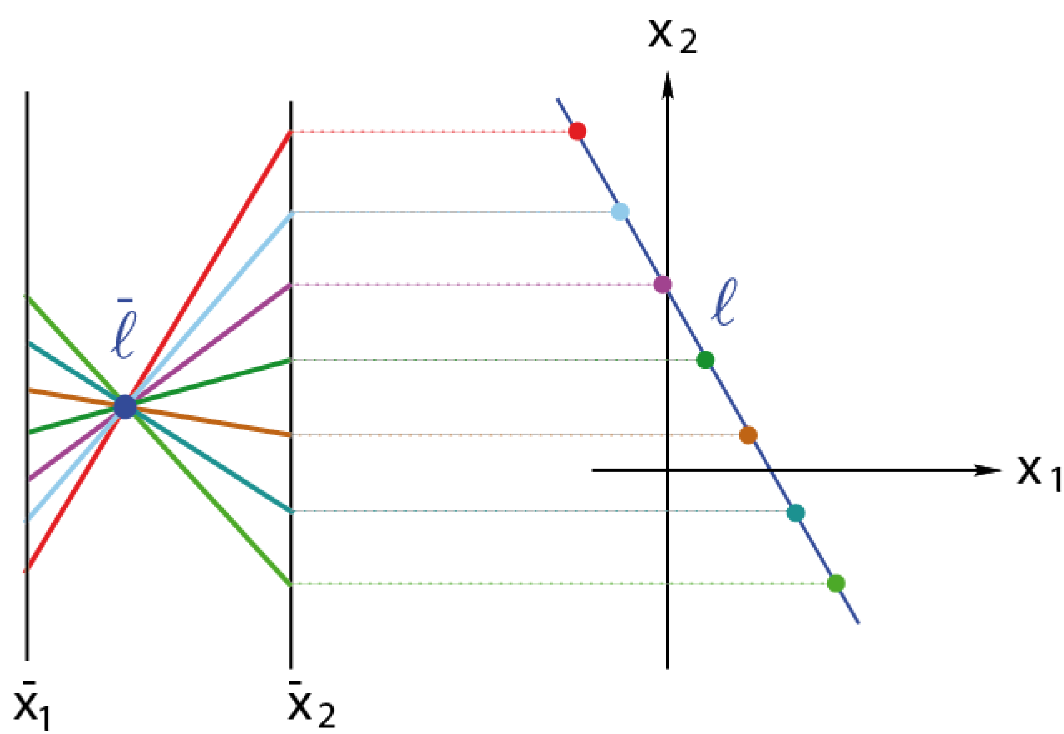


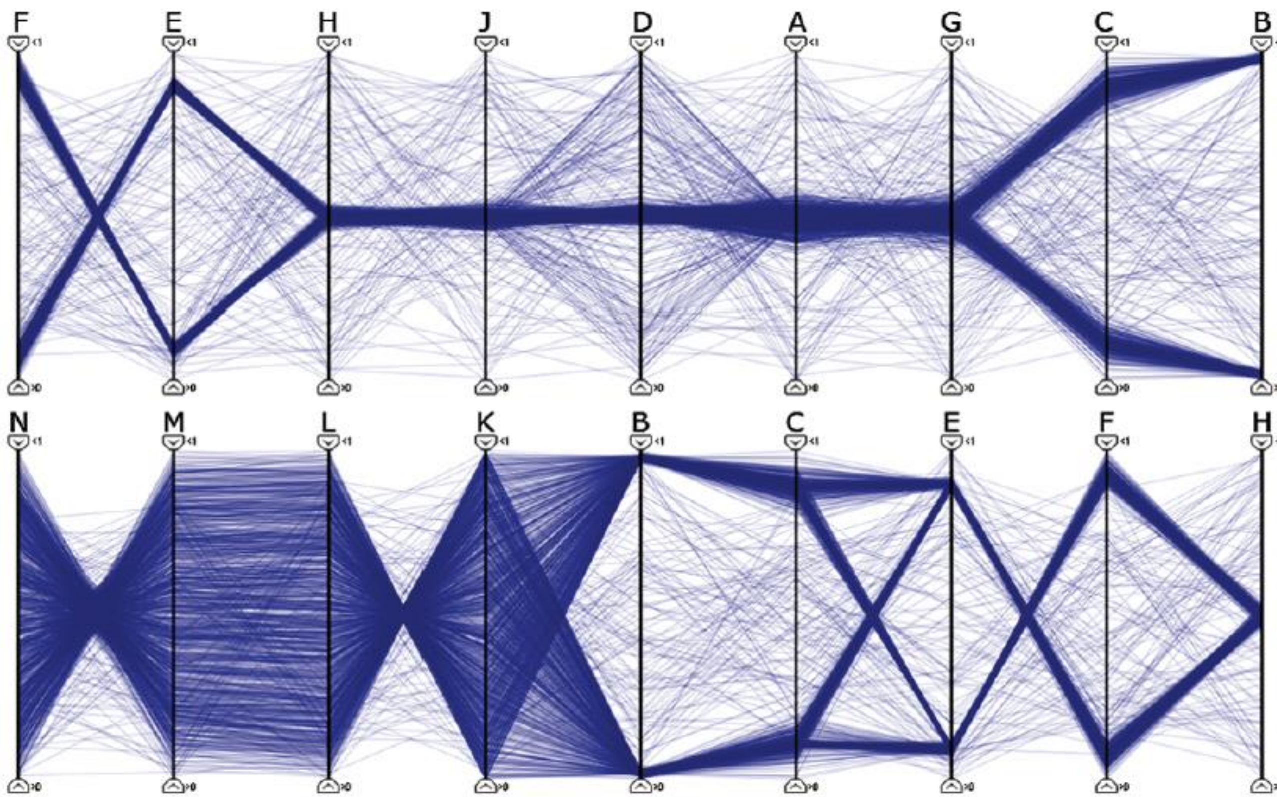
Diagram techniques

SIEMENS
Ingenuity for life

- Parallel coordinates
  - Axis ordering is a major challenge
  - Order by quality metrics

Enhance clustering

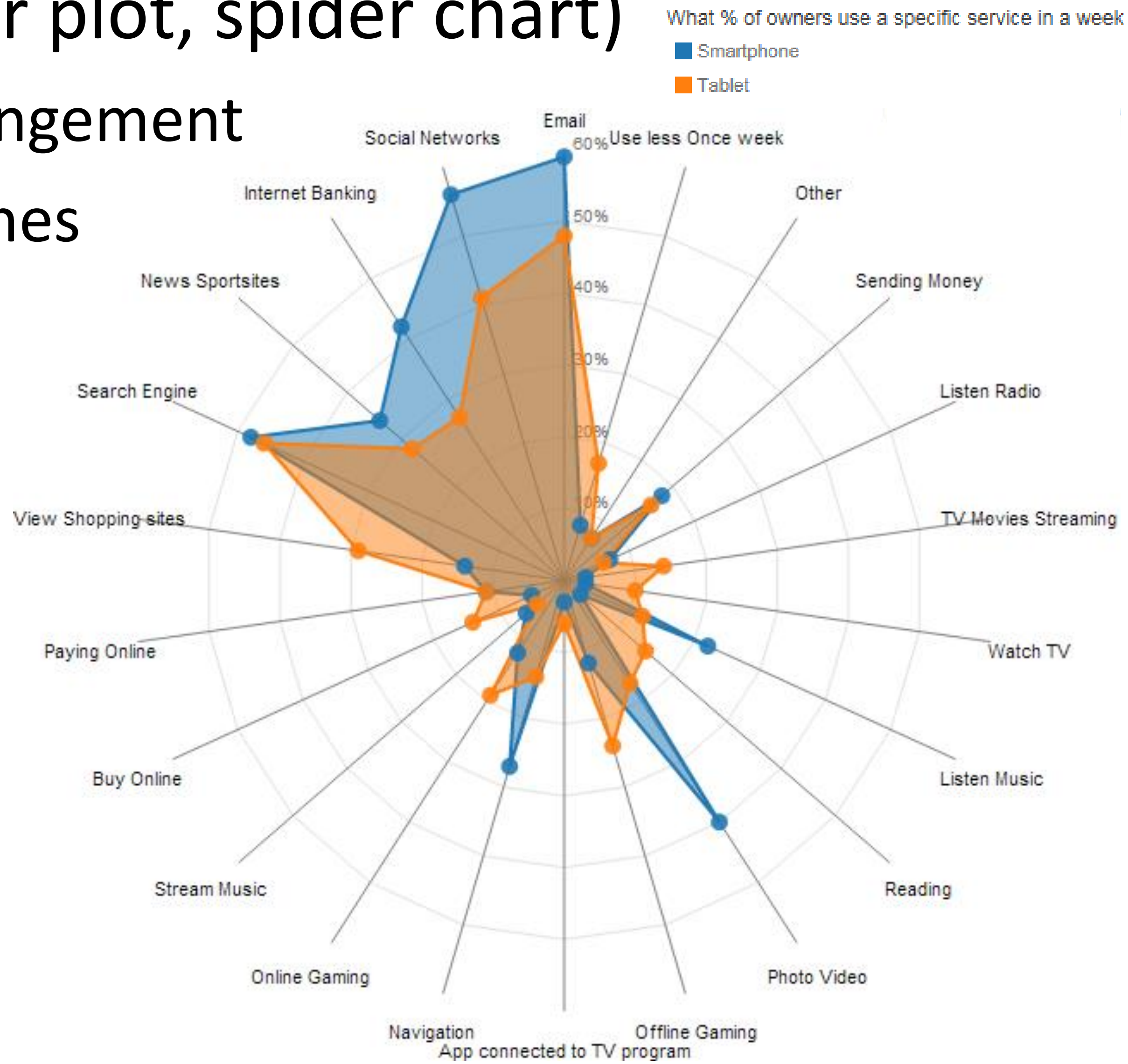
Enhance correlation



[Johansson & Johansson 09]

Diagram techniques

- Radar chart (star plot, spider chart)
  – Radial axes arrangement
  – Items are polylines



http://bl.ocks.org/nbremer/6506614

Diagram techniques

SIEMENS
Ingenuity for life

- Function plot for a 1D scalar field
  - Points $\{(x, f(x)) \mid x \in \mathbb{R}\}$
  - 1D curve
  - Mapping of a discrete set of points to a set of lines by connecting adjacent points

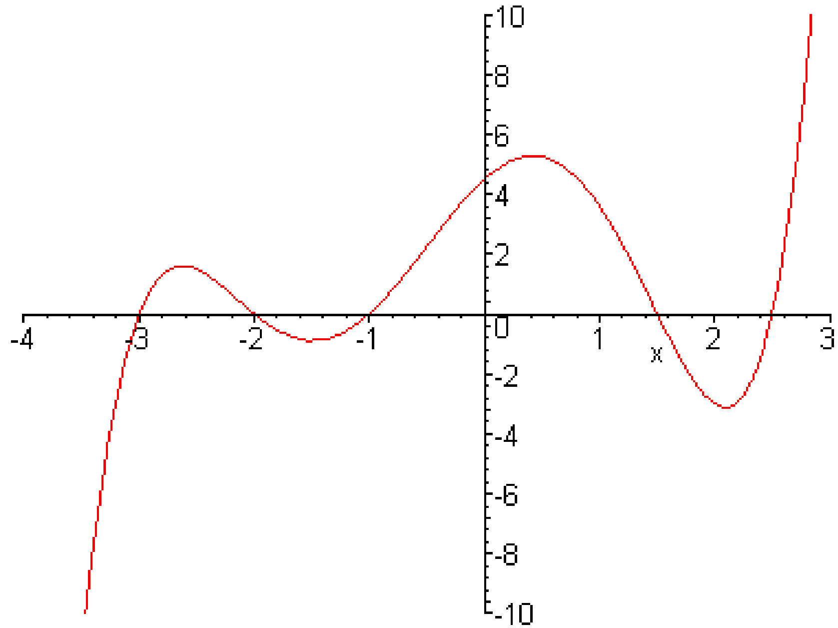


Diagram techniques

- Height field for a 2D scalar field
  - Points $\{(x, y, f(x, y)) \mid (x, y) \in \mathbb{R}^2\}$
  - 2D surface, $f(x, y)$ can be interpreted as height value at (x, y)
  - Mapping of a discrete set of points to a set of faces by connecting adjacent points

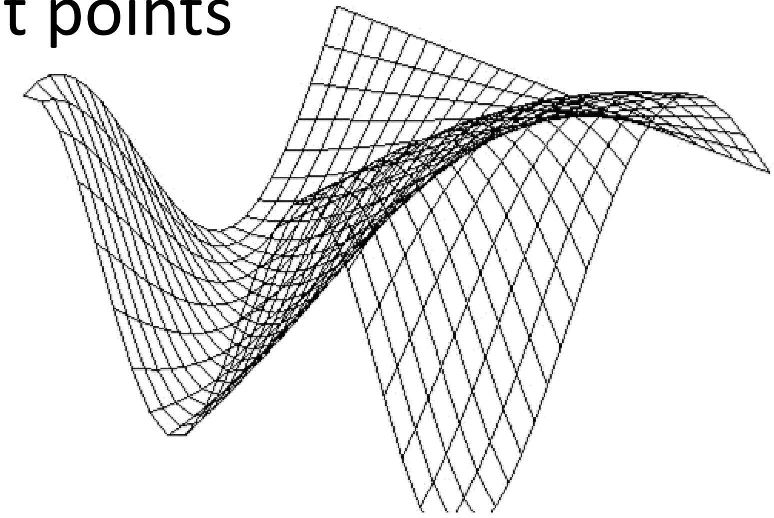
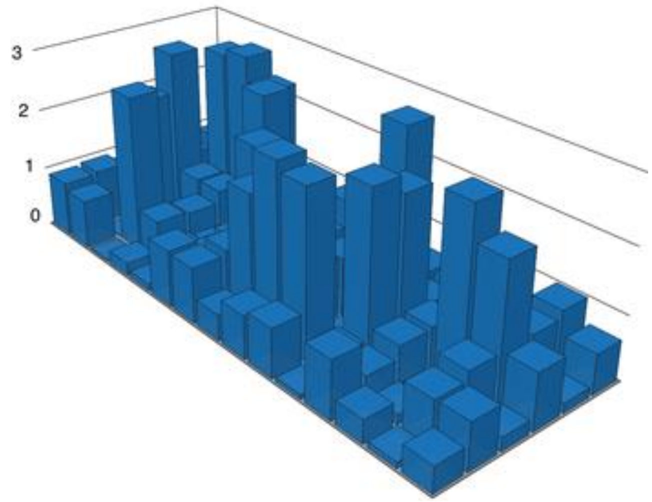
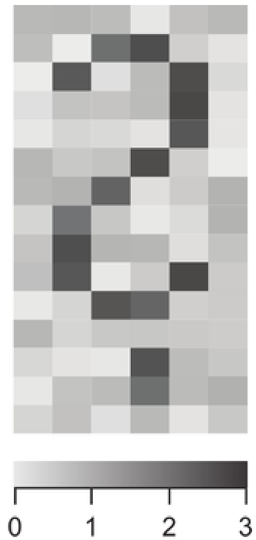


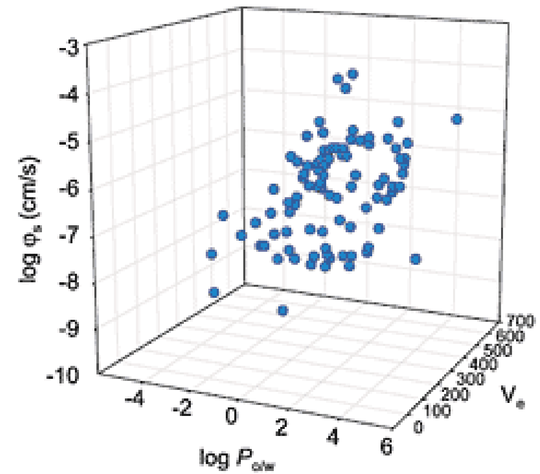
Diagram techniques

SIEMENS
Ingenuity for life

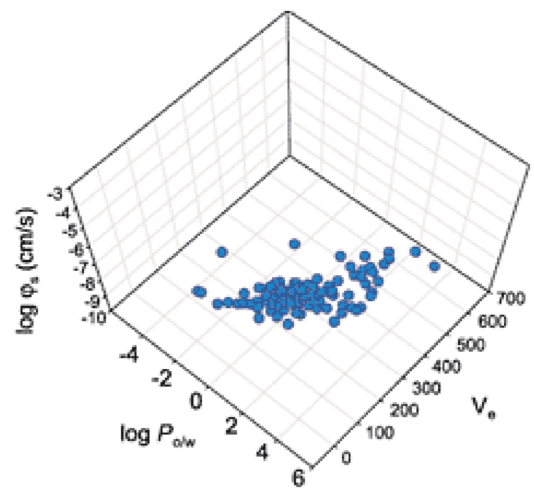
- 3D Pitfalls: Occlusion and Perspective







Which one is the tallest bar?
What is the pattern in the data?



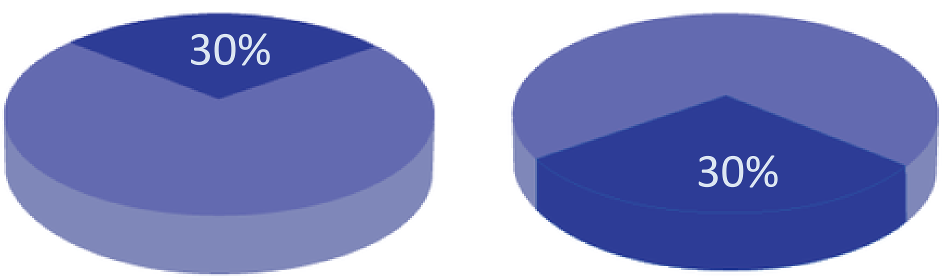
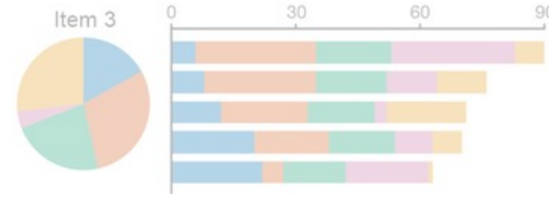
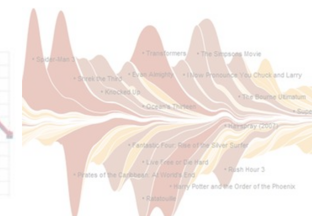
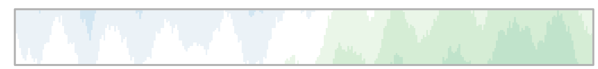


Diagram techniques

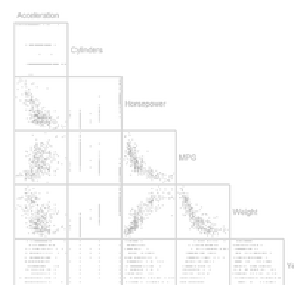
- Categorical + quantitative data
  - Bar/pie chart, stacked bars



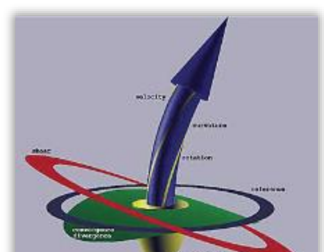
- Time-dependent data
  - Line graph, ThemeRiver, Horizon graph

- Single and multiple variables
  - Histogram, scatterplot, parallel coordinates



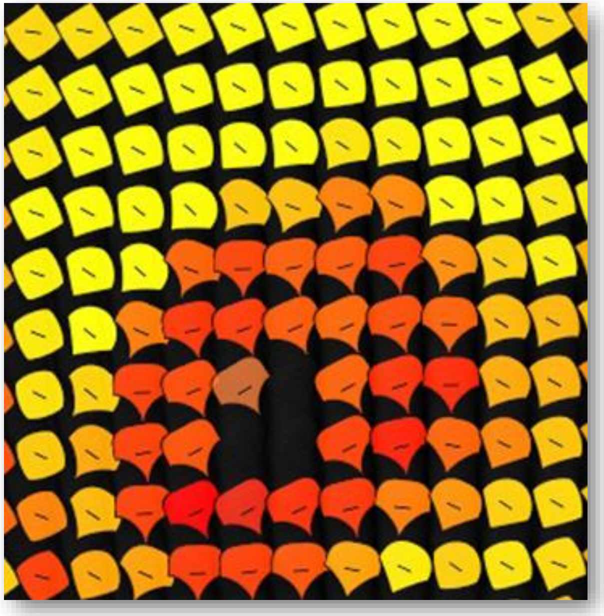
  - Glyphs, color mapping




Glyphs and icons

SIEMENS
Ingenuity for life

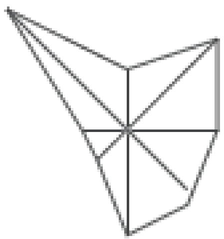
- **Glyphs:** Small independent visual objects that depict attributes of a data record
  - Discretely placed in a display space
  - Data attributes are represented by different visual channels (e.g., shape, color, size, orientation)
  - Visual channels should be easy to distinguish and combine
  - Mainly used for multivariate data



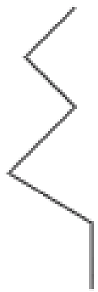
[Borgo et al. 13]

Glyphs and icons

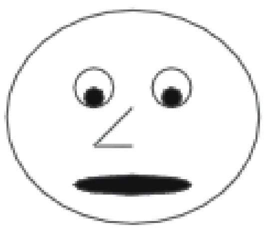
2D glyphs



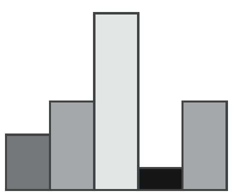
Star glyphs



Stick figures

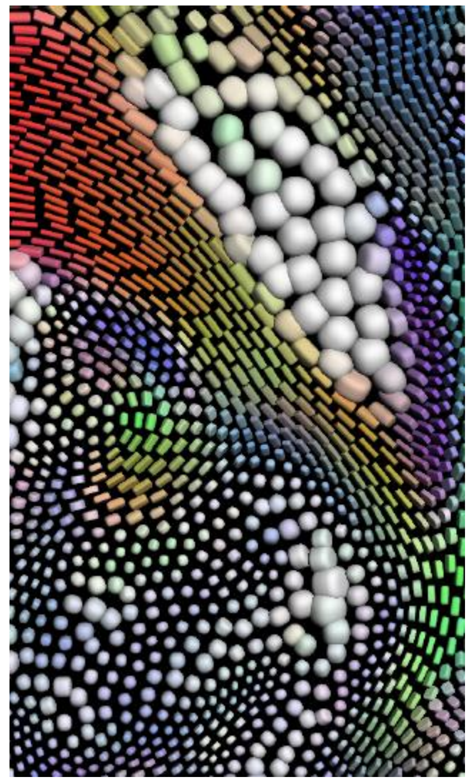


Chernoff faces



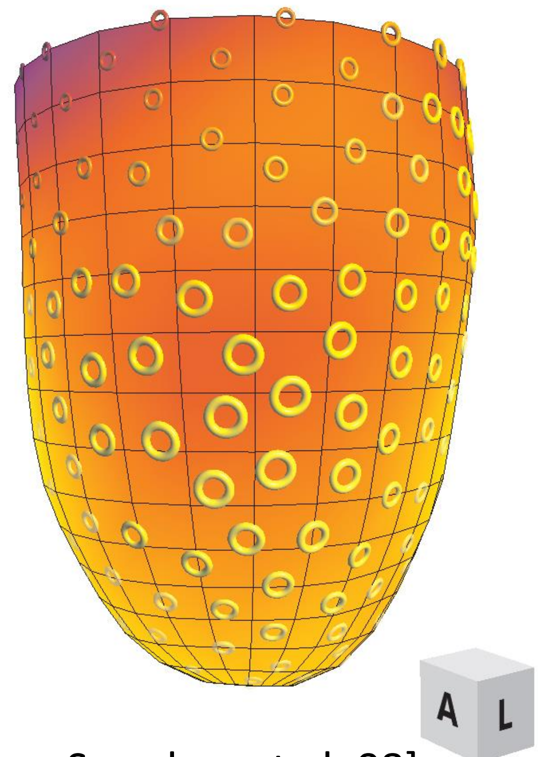
Profile glyphs

3D glyphs



[Kindlmann&Westin 06]

Surface glyphs



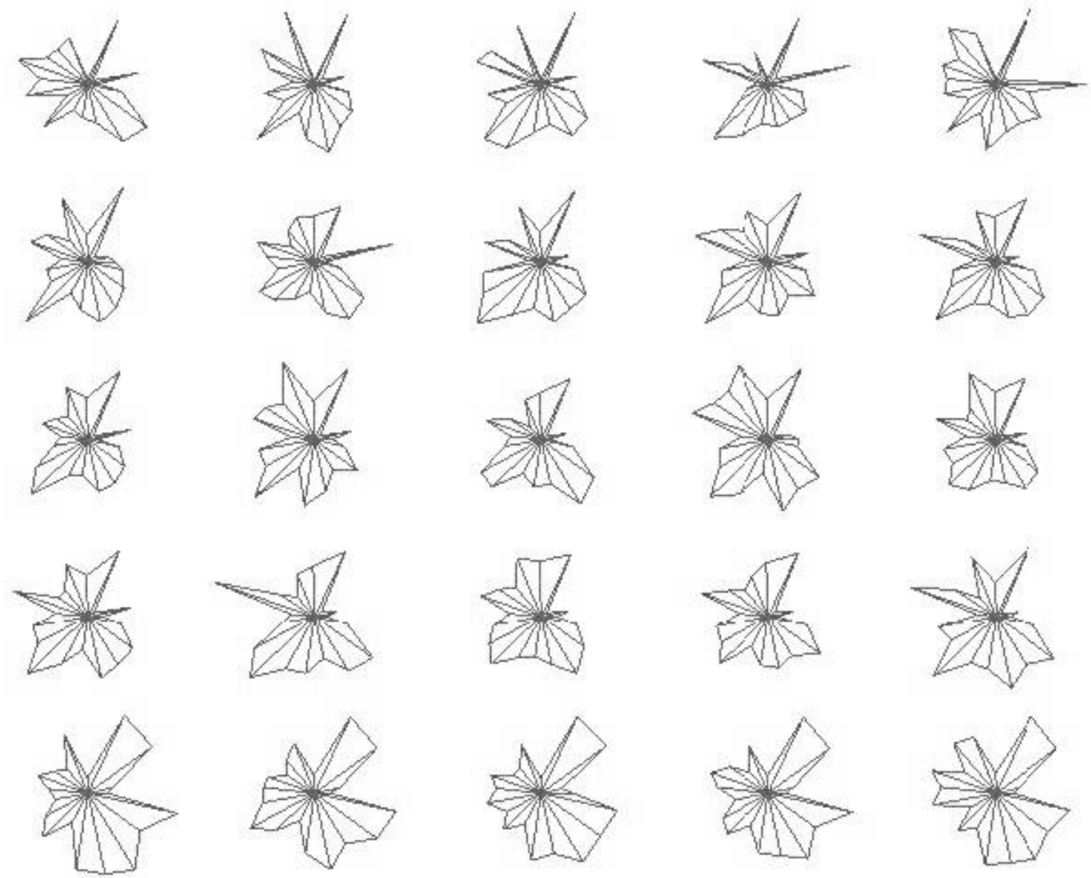
[Meyer-Spradow et al. 08]

[Borgo et al. 13]

Glyphs and icons

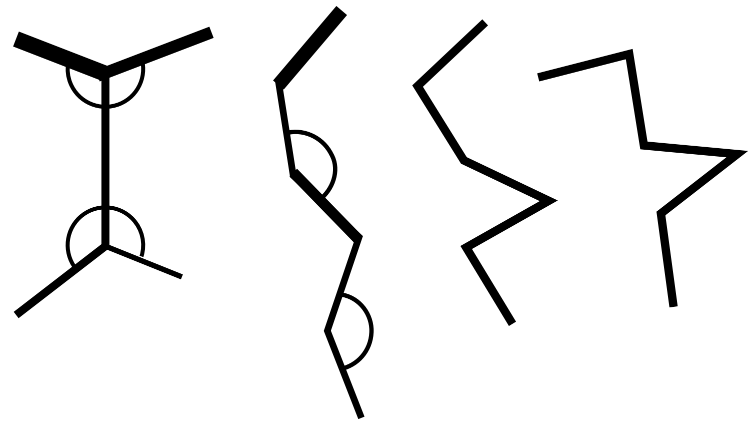
SIEMENS
Ingenuity for life

- Star glyphs
  - A star is composed of equally spaced spikes, originating from center
  - Length of the spikes represents value of respective attribute
  - End of rays connected by line



Glyphs and icons

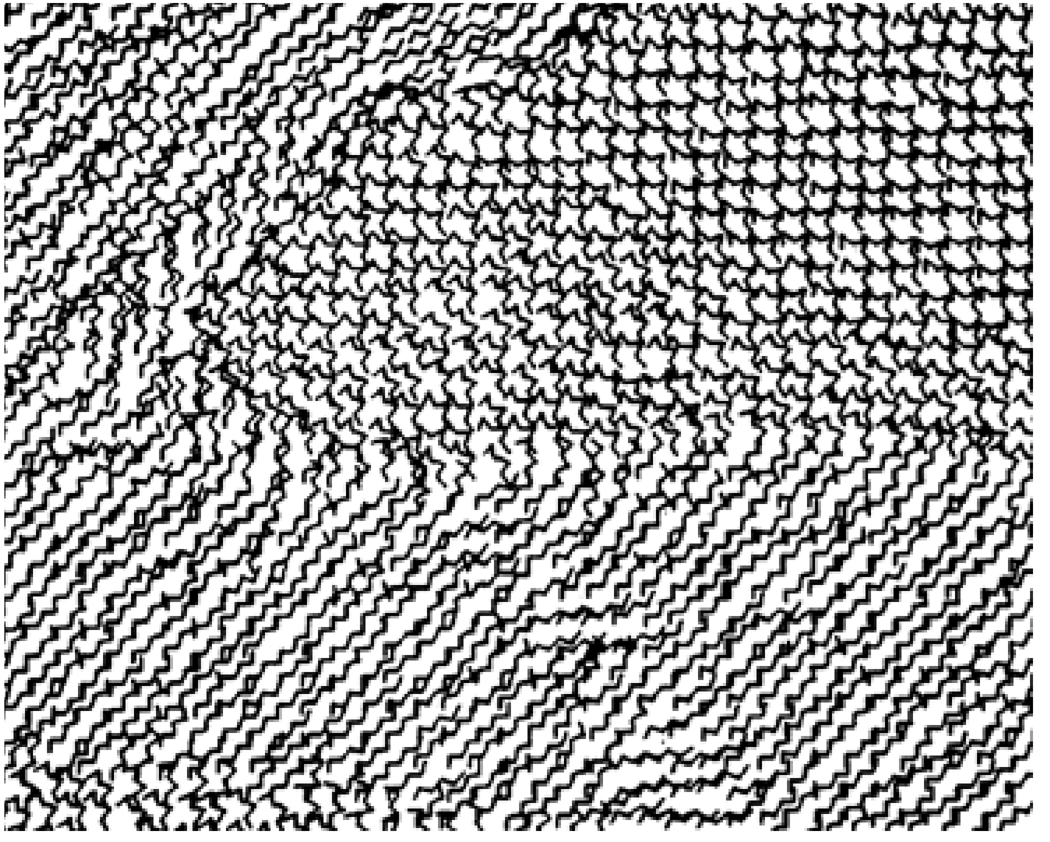
- Stick figures
  - 2D figure with limbs
  - Data encoded by
    - length
    - line thickness
    - angle between lines



[Pickett&Grinstein 88]

Glyphs and icons

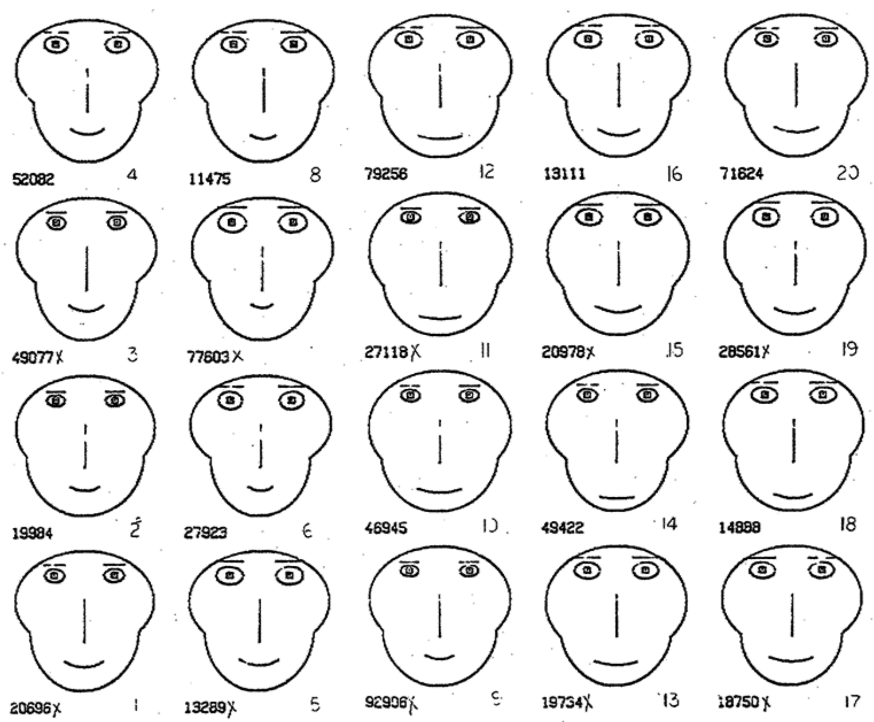
- Stick figures
  - 2D figure with limbs
  - Data encoded by
    - length
    - line thickness
    - angle between lines
  - Recognize texture patterns

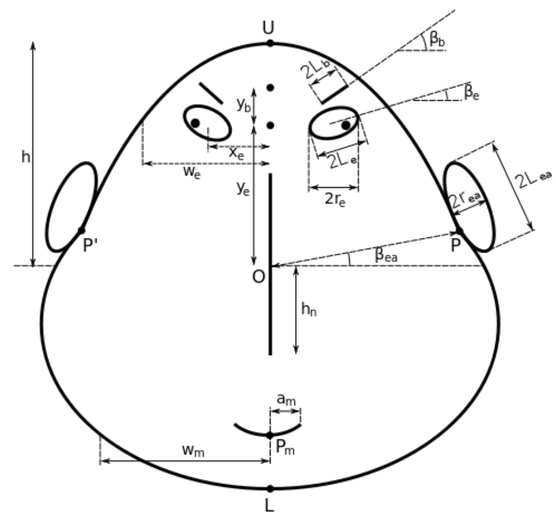


[Pickett&Grinstein 88]

Glyphs and icons

- Chernoff faces
  - Data attributes represented by features of a face (eye position, nose length, mouth form, etc.)



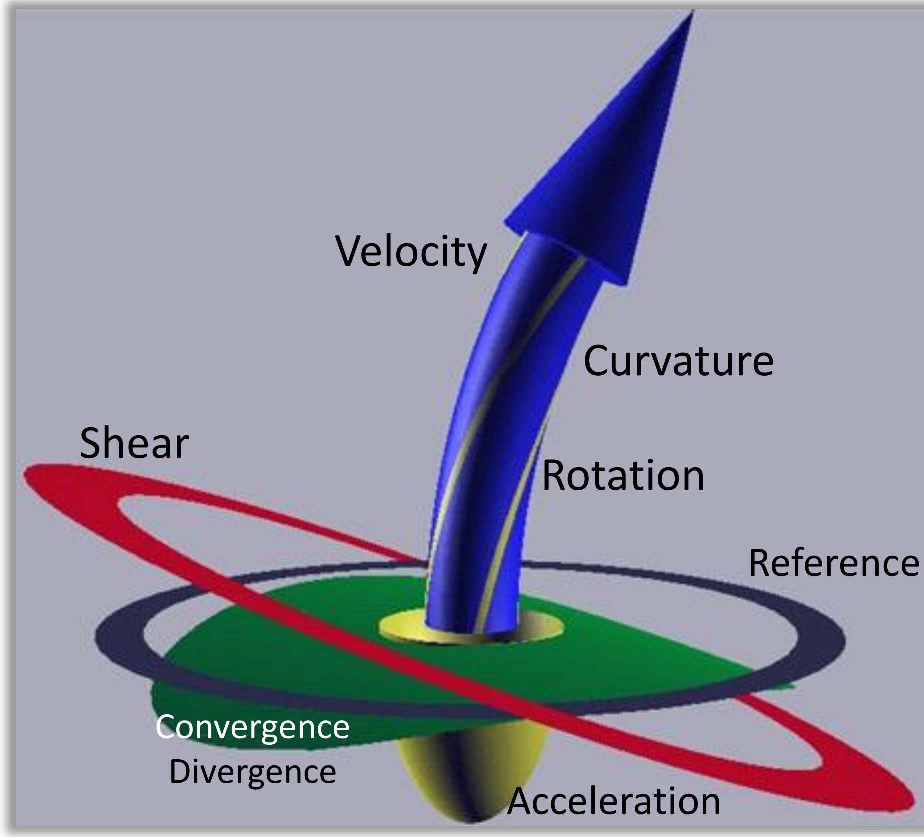


- Faces are perceived holistically
- Efficiency?

Glyphs and icons

SIEMENS
Ingenuity for life

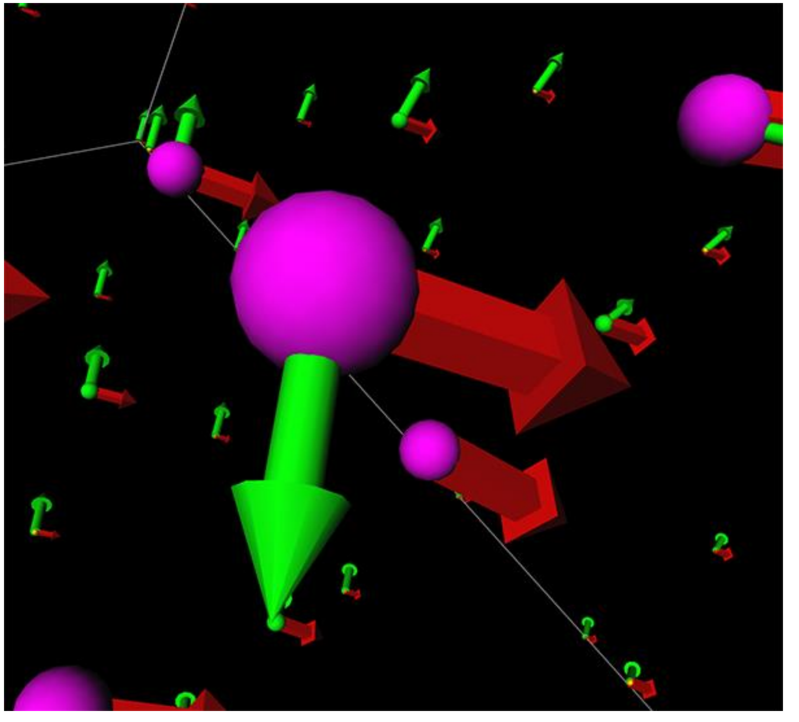
- Local flow probe
  - Depicts multiple flow characteristics
  - Large & complex glyphs need to be sparsely placed to avoid occlusion



[de Leeuw and van Wijk 93]

Glyphs and icons

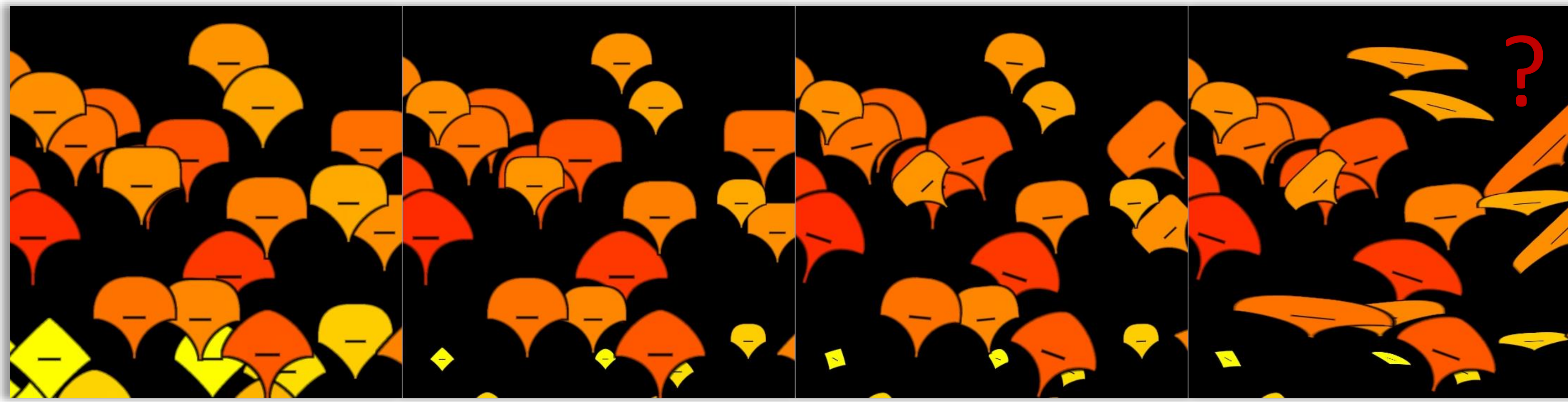
- Glyph shapes
  - Complex combinations of basic primitives (e.g., box, sphere, torus, ellipsoid)



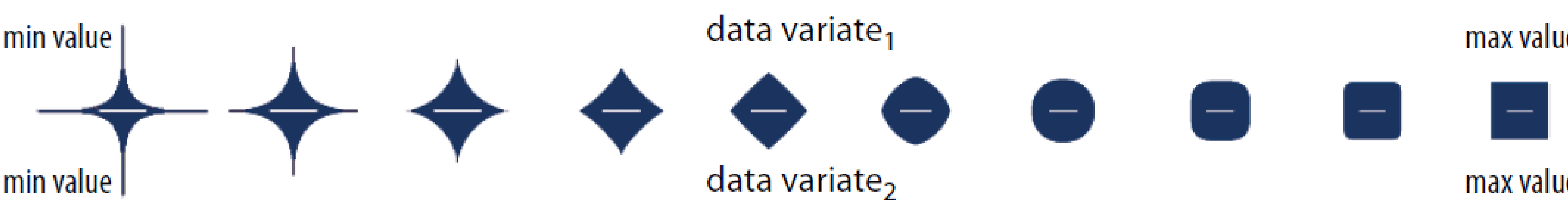
Customized glyphs [Kraus&Ertl 01]

Glyphs and icons

- Separable vs. integral visual channels
  - Perceive each channel independently



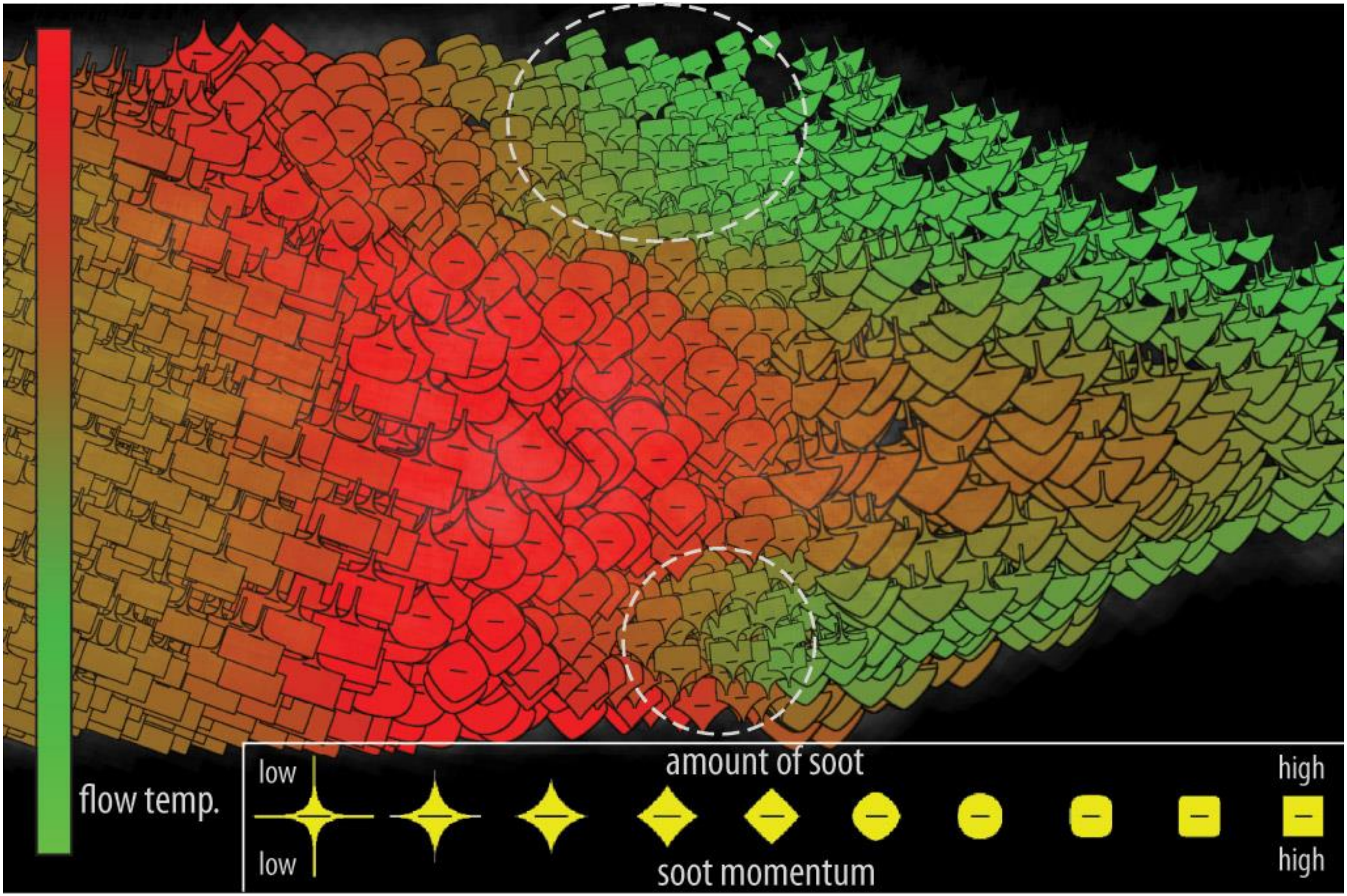
upper/lower shape +color | +size | +rotation | +aspect ratio



[Lie et al. 09]

Diesel Particulate Filter

SIEMENS
Ingenuity for life

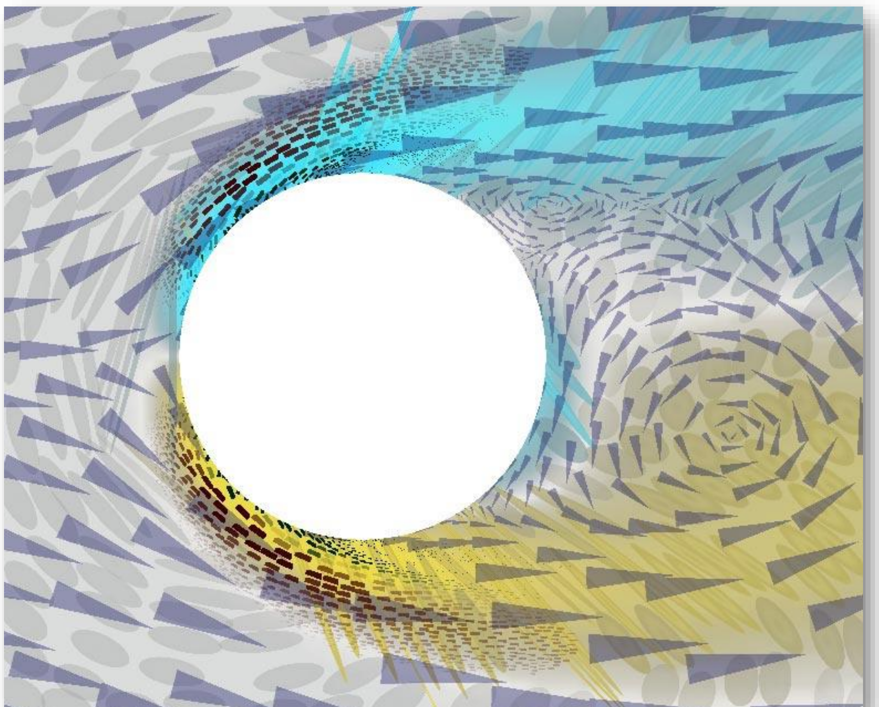


Size & color: flow temp.

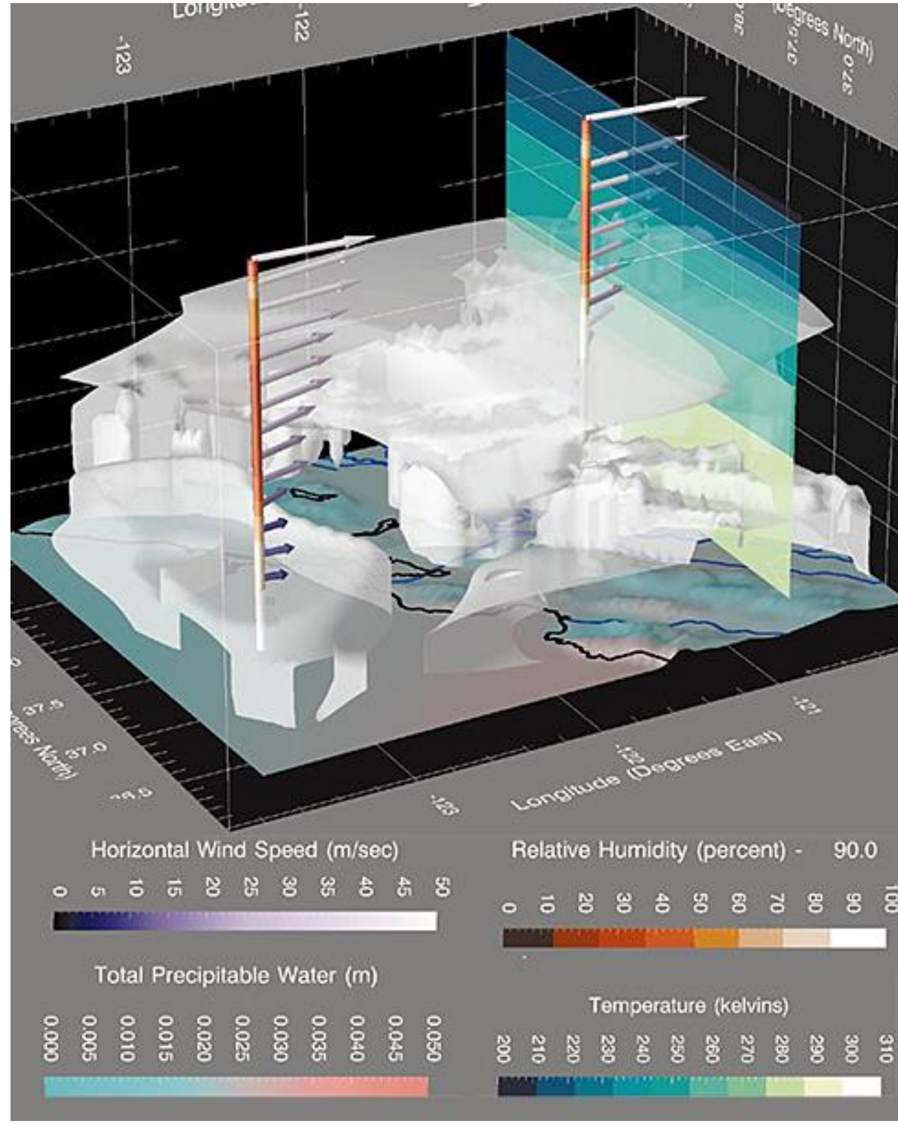
Glyph rotation (-45°, 45°): O_2 fraction

Glyphs and icons

- Hybrid Visualizations
  - Combine glyphs with other visualization techniques



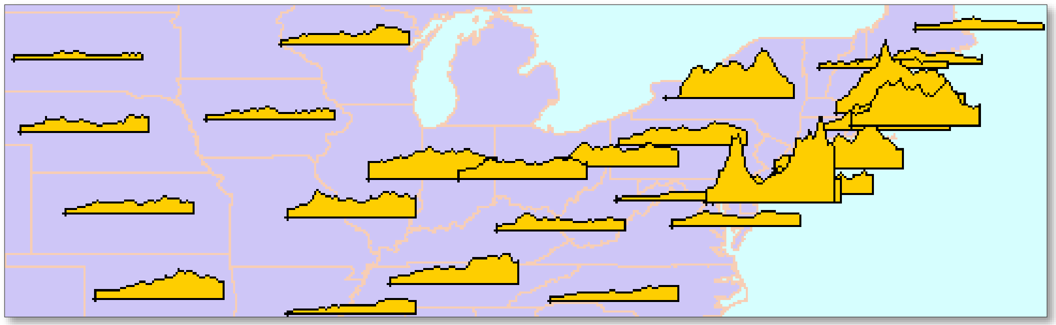
Layering [Kirby et al. 99]

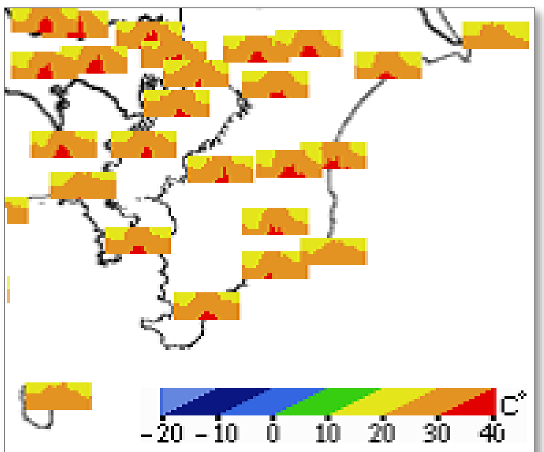


Arrow glyphs, ISO-surfaces [Treinish 99]

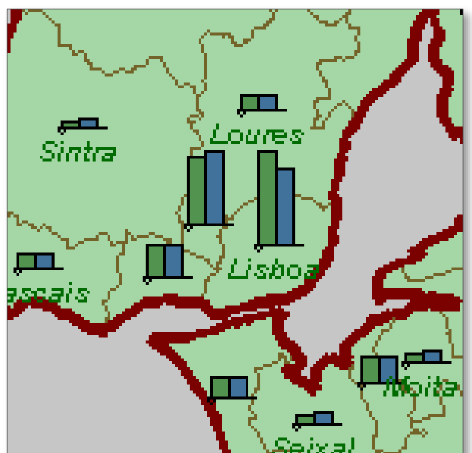
Glyphs and icons

- Glyphs on maps

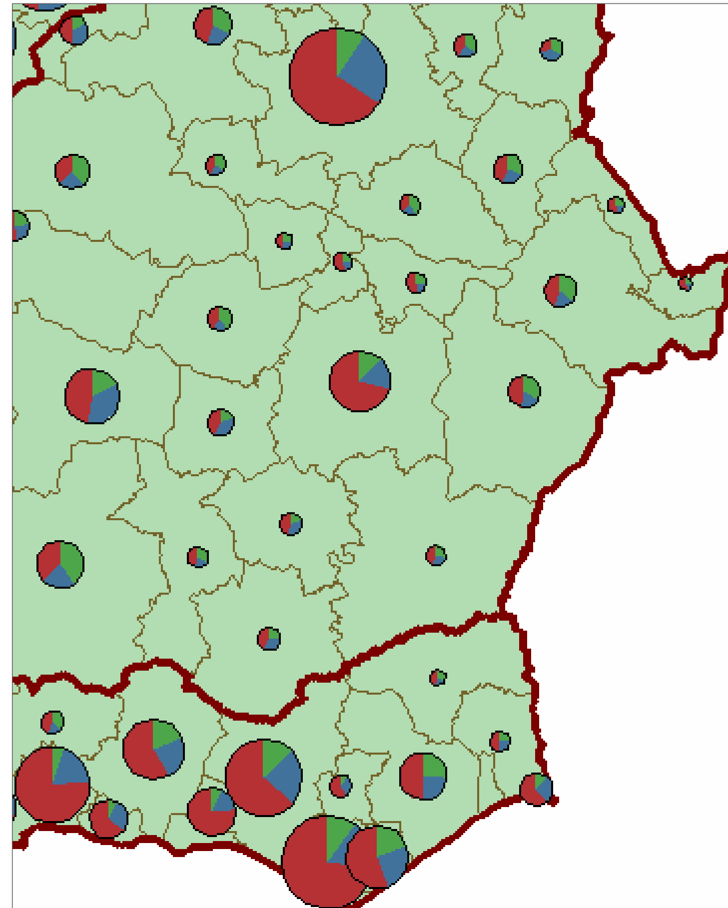




Two-tone colormap
[Saito et al. 05]



Pop. density 1981
Pop. density 1991



Total employment in agriculture
Total employment in industry
Total employment in services
Glyph size = sum of attribute values

[Andrienko]

Glyphs and icons

SIEMENS
Ingenuity for life

- Summary
  - Local & compact representation of many data attributes
  - Just combining visual channels is not enough
  - Glyph design restricted by perceptual limits
  - Design considerations (e.g., separability, perceptually uniform channels, semantics, density vs. complexity, view-point independence)

[Borgo et al. 13]


EUROGRAPHICS 2013/ M. Sbert, L. Szirmay-Kalos — STAR – State of The Art Report

Glyph-based Visualization: Foundations, Design Guidelines, Techniques and Applications

R. Borgo[1], J. Kehrer[2], D. H. S. Chung[1], E. Maguire[3], R. S. Laramee[1], H. Hauser[4], M. Ward[5] and M. Chen[3]

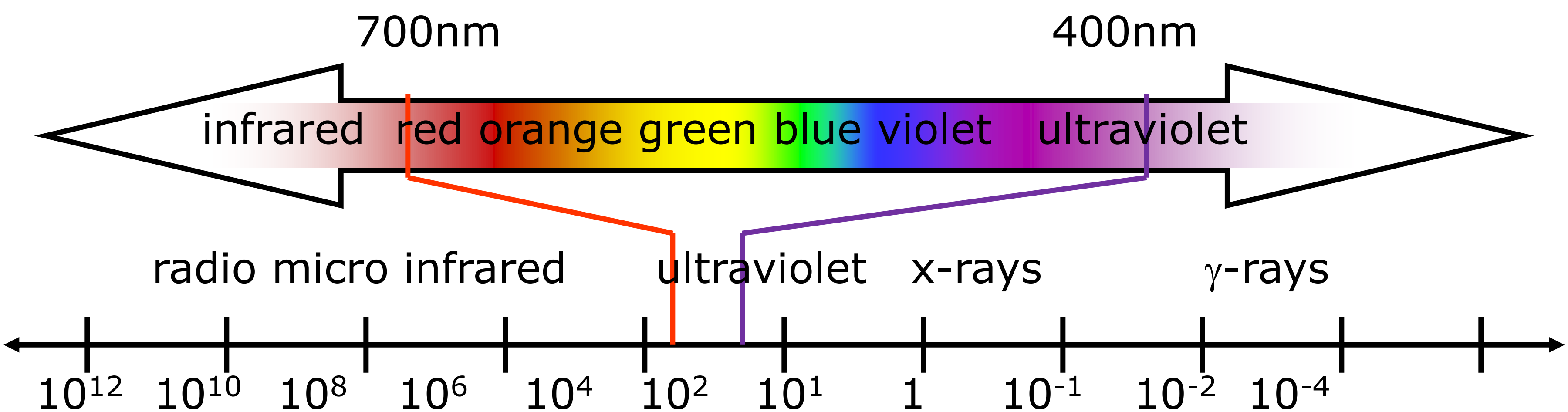
[1] Swansea University, UK; [2] University of Bergen and Vienna University of Technology, Austria; [3] University of Oxford, UK; [4] University of Bergen, Norway; [5] Worcester Polytechnic Institute, USA

Abstract

This state of the art report focuses on glyph-based visualization, a common form of visual design where a data set is depicted by a collection of visual objects referred to as glyphs. Its major strength is that patterns of multivariate data involving more than two attribute dimensions can often be more readily perceived in the context of a spatial relationship, whereas many techniques for spatial data such as direct volume rendering find difficult to depict with multivariate or multi-field data, and many techniques for non-spatial data such as parallel coordinates are less able to convey spatial relationships encoded in the data. This report fills several major gaps in the literature, drawing the link between the fundamental concepts in semiotics and the broad spectrum of glyph-based visualization, reviewing existing design guidelines and implementation techniques, and surveying the use of glyph-based visualization in many applications.

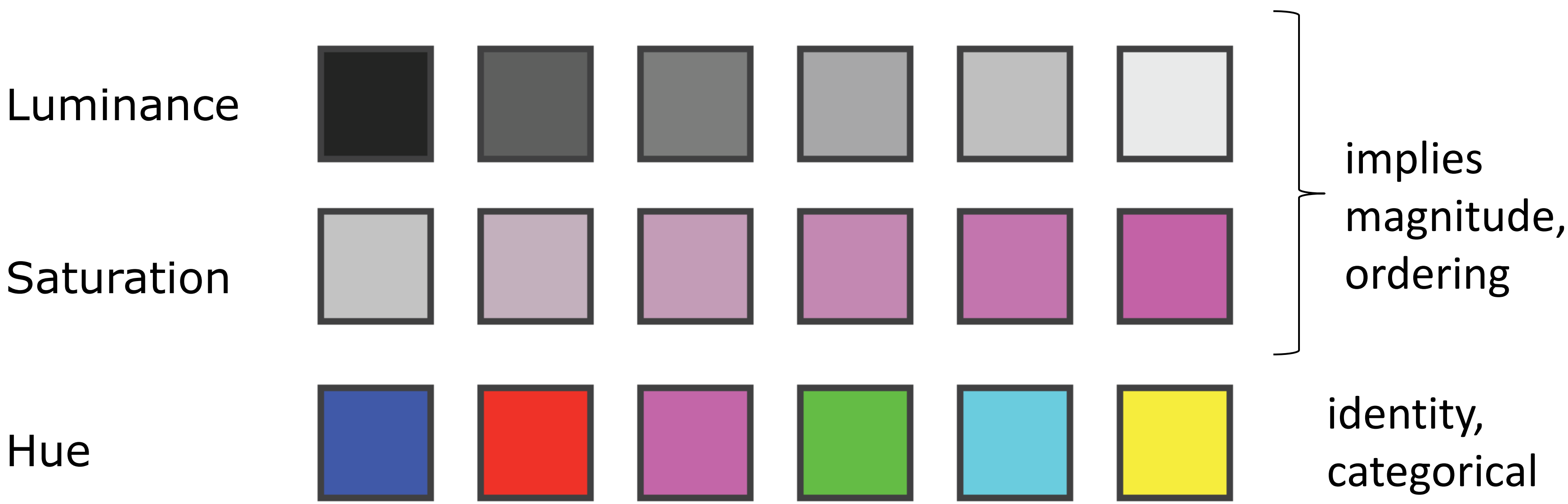
Color

- Light is electro-magnetic radiation
- Different wavelengths are perceived as different colors
  - Human eye can only see light between 380nm and 780nm (visible spectrum)



The human visual system

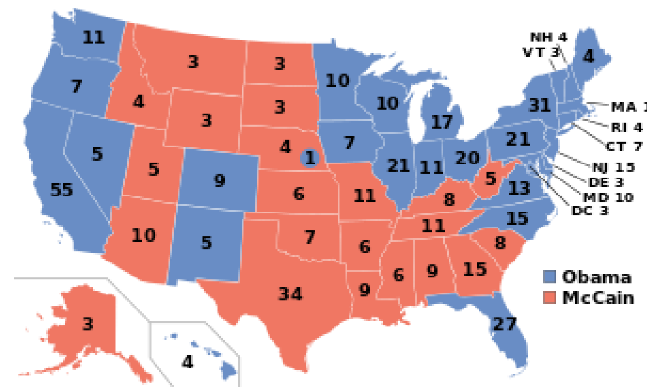
- Visual effect of chromatic light (light spectrum) can be characterized by 3 channels
  - **Hue:** dominant wavelength
  - **Saturation:** pureness, amount of white light
  - **Luminance/Brightness:** intensity of light



Color mapping

- Color mapping tasks
  - Emphasize a specific target in a crowded display (pop-out)

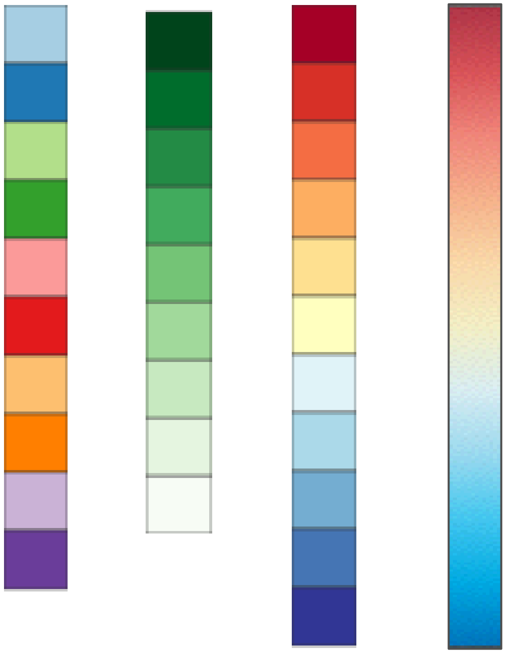
  - Group, categorize, and chunk information



- Possible problems:
  - Dependent on viewing and stimulus conditions
  - Distract the user when inadequately used
  - Ineffective for color deficient individuals
  - Results in information overload

Color mapping

- Color maps can be
  - categorical vs. ordered
  - sequential vs. diverging
  - discrete vs. continuous

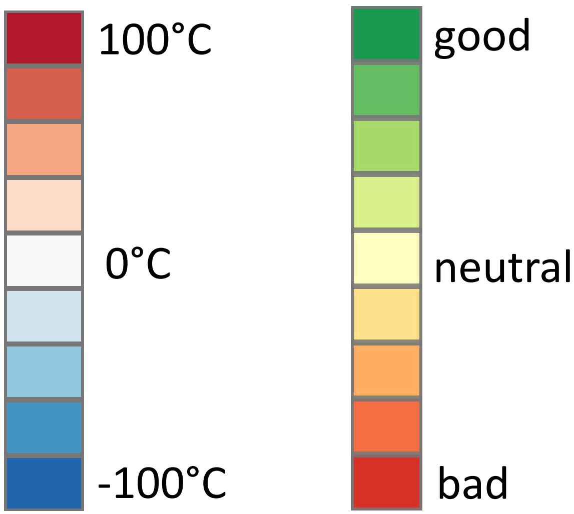


http://colorbrewer2.org

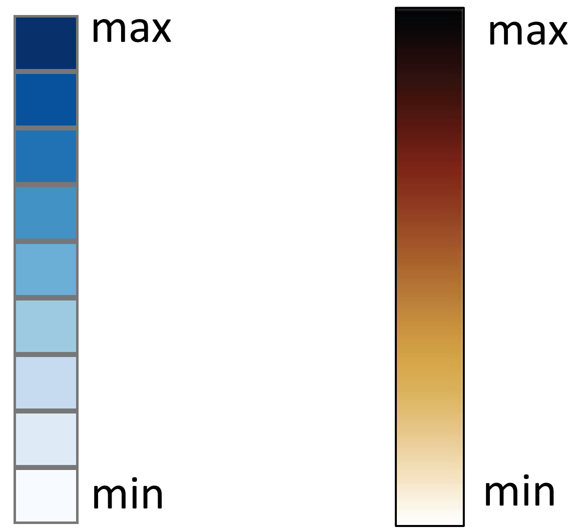
Color mapping

SIEMENS
Ingenuity for life

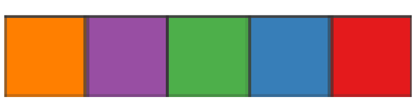
- Use color map that fits data characteristics
  - Diverging data



  - Sequential data



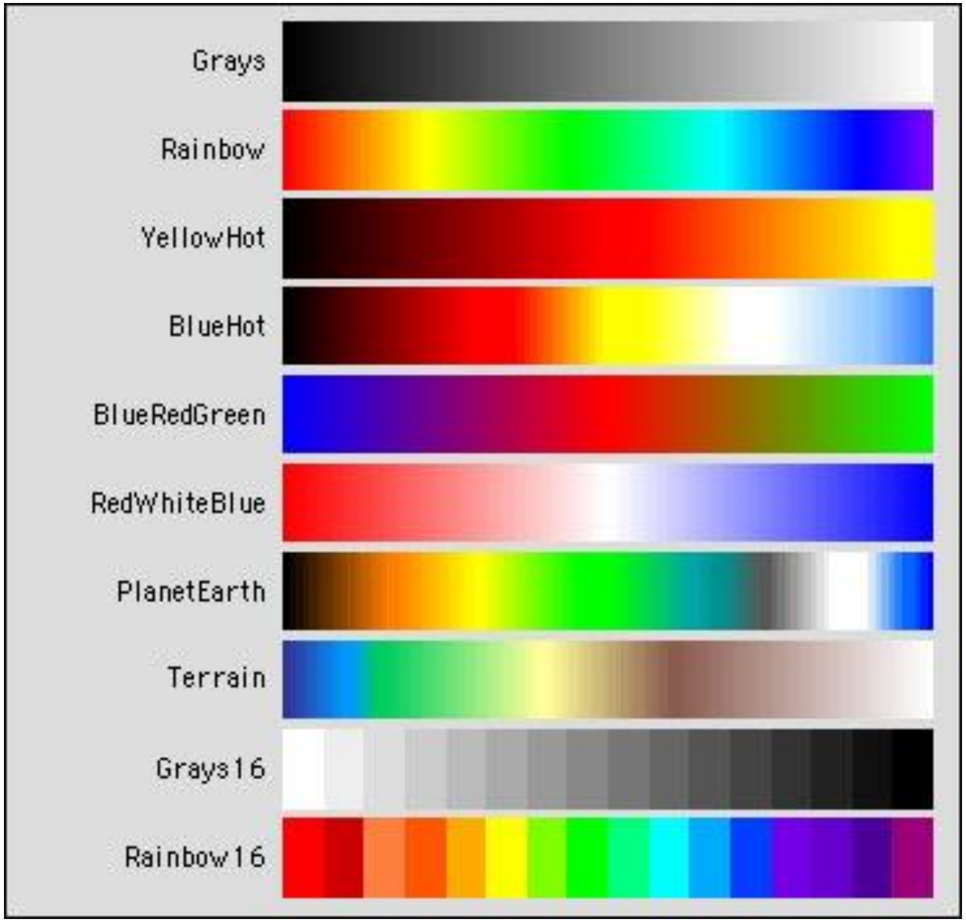
  - Categorical data



[Brewer 99, Stolte et al. 02]

Color mapping

- Some prominent color maps

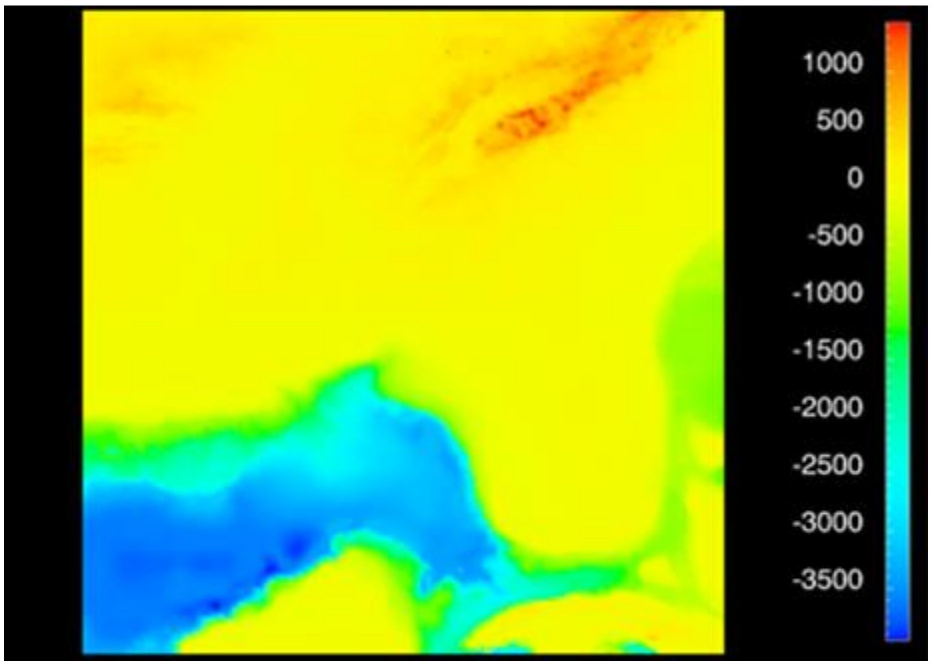


http://www.exelisvis.com/docs/LoadingDefaultColorTables.html

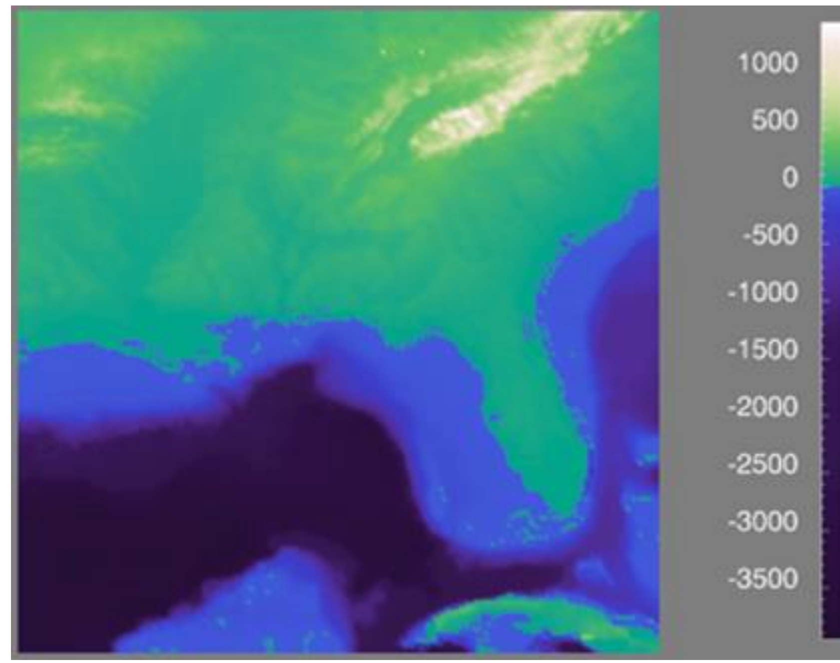
Color mapping

SIEMENS
Ingenuity for life

- Rainbow color map: A useful default?



Rainbow colormap

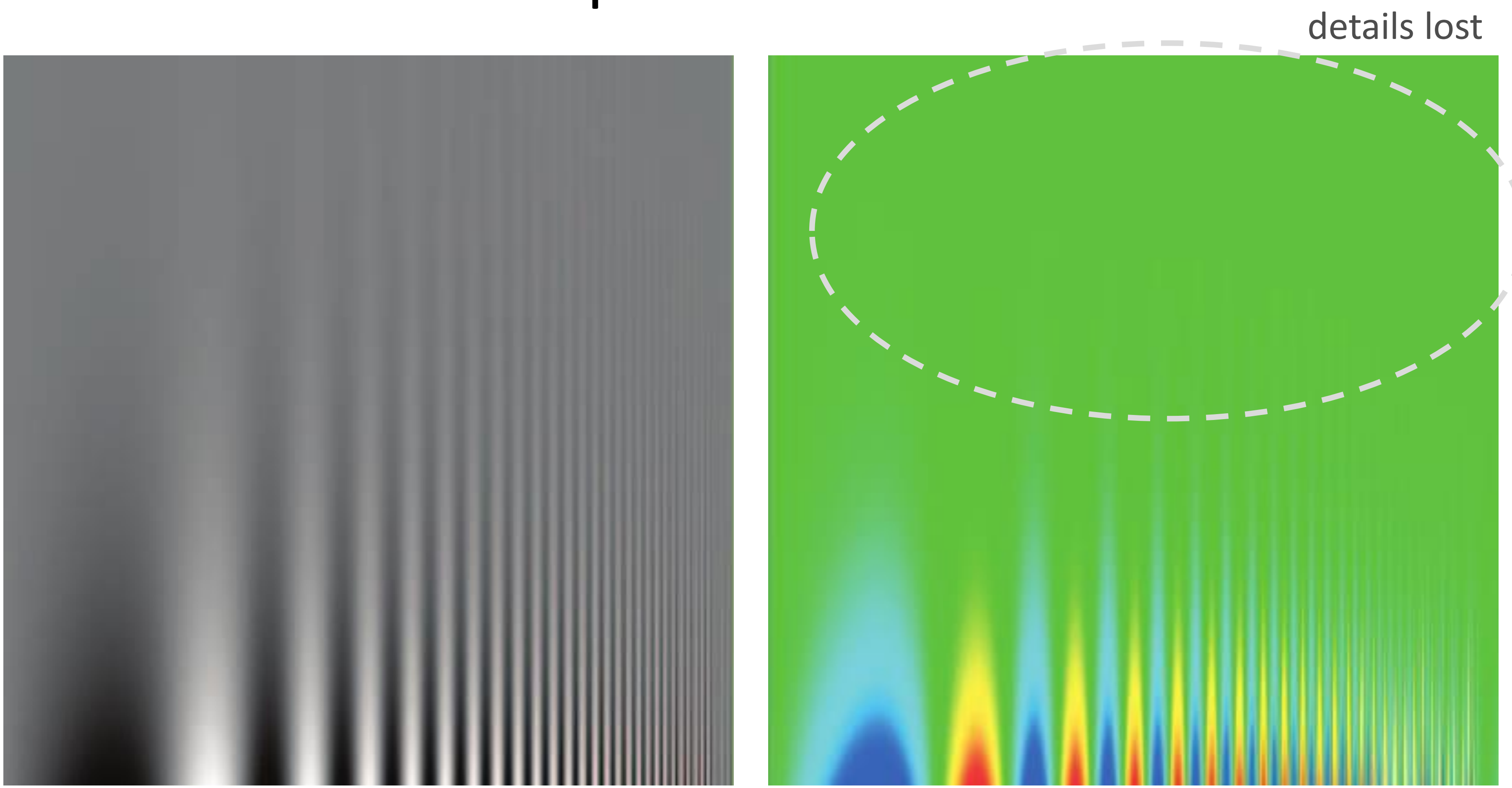


"Perceptual" colormap

Florida peninsula with coastline clearly visible – Appalachian Mountains in lighter colors

Color mapping

- Rainbow color map: A useful default?

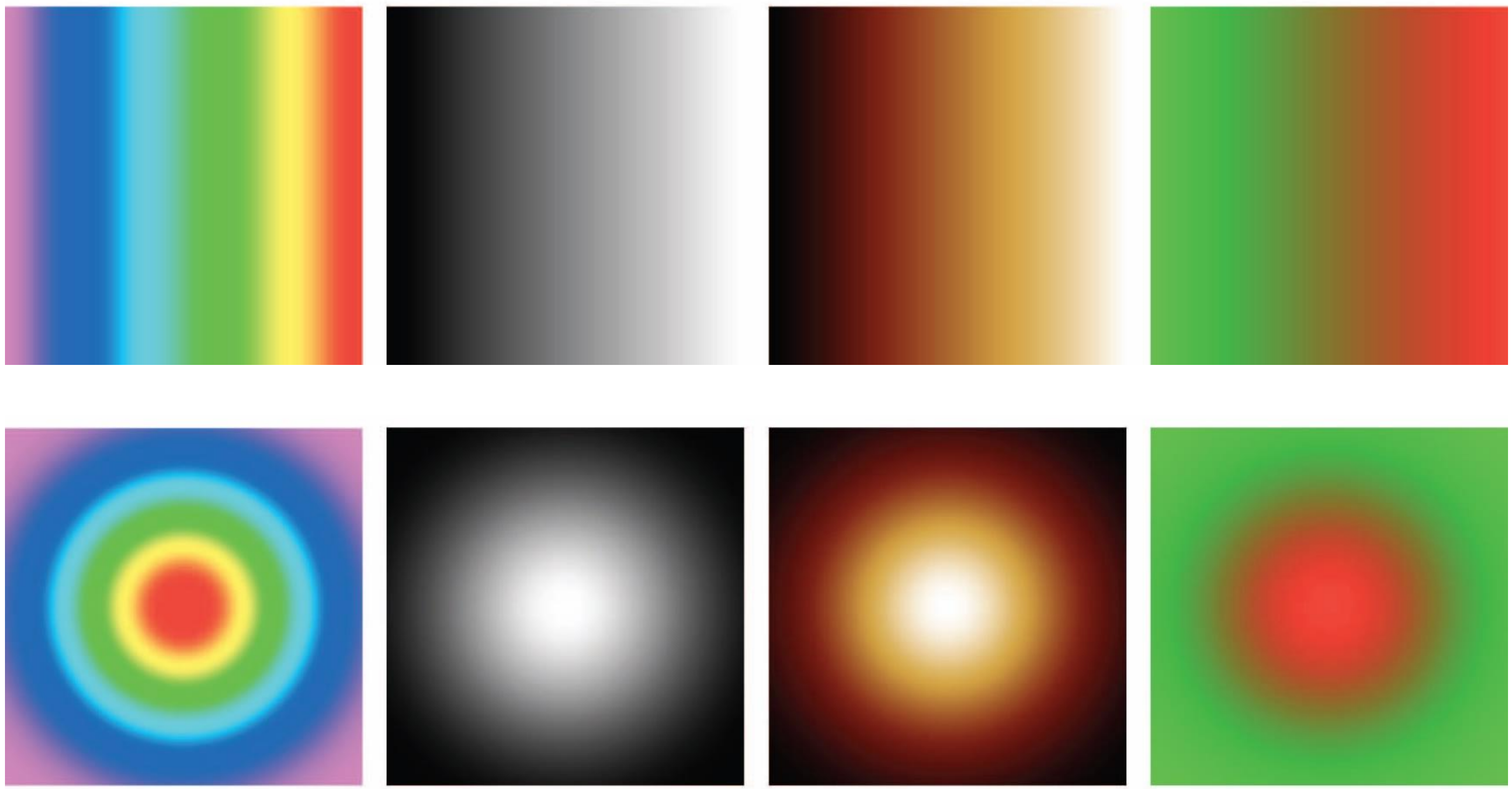


Color mapping

- Perceptual linear: equal steps in color map (i.e., magnitude of data) should be perceived equally

Data: Linear increase in values from left to right

Data: Radial symmetric decrease from center



Rainbow colormap | Grayscale colormap | Black-body radiation | Green-red isoluminant

[Borland&Taylor II 07]

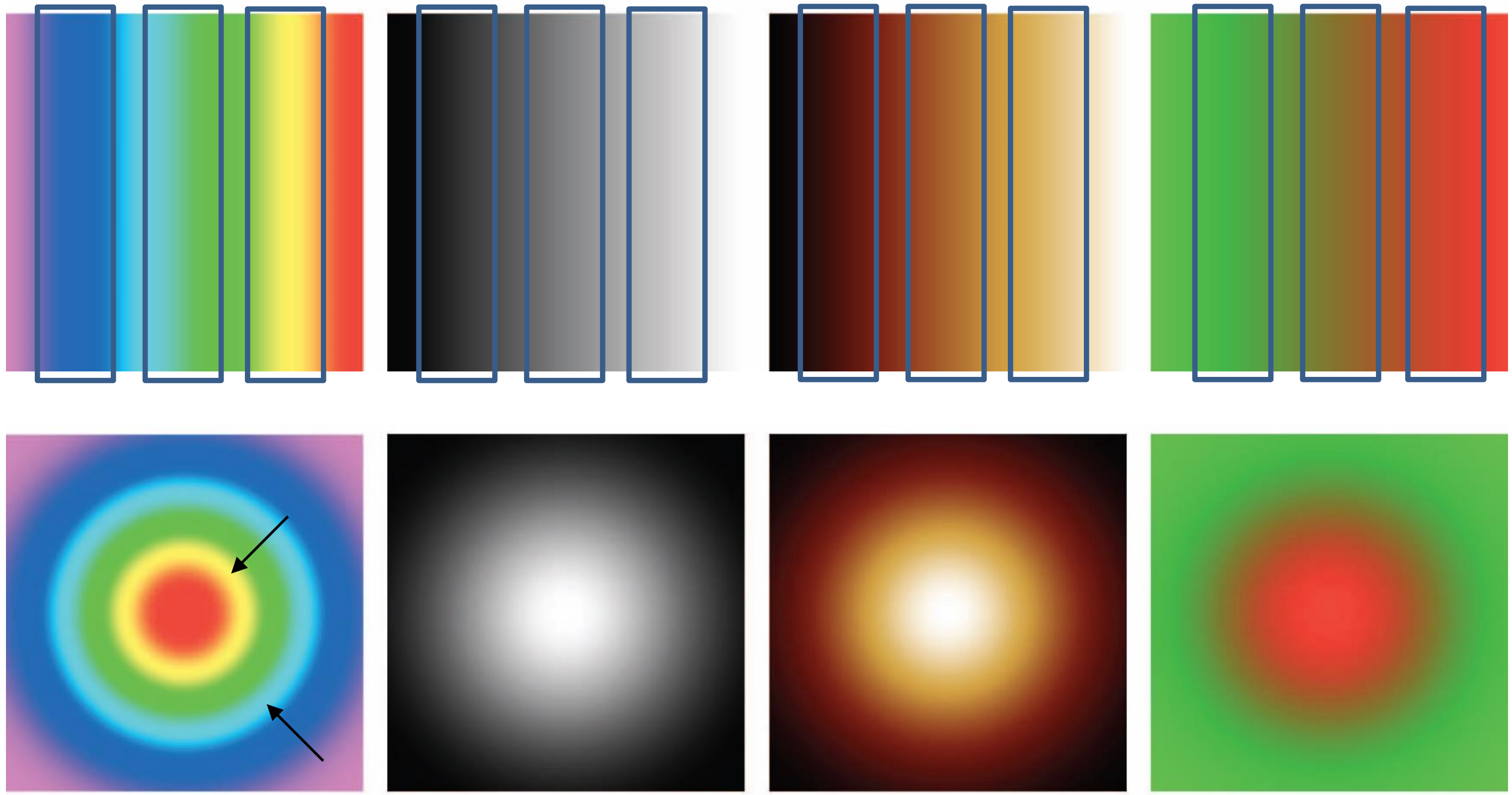
Color mapping

SIEMENS
Ingenuity for life

- **Perceptual linear**: equal steps in color map (i.e., magnitude of data) should be perceived equally

Data: Linear increase in values from left to right

Data: Radial symmetric decrease from center



Rainbow colormap is perceptually **non**linear & introduces artifacts

Grayscale colormap

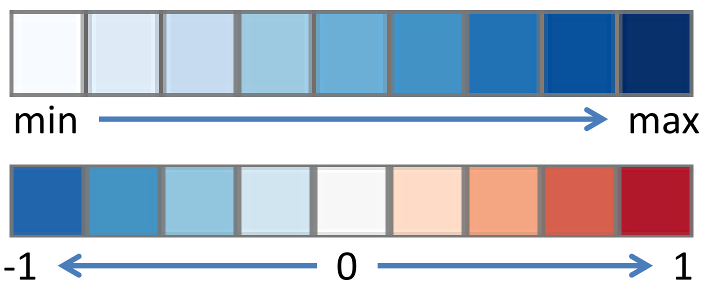
Black-body radiation

Green-red isoluminant

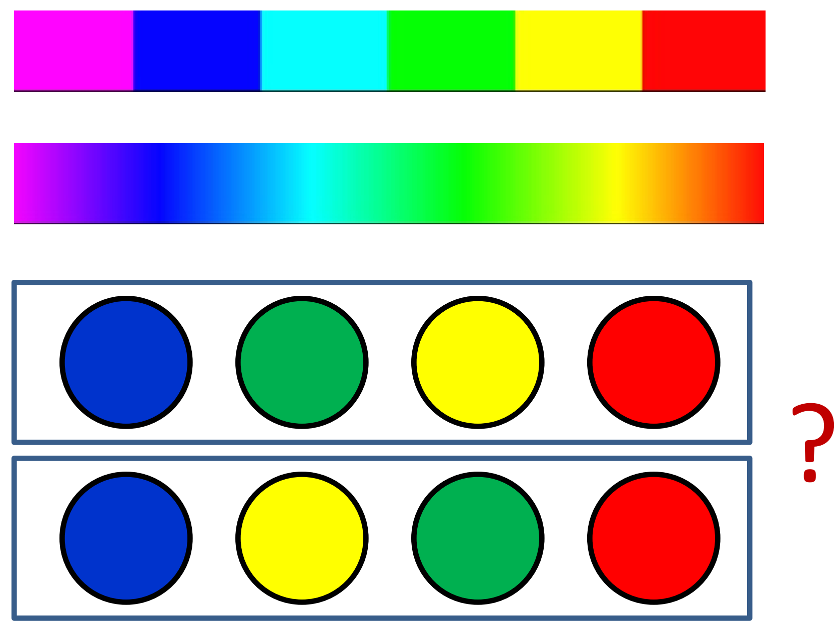
[Borland&Taylor II 07]

Color mapping

- Perceptual ordering: ordering of data should be represented by ordering of colors
  - $x_1 < x_2 < \ldots < x_n \Rightarrow E(c_1) < E(c_2) < \ldots < E(c_n)$ E: visual sensation



sequential vs. diverging

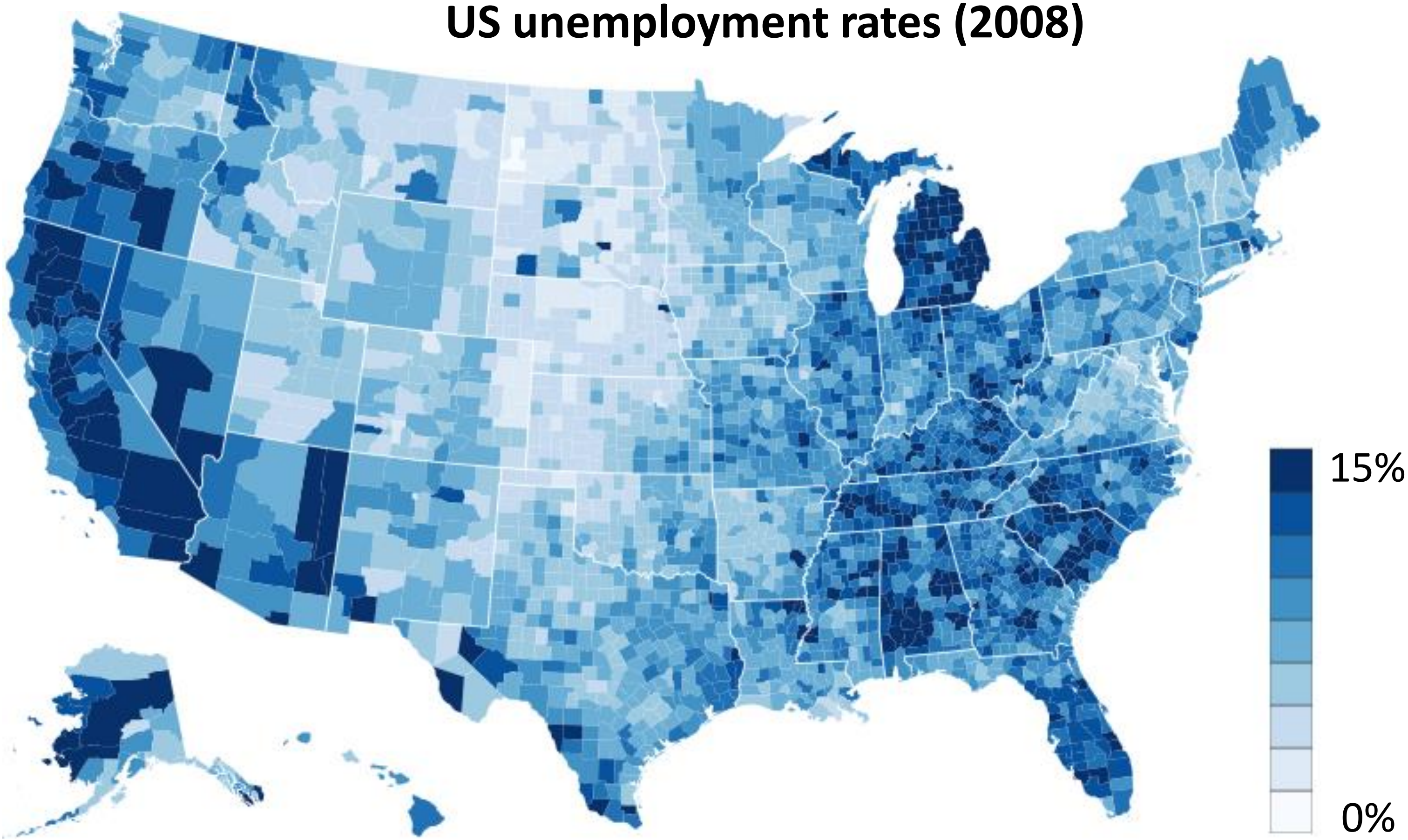


Rainbow colormap is perceptually unordered

Color mapping

- Color coding of quantitative data
  - Choropleth map

US unemployment rates (2008)



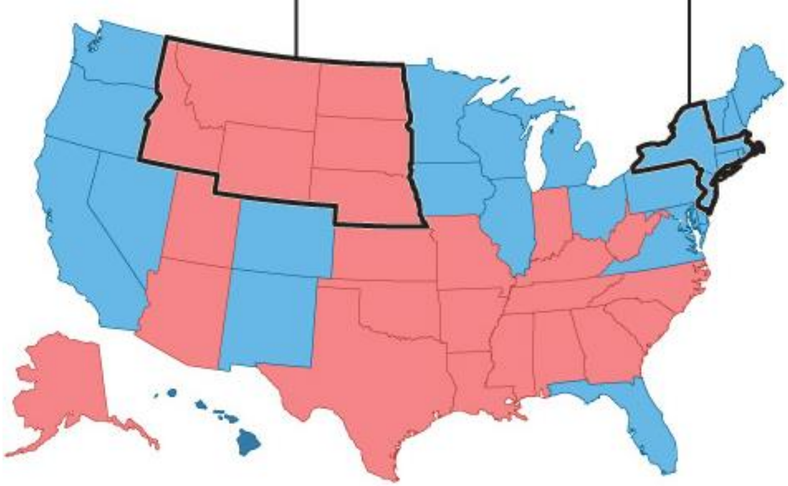
http://bl.ocks.org/mbostock/4060606

Color mapping

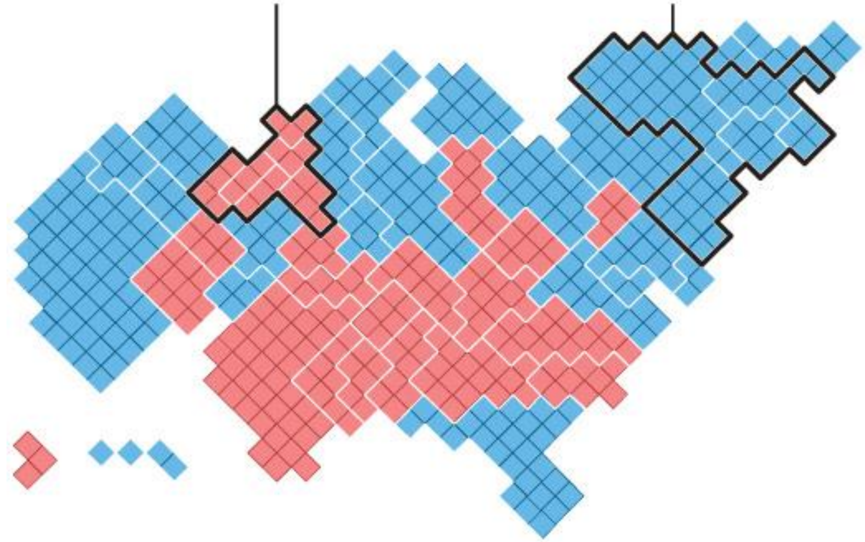
SIEMENS
Ingenuity for life

- Color coding of categorical data

2016 presidential election



Geographic map

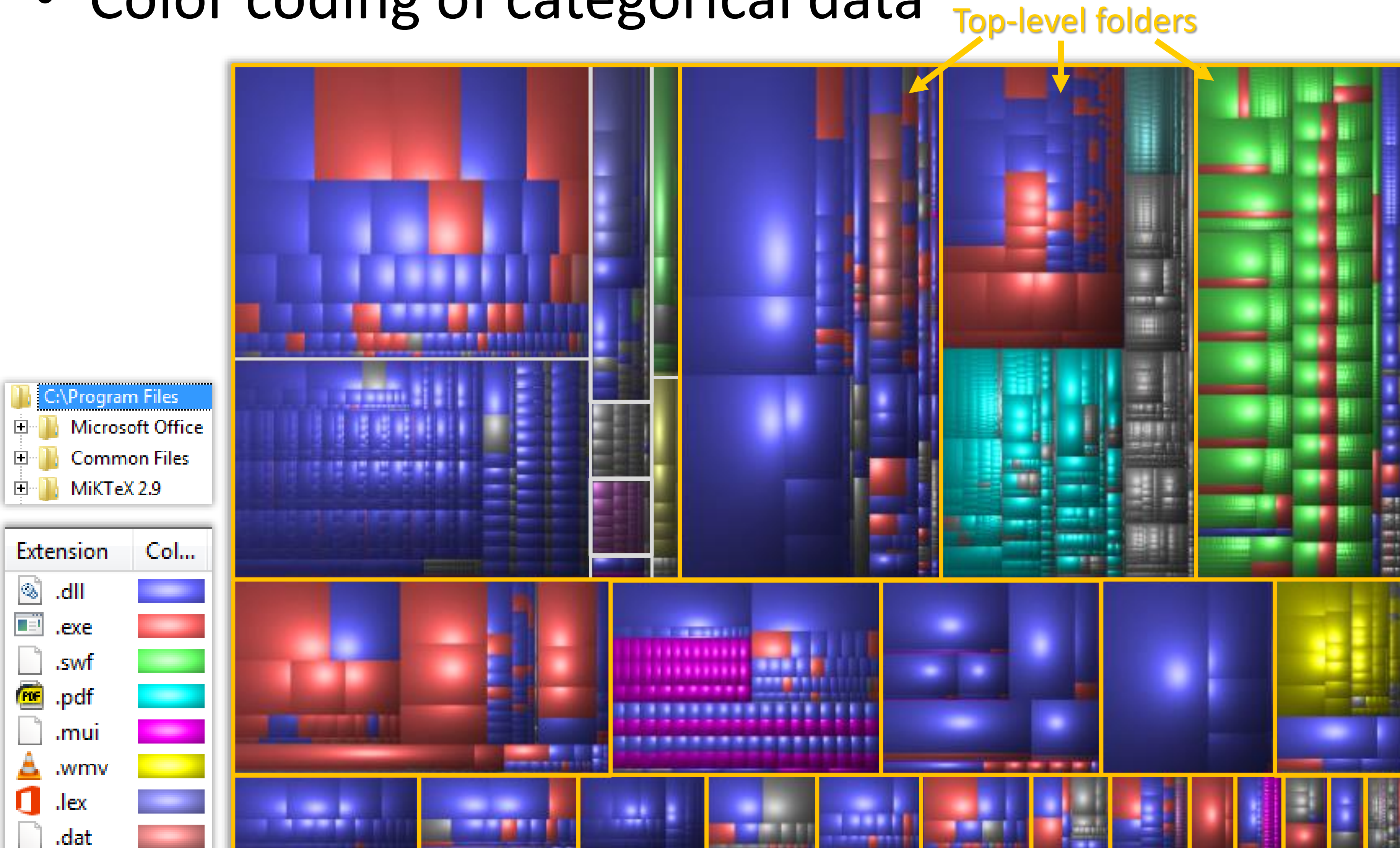


Cartogram of Electoral votes

washingtonpost.com

Color mapping

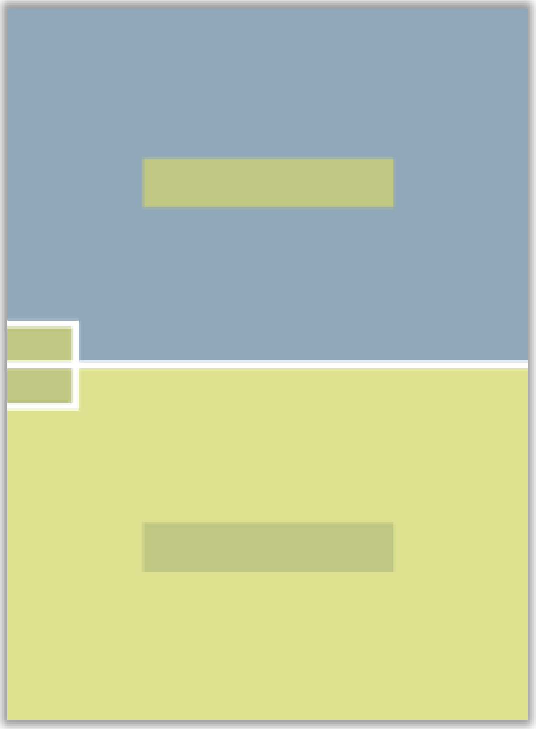
- Color coding of categorical data



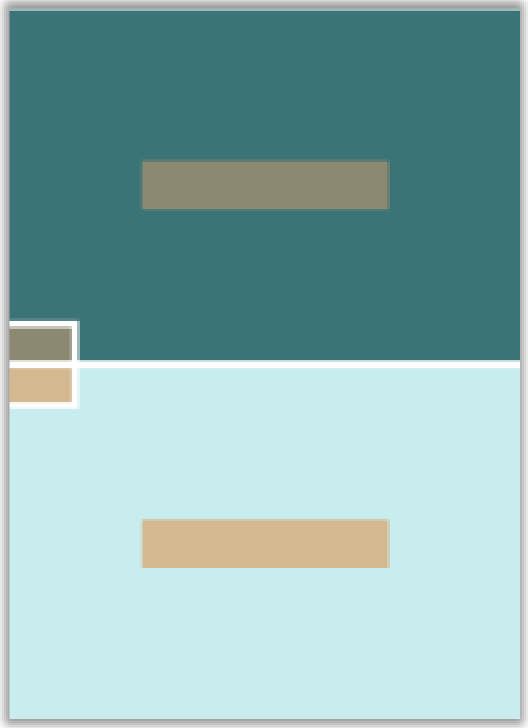
File system visualization

Color mapping

- Things to know when mapping data to colors
  - Perceived color is highly context dependent



Same color can look different



Different colors can look nearly the same

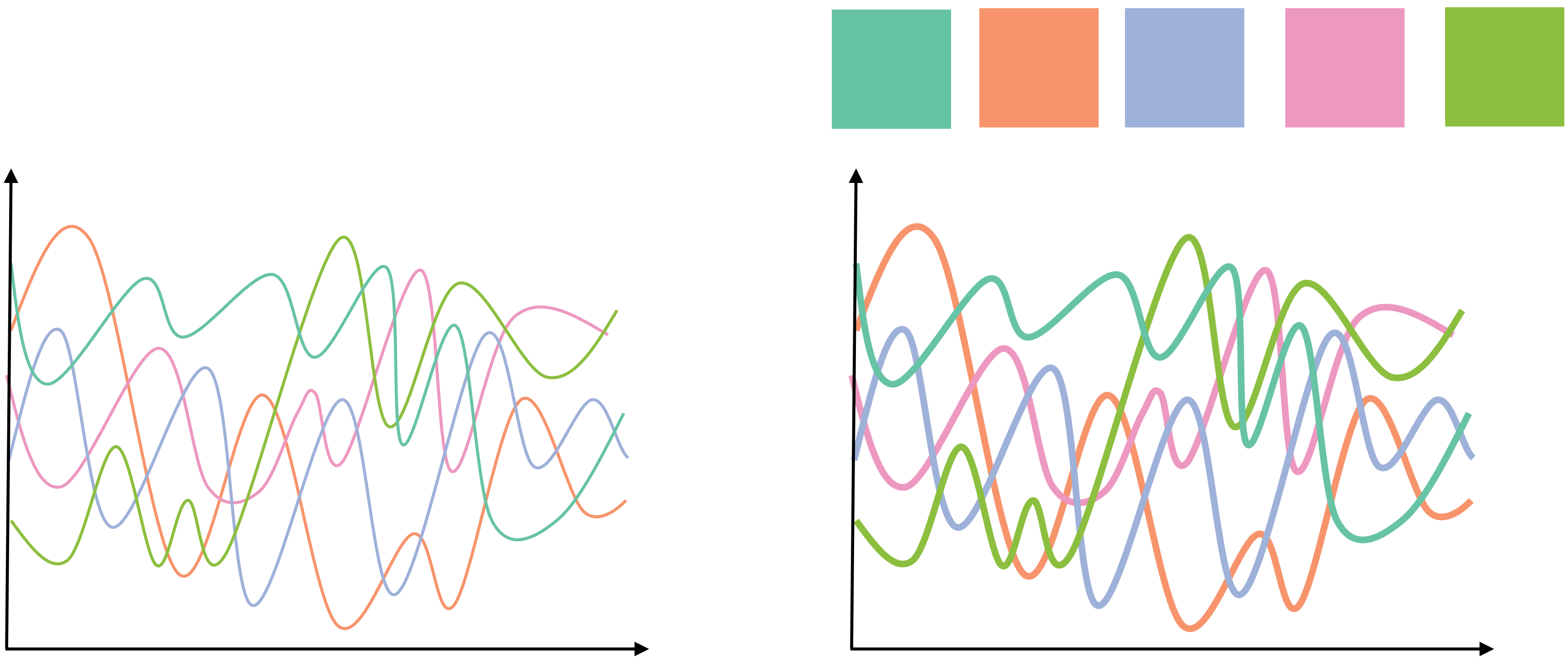
Rectangles with * are the same color

[Wong 2010]

Color mapping

SIEMENS
Ingenuity for life

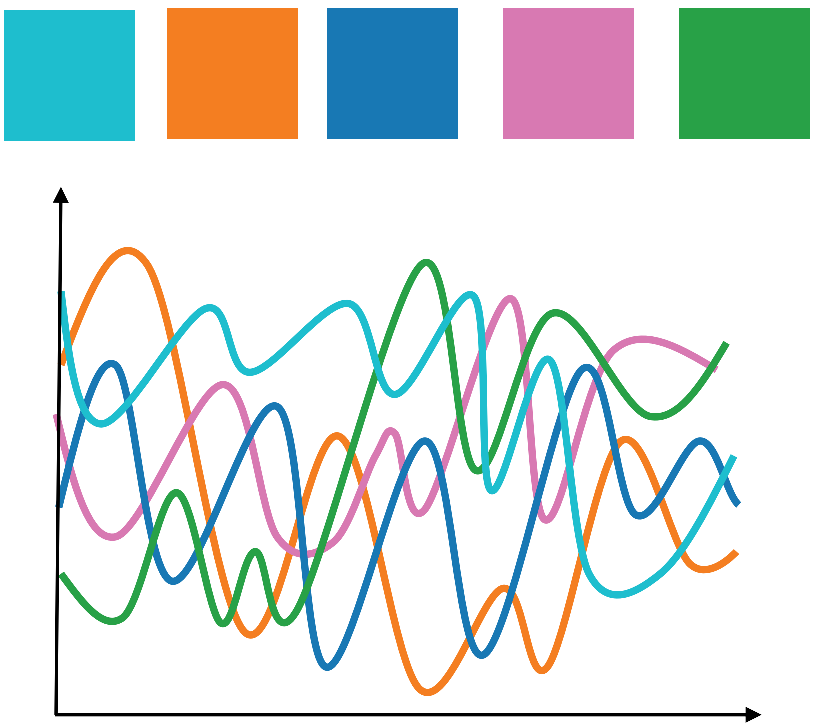
- Things to know when mapping data to colors
  - Size matters



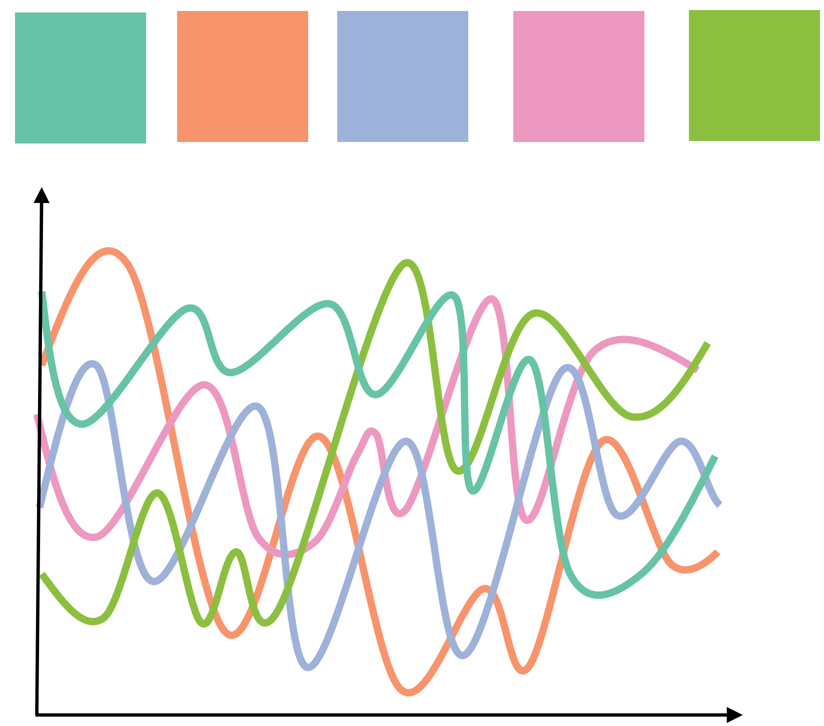
[Stone / van Wijk 12]

Color mapping

- Things to know when mapping data to colors
  - Vary luminance too



Colors with different luminance



Colors with similar luminance

[Stone / van Wijk 12]

Color mapping

- Things to know when mapping data to colors
  - Make sure contrast is high

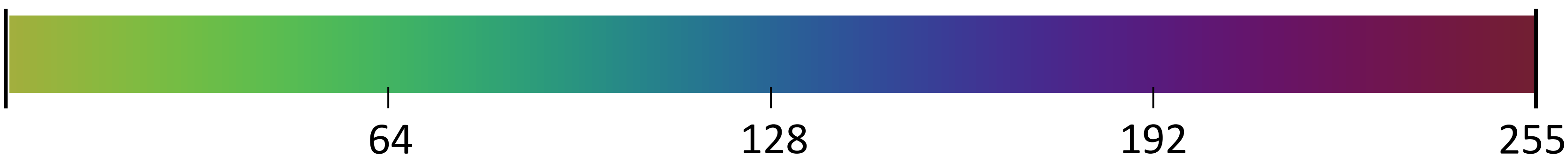

```
fBright = 0.213 * Background.R +
    0.715 * Background.G +
    0.072 * Background.B;

Text.color = (fBright > 0.5) ? BLACK : WHITE;
```

[van Wijk 12]

Color mapping

- Things to know when mapping data to colors
  - Colors are more useful for qualitative statements
  - Do not use color if it is necessary to read out precise values

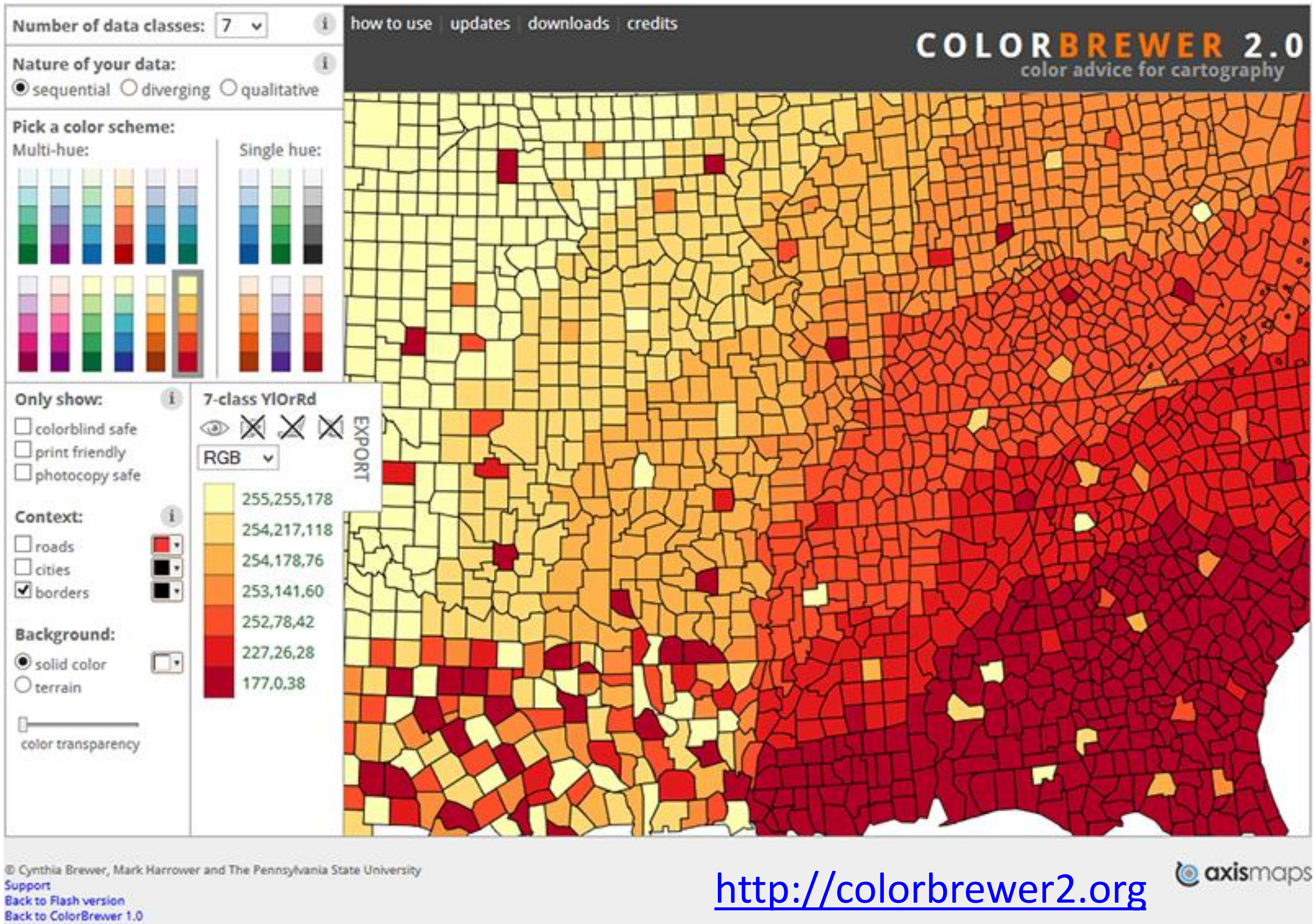


What‘s the value of these colors?

Color mapping

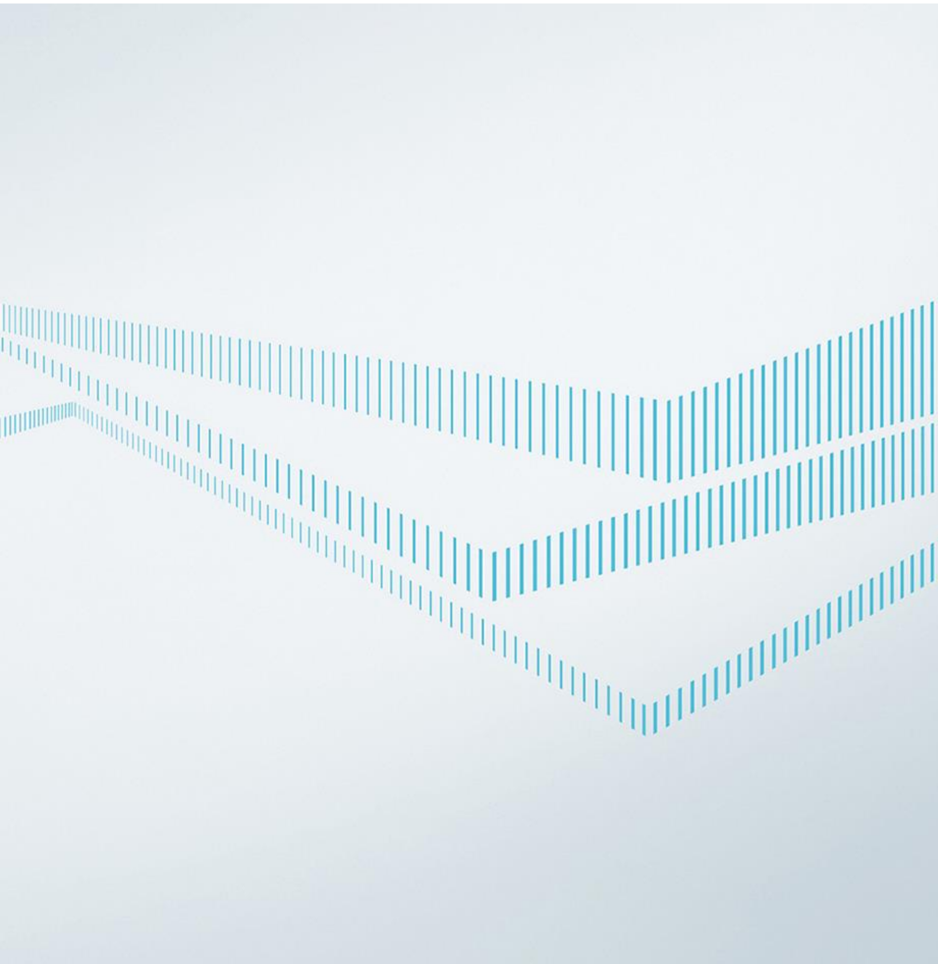
SIEMENS
Ingenuity for life

- Use “good” color maps



http://colorbrewer2.org

Contact information



Dr. Johannes Kehrer

Siemens Technology
T RDA BAM IBI-DE
Otto-Hahn-Ring 6
81739 München, Deutschland

E-mail:
kehrer.johannes@siemens.com

Internet
siemens.com/innovation

Intranet
intranet.ct.siemens.com

Visualization and Visual Analysis of Multi-faceted Scientific Data: A Survey

Johannes Kehrer, *Member, IEEE*, and Helwig Hauser, *Member, IEEE*

Abstract—Visualization and visual analysis play important roles in exploring, analyzing and presenting scientific data. In many disciplines, data and model scenarios are becoming *multi-faceted*: data are often spatio-temporal and multi-variate; they stem from different data sources (multi-modal data), from multiple simulation runs (multi-run/ensemble data), or from multi-physics simulations of interacting phenomena (multi-model data resulting from coupled simulation models). Also, data can be of different dimensionality or structured on various types of grids that need to be related or fused in the visualization. This heterogeneity of data characteristics presents new opportunities as well as technical challenges for visualization research. Visualization and interaction techniques are thus often combined with computational analysis. In this survey, we study existing methods for visualization and interactive visual analysis of multi-faceted scientific data. Based on a thorough literature review, a categorization of approaches is proposed. We cover a wide range of fields and discuss to which degree the different challenges are matched with existing solutions for visualization and visual analysis. This leads to conclusions with respect to promising research directions, for instance, to pursue new solutions for multi-run and multi-model data as well as techniques that support a multitude of facets.

Index Terms—Visualization, interactive visual analysis, multi-run, multi-model, multi-modal, multi-variate, spatio-temporal data.

1 Introduction

Our society is confronted with rapidly growing amounts of scientific data that arise in various areas of science, engineering, and others. Examples are multi-variate and time-dependent climate simulations, computational fluid dynamics, sensor logs, and medical scans. Visualization has proven to be very useful to explore, analyze and gain insight into such data [1]. However, due to the increasing complexity and heterogeneity of scientific data, new sophisticated approaches are needed [2]. Interactive visual analysis is a still new multidisciplinary field that combines analytical and interactive visual methods [3], [4], [5], [6]. Interaction schemes such as linking and brushing enable a powerful drill-down mechanism into the represented information [7].

1.1 Multi-faceted Scientific Data

The integration of abstract data from multiple sources is more common in *information visualization* (InfoVis), for example, when visualizing relational databases [8], [9] or web data [10]. In this survey, however, we focus on challenges that arise from the heterogeneous nature of *scientific data*. Such data are usually given with a strong inherent reference to space and often also time and result from a scientific data acquisition method such as simulation or imaging. When talking about *multi-faceted* scientific data, we consider mainly the following facets: (f1) *spatio-temporal* data that represent spatial structures and/or dynamic processes; (f2) *multi-variate* data consisting of different attributes such as temperature or pressure; (f3) *multi-modal* data stemming from different acquisition modalities (data sources); (f4) *multi-run* data (also called ensemble data) stemming from multiple simulation runs that are computed with varied parameter settings; and (f5) *multi-model* data resulting from coupled simulation models that represent physically interacting phenomena [11] or neighboring climate compartments such as ocean and atmosphere [12]. Current methods for visualization and visual analysis typically address only one data facet (see Fig. 1). In practice, however, we increasingly often find model and data scenarios that are more heterogeneous. A central goal is "to synthesize different types of information from different sources into a unified data representation" [4].

Before going into more details with respect to our survey, examples of different facets of scientific data are discussed together with related challenges for visualization research. Advanced computing allows the simulation of dynamic phenomena on high-resolution grids over large timescales (e.g., global climate models or automotive engine simulations). The visualization and analysis of such spatio-temporal data (f1) is challenging and a lot of research has been dedicated to this facet. A number of surveys give a good overview [13], [14], [15], [16], [17], [18]. Analysts commonly investigate how their data relate to time and space. They want to study changes, compare different points in space and time, and uncover spatio-temporal patterns (e.g., special events or repeated behavior). A frequent goal is to integrate data from multiple time steps in a single image, for instance, by using a linear or cyclic axis to represent time [17], [18].

- *J. Kehrer was with the Department of Informatics, University of Bergen, Norway, and is now with the VRVis Research Center, Donau-City-Str. 1, 1220 Vienna, Austria. E-mail: kehrer@VRVis.at.*
- *H. Hauser is with the Department of Informatics, University of Bergen, Thormøhlensgate 55, P.O.Box 7803, 5020 Bergen, Norway. E-mail: Helwig.Hauser@UiB.no.*

Digital Object Identifier no. 10.1109/TVCG.2012.110.

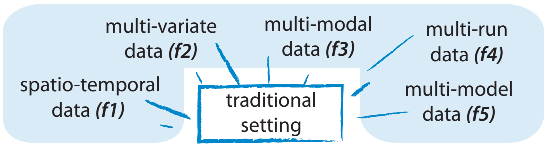


Fig. 1. Multiple facets of heterogeneous scientific data.

Alternatively, different locations or time steps can be shown side-by-side to facilitate comparison [19], [20]. The decision whether to use a 2D or 3D representation is a general question in visualization and usually depends on the task at hand [7], [13], [18]. However, some kinds of data such as volumetric or 3D flow data inherently suggest a 3D representation. Automated analysis methods are often applied in order to *abstract* time-related data characteristics, for example, by computing temporal data trends [21] or statistical aggregates such as mean values or standard deviations [15].

Scientific data often contain multiple attributes per space-time location. The interactive visualization of such multi-variate data (f2) is challenging too [2]. Wong and Bergeron [22] and, more recently, Fuchs and Hauser [23] give comprehensive overviews on the topic. Interesting data subsets (*features*) can often be extracted only when considering multiple data attributes and their relations. Many visual analysis approaches, therefore, integrate computational methods such as statistics or dimensionality reduction [24], [25]. When *fusing* (intermixing) multiple scalar fields in a visualization, one typically has to cope with cluttering and occlusion. Multi-variate data are often analyzed using multiple linked views that support interactive feature specification via brushing [26].

While multi-variate data usually result from one data modality and describe different physical (or other) properties for the same spatio-temporal locations, multi-modal data (f3) stem from multiple data sources. Examples are different medical scans such as computer tomography (CT) or magnetic resonance imaging (MRI), which have to be co-registered first [27], [28]. Another example are data from different numerical models that simulate the same physical object or phenomenon (e.g., different atmospheric or ocean models). Such data can be given on various data grids (e.g., 2D/3D, unstructured or hybrid) with different temporal or spatial resolutions. Accordingly, one challenge is to fuse such multi-modal data in the visualization [23], [29]. An analytic task can be, for instance, to compare data from a climate simulation with observational measurements in order to find errors and to reduce uncertainties [30].

In areas such as climate research [12], [31] and engineering [32], multi-run simulations (f4) are increasingly often performed to study the variability of a simulation model and to understand the model sensitivity to certain control parameters. The simulation is repeated multiple times with varied parameter settings (also called an ensemble simulation). In the resulting data, a collection of values co-exists for the same data attribute at each space-time location [33] (one value for each simulation run). The goals of such a *sensitivity analysis* include [34]: the identification of model parameters that require additional research, which also reduces the output uncertainty; identifying control parameters that are strongly correlated with the simulation output; or finding insignificant parameters that can be eliminated from the model [34]. In the analysis, the data is often aggregated, for example, by computing statistical properties with respect to all runs [31], [35]. While often only the summarized information is analyzed further, it can be useful to integrate and related both multi-run and aggregated data in the visual analysis [36], [37]. It is generally very challenging to depict and analyze a larger number of co-located data volumes; to extract interesting patterns and trends that occur in different runs; to investigate how many of the runs exhibit a certain pattern; or to study correlations between input and output variables of the simulation (compare also to Wilson and Potter [38]).

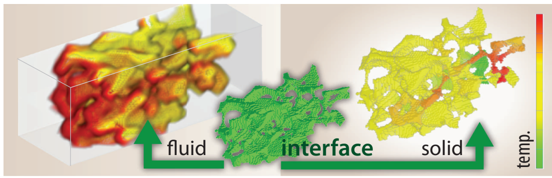


Fig. 2. Fluid–structure simulation of warm water flow though a cooler aluminum foam [37]: The fluid and solid part of the simulation are connected by an interface that relates grid cells sharing a common face and allows the exchange of properties such as heat.

While dynamic flow is traditionally simulated with respect to a rigid (or open-ended) boundary, fluid and solid parts interact during modern *multi-physics* simulations [11]. The solid part, for instance, can be heated or deformed by the surrounding flow. The different data parts are commonly modeled individually on spatially adjoining grids that are connected by a so-called interface (see Fig. 2). During the simulation, the parts can interact with each other via the interface and exchange physical properties such as heat. In the climate system, as another example, components such as atmosphere, ocean, ice, and land interact with each other, as well. Atmosphere and ocean, for instance, exchange through thermal absorption, precipitation and evaporation [12]. To understand such dynamics, models for the different climate components are coupled in the simulation, commonly with additional coupler modules. Creating a coherent visualization from such multi-model scenarios (f5) with two or more interacting data parts (e.g., fluid and structure or atmosphere and ocean) is a challenge for visualization, which has been hardly addressed so far [37]. How can, for instance, feedback and relations between spatially neighboring data parts be investigated?

We focus on the data facets described above, since they are typical examples for scientific visualization. There exist also other interesting modalities such as pictures, video or text. Textual information, for example, can provide semantic context to the data and will become more important in the future, also in scientific visualization (e.g., patient reports or description of genes). A detailed description of these types, however, is beyond the scope of this survey.

1.2 Survey Structure and Contributions

In this survey, we give an encompassing view on the visualization and visual analysis of multi-faceted scientific data. We explain our choice of which facets to focus on and which to only survey in terms of an overview. Based on an extensive literature review, we have identified a number of techniques that are common to all the facets of scientific data. We propose a novel categorization of approaches based on characteristics of these techniques and discuss them with respect to the different facets. We identify mature areas in visualization and visual analysis as well as promising directions for future research.

The visualization of spatio-temporal (f1) and multi-variate (f2) data, for instance, have been broadly investigated, and a lot of good solutions are available. Although these areas belong to the topics discussed here, we only touch them briefly and refer to other good overview articles. As part of our contribution, we add a discussion of newer approaches that came up more recently. The primary focus of this survey, however, is on multi-modal (f3), multi-run (f4) and multi-model (f5) data. Especially multi-run and multi-model data scenarios are relatively new to the visualization community [33], [37], even though these types of data are getting more popular in important application domains such as engineering, climate or multi-physics research. While we aim at putting existing approaches for multi-faceted scientific data into a broad common framework, not all related work can be discussed in such an article. We consider it valuable, however, to present an integration of such a broad spectrum of topics in a joint survey.

The remainder of this paper is organized as follows: before going into detail with respect to related work, some basic notations are clarified in section 2. We also present our classification of approaches. Section 3 discusses important concepts in interactive visual analysis such as coordinated multiple views and the combination of computational analysis and interactive visualization. Section 4 addresses the visualization and visual analysis of spatio-temporal data (f1), and section 5 does this for multi-variate scientific data (f2). The representation, fusion and comparison of multi-modal data (f3) are described in section 6. Section 7 discusses the visual analysis of multi-run data (f4), and section 8 addresses challenges for multi-model data (f5). Finally, an outlook to promising future research and open challenges is given in section 9. In sections 4 to 7, we distinguish between approaches for representation (in terms of visual metaphors), computational analysis and interaction, which also integrate analytical/exploratory procedures.

2 Terminology and Categorization

In this section, some basic notations are clarified first. Next, we propose our categorization of approaches for multi-faceted scientific data with respect to common visualization, analysis and interaction methods.

A continuous data model is very often assumed with *scientific data*, which means that the data can be interpolated between discretely sampled values [39]. In many cases, such data can be denoted as $f_i(\mathbf{x})$ where different data values f_i (e.g., velocity vectors, temperature or pressure values) are measured or simulated with respect to points $\mathbf{x}$ in an n-dimensional domain. The domain (i.e., the independent data dimensions) can be 2D or 3D space, time, but also independent input parameters to a simulation model. *Multi-run* data (f4), for example, stem from a simulation that is repeated multiple times with varied parameter settings, leading to a larger number of co-located data volumes given for the same space-time [12], [33], [38]. In this understanding, the word *multi-dimensional* (f1) refers to the dimensionality of the independent variables, while *multi-variate* (f2) refers to the multitude of dependent variables [22].

Categorization of approaches: Based on a literature review of more than 200 papers that address at least one facet of modern scientific data, we aim at identifying common groups of visualization, analysis and interaction methods. The different categories are described in the following and represented in the columns of Table 1. Similar to the categorization of Bertini and Lalanne [24], the groups cover a broad spectrum, ranging from techniques that mainly address *visual mappings*, i.e., how to represent the data (left in Table 1), to methods that focus on *computational analysis*, i.e., what are the main characteristics or features of the data (right). Additionally, many approaches rely on *interaction* concepts such as linking and brushing, zooming, panning, or view reconfiguration [40]. Although often discussed separately, visualization, interaction and computational analysis clearly are not mutually exclusive. The tight integration of all three levels is a major goal in visual analysis (see Sec. 3).

Approaches for *visual data fusion* aim at intermixing different facets of scientific data in a single visualization, using a common frame of reference. Different time steps can, for instance, be shown along a spatial axis (e.g., as function graphs or on a spiral). According to Fuchs and Hauser [23], multi-variate data can be fused at different stages of the visualization pipeline, for instance, when mapping variables to different visual properties (e.g., glyphs, texture or color), during rendering, or in image stage using layering techniques. For multi-modal data, the different data sources first need to be registered and normalized to each other in order to make them comparable (e.g., resampling to a common grid). The visualization thereby has to be designed carefully to avoid the introduction of artifacts that can be erroneously interpreted as features [29]. Multi-run data can, for example, be represented as families of data surfaces [41] or spaghetti plots [42]. However, it is often not practical to directly visualize such data since they can consist of multiple co-located volumes of spatio-temporal (and often multi-variate) data. Consequently, some approaches compute summary statistics from the multiple runs, which are represent by glyphs or box plots [33], [37], [43].

Comparative visualization investigates the data for similarities and differences [30]. Examples are the comparison of different time steps, spatial locations, data variables or modalities. Dependent on the level of data

TABLE 1
Categorization of techniques for the visualization and visual analysis of multi-faceted scientific data.

	visual mapping	**interactive visual analysis**				**computational analysis**
	visual data fusion	*relation & comparison*	*navigation*	*focus+context & overview+detail*	*interactive feature specification*	*data abstraction & aggregation*
multi-dimensional	maps [13], [14], [92]; Helix glyphs [93]; flow maps [105]; function graphs [70], [71], [72]; Time Histograms [94], [110], [111]; chrono volumes [98]; illustrative techniques [99]; texture-based flow vis. [100]	2-tone coloring [20]; Helix glyphs [93]; juxtaposed views [19], [110]; difference views [107]	search, zooming and panning [40], [54]	2-tone coloring [20]; multi-level focus+context [71]; pixel-based multi-resolution techn. [104]	brushing [21], [70], [71], [95], [113]; transfer functions [110], [111]	aggregation [15], [103], [105]; trends [21]; flow features [82]; clustering [83], [84], [110]; PCA [17], [78], [85]; SOM [89], [90]; KDE [106], [107]; information theory [108]; wavelet analysis [109], [110]
multi-variate	attribute views [22], [50], [67]; color & texture [119]; layering [115], [124], [126]; 2-level volume rendering [127], [128]; glyphs [120], [121], [122], [123], [124], [125]	correlation fields [133]; operators [134]; multiple linked views [9], [26], [29], [73], [74], [76]	grand tour [47]; ScatterDice [46]; ranking & quality metrics [48], [130], [131], [132]	illustrative vis. [115], [116]; outlier-preserving methods [69]; smooth brushing [80]	brushing [9], [50], [74], [75], [112]; multi-dim. transfer func. [114], [115]; machine learning [91], [135], [136]	clustering [68], [130]; data binning [69]; PCA [78]; MDS [86], [87]; SOM [88], [89]; projections [47], [48], [130], [132]; point clouds [129]
multi-modal	resampling [138]; data model [142]; illumination model [143]; multi-volume rendering [128], [139], [143], [144], [145], [146]	difference views [107]; multi-image view [153]; nested surfaces [31], [154], [156]; features [44], [155]	viewpoint selection [49]	cutaway views [147], [139], [49]	transfer functions [143], [144]	registration [27]; mutual information [28]; comparison metrics [148], [151], [152], [133]
multi-run	glyphs & box plots [37], [43], [162], [163], [164]; shape descriptors [164]; families of surfaces [41]; spaghetti plots [35], [42], [165]	aggregated & multi-run data [36], [37], [41], [174]; HyperMoVal [51], [52]	aggregated & multi-run data [36], [37], [41]; parameter space nav. [51], [52]	aggregated & multi-run data [36], [37], [41]; simulation process vis. [173], [174]	trends & outliers [36], [37], [41]; visual steering [172]	overview statistics [31], [35], [36]; projections [41], [51], [170], [171]; operators [33]; PCA [169]; clustering [167], [168], [169]
multi-model	feature fusion across multiple data parts [37]	feature relation across data parts [37]	x	x	feature spec. across data parts [37]	x

abstraction, scientific data can be compared at the image, data or feature level [44]. In the context of information visualization, Gleicher et al. [45] classify comparative visualization according to three categories: 1) juxtaposition, which compares objects side-by-side; alternatively, 2) the data can be overlaid in the same frame of reference (similar to visual fusion [29]); or 3) computed relationships can be explicitly encoded, for example, by showing differences or correlations. Additional interaction techniques such as linking and brushing or repositioning of objects can facilitate visual comparison [45].

Navigation is an important task in visualization and can be done manually [40], [46], automatically [47], or computationally assisted [48], [49]. The user typically explores the data by zooming, rotating or panning. Selecting a good viewpoint is a challenge for volumetric data [49]. In the context of multi-variate data, it is very challenging to find those attributes that describe important data characteristics such as correlations or outliers. Using an overview visualization such as a scatterplot matrix [46], [50], the user can select an interesting attribute combination. Alternatively, views can be ranked by computing quality measures [48]. Navigating the space of input and output variables of a multi-run simulation is highly challenging as well [51], [52].

Focus+context visualization can be generalized as the "uneven use of graphics resources (space, opacity, color, etc.)," where the focus is shown in detail and the context is provided for orientation or navigation purpose [53]. Such methods typically rely on interaction, where the user specifies the data in focus (e.g., by pointing, querying or brushing). While related visualizations aim at seamlessly integrating focus and context in a single view, *overview+detail* techniques spatially separate both concepts (e.g., using juxtaposed views). Cockburn et al. [54] recently provide a survey on the related topics.

Interactive feature specification enables the user to manually select interesting data, for example, via brushing [50]. The resulting markup information can then be used to highlight and relate the selected data, for example, using coordinated multiple views (Sec. 3.1).

Closely related to manual feature specification are automated *data abstraction and aggregation*. Such analysis methods aim at algorithmically extracting meaningful values or patterns from the data, where the main data characteristics are still represented but irrelevant details are suppressed [17]. The abstracted data can then be visualized or analyzed instead of the original one. Many related approaches come from the fields of statistics, data mining, or machine learning [55]. Visual analysis aims at combining such methods with interactive visualizations, where the user can steer the analysis process (see Sec. 3).

By grouping the related work by techniques, we aim at presenting alternatives for addressing different kinds of scenarios. While we give a comprehensive view on the visual analysis and representation of multi-run and multi-model data, only selected examples for spatio-temporal and multi-variate data are discussed together with existing surveys. It should be noted that also other techniques (e.g., knowledge-assisted visualization [56] or clutter reduction) and other factors could be used for this classification, for example, visualization challenges [2], user tasks [7], [39], or application domains. Keller and Keller [57], for instance, present an early task-based categorization for visualization. They illustrate methods for a variety of data types and visualization goals. Amar and Stasko [58], more recently, discuss higher-level analytical tasks such as showing uncertainty, exposing relationships, identifying cause and effect, including metadata, finding multi-variate correlations and constraints, and validating hypotheses. Some of these tasks are closely related to individual techniques discussed also here.

3 Exploratory Data Analysis, Visual Data Mining, and Visual Analytics

Visual analytics is the interdisciplinary science of analytical reasoning facilitated by interactive, visual and analytical methods [3], [4], [5], [6]. Since automated analysis methods only work reliably for well-specified problems, the idea is to combine such approaches with interactive visualization. Visualization can then, for example, support the specification of parameters at different steps of a data mining algorithm [59]. By interactively and visually exploring the original data as well as derived properties, analysts should be enabled to [4]: detect the expected and discover the unexpected; draw conclusions and generate hypotheses based on the visual information; reject or verify hypotheses; and communicate and present the results of the analytical reasoning process.

While statistical tools utilize static visualization mainly for presentation purposes (confirmatory analysis), Tukey suggests in his seminal work on *exploratory data analysis* [60] to also support direct interaction with the data. Some of the early works in InfoVis were inspired by considerations from statistics [61], [62], [63]. The analysis often follows Shneiderman's *information seeking mantra* [7]: "overview first, zoom and filter, then details-on-demand." In later work, Shneiderman [59] compares the different philosophies behind exploratory data analysis (used for hypothesis generation) and statistical hypothesis testing. The latter requires a hypothesis beforehand in order to work. Also, it is challenging to identify features that are not anticipated prior to the analysis. The author thus suggests the combination of data mining and visualization, where users should be able to express their interest in the data and specify what they are looking for (e.g., outliers or correlations).

If the data are too large and complex to be represented directly, the application of automated data abstraction techniques is often necessary. Keim proposes an according extension to the information seeking mantra for visual analytics [5]: "Analyze first, Show the Important, Zoom, filter and analyze further, Details on demand." While *visual data mining* [64], [25] mainly focuses on the integration of data mining[1] into the visualization, visual analytics [4], [5], [6] aims at integrating other methods of analytical reasoning as well (e.g., cognitive, perceptual or decision science). Excellent overviews on visual data mining are given by Keim [64], Keim et al. [25], and de Oliveira and Levkowitz [66]. Bertini and Lalanne [24] more recently survey the integration of visualization and automated analysis in knowledge discovery. Based on the degree to which such methods are combined, the authors categorize solutions into 1) computationally enhanced visualizations, 2) visually enhanced mining, and 3) integrated visualization and mining (compare to a similar categorization by Keim et al. [25]).

In the following, typical concepts for interactive visualization and computational analysis are briefly discussed, namely coordinated multiple views (Sec. 3.1) and automated data abstraction (Sec. 3.2), respectively.

3.1 Coordinated Multiple Views

The concept of coordinated multiple views originates in the InfoVis community and has been steadily developing over the last two decades (see Roberts [26] for an overview). Different data variables are shown, explored and analyzed in multiple linked views that are utilized side-by-side. The views include histograms, scatterplot matrices [46], [50], parallel coordinates [67], [68], [69], or function graphs [70], [71], [72]. Data can be interactively selected (*brushed* [50]) in a view, the related data items are instantly highlighted in all *linked views* (compare to Polaris/Tableau [9] or the XmdvTool [73], for example). Logical combinations of brushes across multiple views support the specification of complex features, for example, in a hierarchical feature definition language [74]. In cross-filtered views [75], as another example, brushing filters between pairs of views can be enabled/disabled and the data are filtered, accordingly. Relationships between different variables can thus be explored, also across multiple datasets. Several visual analysis frameworks support the computation of new data attributes from existing ones, which facilitates the investigation of features [3], [9], [26], [74], [75].

Interaction and flexibility of the application are both crucial for visual analysis. The user should be able to query data in many different ways and quickly change what data portions are shown and how they are represented [9], [40]. ScatterDice [46] is such an example where the user explores multi-variate data using scatterplots. A scatterplot matrix [50] gives an overview of the possible axis combinations in a plot and is used for navigation. Transitions between the scatterplots are then performed using animated 3D rotations. Features can be explored and iteratively refined via brushing.

SimVis [74], WEAVE [76] and PointCloudXplore [77] are just three examples of visual analysis systems for scientific data. Such frameworks link *attribute views* such as scatterplots or parallel coordinates with *3D views* for volumetric data (usually given on grids over time). This combination enables the analyst to investigate brushed features also in the spatial context (compare to the three typical patterns of visual analysis of scientific data [78], i.e., feature localization, multi-variate analysis, and local investigation). Instead of a binary selection information, some systems integrate a fractional *degree-of-interest* attribution ($DOI_j \in [0, 1]$ for each data item j, compare to the DOI information in generalized fisheye views [79]). With *smooth brushing* [80], a transition can be specified around the main region of interest, where the DOI information gradually changes (compare also to ramped brushing in the XmdvTool [73]). The DOI values are then used in all

1. Data mining denotes the algorithmic extraction of valuable patterns and models from data. It is part of a more general process of *knowledge discovery in databases* (KDD), which also includes steps such as data preparation, selection and cleaning [65].

linked views to visually discriminate interesting features (focus) from the rest of the data (context), leading to a *focus+context visualization* [53], [71], [74], [81].

3.2 Automated Data Abstraction

Typical (semi)automated analysis methods that are often combined with InfoVis techniques include [24], [25]: *data reduction* via sampling or algorithmic feature extraction [82]; *clustering* [68], [83], [84] where data items are grouped by similarity; and *dimensionality reduction* that aims to reduce the data dimensionality while maintaining the higher-dimensional data characteristics. Dimensionality reduction approaches include: principal component analysis [17], [78], [85] (PCA), which transforms multi-variate data into an orthogonal coordinate system that is aligned with the greatest variance in the data; multi-dimensional scaling [86], [87] (MDS), where higher-dimensional data items are mapped into a lower-dimensional space while preserving the dissimilarities between the items;[2] and self-organizing maps [88], [89], [90] (SOM) which represent an unsupervised learning method that reduces the data dimensionality and also provides a classification of the data. An issue with dimensionality reduction approaches is, however, that it can be hard to mentally relate the derived attributes to the original data. One solution can be to analyze both side-by-side in a multiple views framework with linking and brushing (see Oeltze et al. [78], for instance).

Ma [91] suggests to go a step beyond visual data mining by integrating machine learning into the analysis process. Such methods could learn from previous analysis sessions and input data, and abstract away many details of the utilized algorithms, for instance, using case-based reasoning (compare to an infrastructure supporting knowledge-assisted visualization [56]). Only high-level decisions are then left to the user by providing an "intelligent interface" for the visual analysis [91].

4 Multi-dimensional Data

Multi-dimensional data such as time-varying 3D measurements and simulations are ubiquitous in disciplines such as medicine, climate research, or engineering. Being able to understand time-related developments allows one to "learn from the past to predict, plan, and build the future" [18]. When visualizing the data, time and space can be treated "just" like any other data attribute using parallel coordinates, scatterplots, or other information visualization techniques [6], [18]. In many applications, however, the independent dimensions of time and space have a semantic meaning and often play a central role in the data. Accordingly, there has been a lot of work in related fields such as cartography [92] or geovisualization [13], [14]. A number of useful reviews have been published on the visualization and analysis of spatio-temporal data [13], [14], [15], [16] as well as time-dependent data [17], [18]. According to Andrienko et al. [14], approaches for spatio-temporal data can be categorized into the visualization of raw data, computed summaries, or automatically extracted features.

This section gives a brief overview on methods that address mainly the spatial and/or temporal characteristics of scientific data. Such data are often multi-variate as well, which is elaborated in further detail in section 5. Related surveys are discussed together with selected examples. In this context, the following subsections address the visual representation (Sec. 4.1), computational analysis (Sec. 4.2), and interactive methods (Sec. 4.3) for spatial and temporal data.

4.1 Representation of Multi-dimensional Data

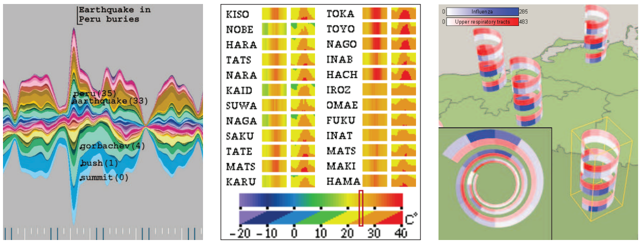
Aigner et al. [18] give a systematic view on the visualization of time-oriented data. In their categorization, they consider different characteristics of the time axis such as temporal primitives (discrete points vs. time intervals) or the structure of time (linear vs. cyclic vs. branching time). These considerations are important when designing a visual analysis system, since they address the data validity and the possible relations among temporal primitives [18]. Common approaches for time-varying data include automatic animations or interactive visualizations. The latter, for instance, show the data at different time steps (e.g., juxtaposed views) or along a common time axis (e.g., function graphs or spirals). Selected visualization methods for spatio-temporal data are discussed in the following.

Time-varying data can be represented in a single view by showing them with respect to a linear or cyclic time axis [18]. The latter supports the comparison of different points in time and the analysis of recurring patterns such as seasonal trends. An example for such an approach are Helix glyphs [93] that can be placed on a geographic map (see Fig. 3c). The "tunnel view" at the bottom of the figure reveals hidden information by increasing the ribbons height and radius for each time step. The ThemeRiver [72] is an example visualization that uses a linear time axis. Thematic changes in large document collections are depicted, where the number of occurrences per topic is represented as the width of the corresponding river band (see Fig. 3a). The Time Histogram [94] is another example showing consecutive 1D histograms of the data for every time step.

Two-tone coloring [20] is an example for an integrated overview+detail technique, which enables the compact representation of many time series (details) in an overview visualization (see Fig. 3b). By showing the data values of each time series as a combination of two colors, the actual values can be read out more precisely as compared to using a continuous color map. Other approaches [70], [71], [74], [95] use color and saturation for a focus–context discrimination and are discussed in conjunction with interactive feature specification (Sec. 4.3).

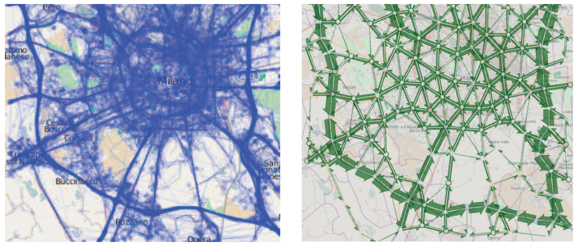
Spatio-temporal data have additional characteristics, for instance, that "near things are more related than distant things" (Tobler's first law of geography [96]) or that events can happen at different spatio-temporal scales [14]. Geospatial data are often shown on cartographic maps, following a set of well-established con-

2. Since MDS also maintains the higher-dimensional structure of the data, it is well suitable for subsequent clustering.



(a) (b) (c)

Fig. 3. Linear vs. cyclic time: (a) ThemeRiver [72] for analyzing temporal trends in document collections (image courtesy of S. L. Havre, © 2002 IEEE). (b) Visualization of temperature variations using a continuous and two-tone color map [20], respectively. The latter represents the data value of each time series as a combination of two colors (image courtesy of T. Saito, © 2005 IEEE). (c) Helix glyphs [93] for analyzing cyclic temporal patterns for two diseases (image courtesy of C. Tominski, © 2005 IEEE).



(a) (b)

Fig. 4. Visualization of car movement data [105]: (a) Original movement trajectories are drawn using opacity. (b) Trajectories are aggregated and shown in a flow map (images courtesy of N. Andrienko, © 2011 IEEE).

ventions [13], [14], [92]. In this context, the visualization of textual information is highly relevant, for instance, by placing labels that give semantic information to the data. Many approaches for spatio-temporal data support navigation techniques such as zooming and panning [54].

Ma and Lum [16] discuss techniques that support the efficient rendering of time-dependent *volumetric data*, for example, data compression, automated feature extraction, hardware acceleration, or parallel rendering. Jankun-Kelly and Ma [97] study the (semi)automatic generation of a single or multiple transfer functions, which capture important structures in time-varying volumetric data such as regular, periodic or random pattern. The generated transfer funtion(s) can then be used for batch-mode rendering, for instance. Woodering and Shen [98] propose chronovolumes that fuse multiple time steps in a single image using color composition techniques. Joshi and Rheingans [99] present illustrative techniques that are inspired by depictions of motion in comics. Examples are speedlines, flow ribbons, or silhouettes showing the previous positions of an object.

Besides the visualization of time-dependent scalar volumes, also the visualization of time-dependent vector fields is important in many areas. Such approaches for *flow visualization* can be classified into [82], [100], [101]: 1) direct flow visualization such as color coding or arrow plots; 2) dense, texture-based approaches using, for instance, spot noise, line integral convolution, or texture advection; 3) geometric flow visualization depicting geometric objects that are extracted/computed from the flow such as streamlines, stream surfaces, streaklines, or pathlines; 4) feature-based techniques that are based on the extraction of relevant structures such as vortices or shock waves; and 5) partition-based flow visualization that subdivides the domain with respect to certain flow characteristics. While the first three categories depict basic quantities of the flow, the later two provide a more abstracted view on the data.

4.2 Analysis in Multi-dimensional Data Visualization

The approaches presented in the previous section usually reach their limits when representing larger amounts of data with several million entries, for instance. For such data, (semi)automated data reduction and abstraction techniques need to be applied first, which transform the data into a compressed but still representative form [5]. Andrienko and Andrienko [15] give a systematic overview on the visual analysis of spatial and temporal data. Aigner et al. [17] discuss approaches for time-oriented data where visual and analytical methods are combined. Many approaches for temporal data abstraction come from the field of data mining (see Fu [102] for a recent survey). Examples include clustering, principal component analysis, or wavelet analysis.

In order to reduce the data complexity or visual cluttering, spatial and/or temporal aggregation is often applied (see López et al. [103] for an overview). With such an approach, data items sharing the same spatio-temporal domain are summarized and depicted instead of the individual data values. According to Andrienko and Andrienko [15], data aggregation can be done either by calculating data characteristics (e.g., the sum, arithmetic mean, or variance) or by grouping techniques such as clustering or binning. Aggregation techniques, however, need to be applied with care to preserve important information such as outliers [69].

Hao et al. [104] use pixel-based techniques to visualize time series data at multiple levels of aggregation, based on importance values per data interval. Andrienko and Andrienko [105] visualize movement data by combining data aggregation with flow maps. The spatial domain is subdivided into appropriate areas, based on significant points in the movement. Aggregated trajectories with common start and end points are shown with arrows (see Fig. 4b). Willems et al. [106] propose a visualization approach based on the convolution of dynamic movement data with a kernel, where the resulting density field is visualized as an illuminated height map. Daae Lampe et al. [107] propose interactive difference views based on kernel density estimates (KDEs). Quantitative differences between different categories (or bins) of aggregated data are analyzed using juxtaposed views.

Nocke et al. [83] discuss visualization techniques for clustered climate data such as the ThemeRiver [72] or Cluster Calendar View [84]. The latter approach, for instance, groups time series data over a certain period (e.g., month or day) into clusters. The clusters are then visualized using function graphs and also encoded in color in a calendar-like representation. As a result, the frequency of occurrence of each cluster can be seen as well as the daily trends and patterns.

Dimensionality reduction techniques typically aim at reducing the data dimensionality while preserving the higher-dimensional characteristics. Aigner et al. [17] discuss the integration of PCA into the visualization of time-dependent climate data (compare also to Müller et al. [85]). Self-organizing maps [88] (SOM) can be seen as a combination of dimensionality reduction and clustering. Andrienko et al. [90] apply this approach for analyzing spatio-temporal data from two complementary perspectives: as spatial situations in different time units (space-in-time SOM) and as profiles of temporal changes at different places (time-in-space SOM). For each perspective, a SOM matrix display provides an overview of data objects arranged by similarity. The matrix is linked to views such as spatial maps, function graphs, and periodic pattern views, which enable the investigation of spatio-temporal patterns.

Concepts from *information theory* can be applied to automatically extract distinctive structures in the data. Jänicke et al. [108], for example, compute the local statistical complexity in order to identify regions with different temporal behavior than the rest of the field. The measure assesses the amount of information from the local past that is necessary to predict the local future. In later work, the same authors utilize *wavelet analysis* for exploring climate variability changes [109]. Clustering techniques based on mutual information are applied, amongst others, in order to identify coherent structures in the data. Similarly, Woodering and Shen [110] apply wavelet transformation to time-dependent volume data. The resulting multi-resolution data representation is clustered and visualized in a spreadsheet [19]. Here, multiple Time Histograms are shown that also support linking and brushing (compare to Akiba et al. [111]).

4.3 Interactive Methods for Multi-dimensional Data

While computational analysis methods typically rely on well-defined problems, certain data features are difficult to describe mathematically or hard to anticipate prior to the analysis. Consequently, many applications support interactive feature specification using brushing or querying techniques. Additionally, computational methods can be applied on-demand to facilitate the analysis.

The TimeSearcher [95] is especially designed for the visual analysis of time-dependent data using Time Boxes or angular query widgets. The latter are applied for selecting time series that have a similar slope on a sequence of time steps (compare to angular brushing for parallel coordinates [112]). Konyha et al. [70] introduce line brushes that select function graphs, which intersect a line segment drawn in the view. Akiba et al. [111] utilize a Time Histogram [94] to specify transfer functions for time-varying volume data.

Jern et al. [113] propose a coordinated multiple views system for exploring spatio-temporal multi-variate data. Cartographic maps are linked with attribute views such as parallel coordinates that also support brushing. Interactive feature specification in multiple linked views is also an integral part of the SimVis framework [74]. Oeltze et al. [78] study the integration of both correlation analysis and PCA into the visual analysis of perfusion data. Parameters describing the temporal perfusion characteristics are extracted and analyzed together with the principal components using linking and brushing. The approach is applied in the diagnosis of breast cancer, ischemic stroke, and coronary heart disease.

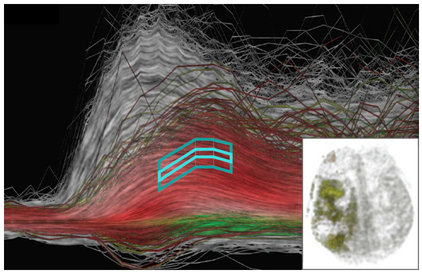


Fig. 5. Visual analysis of perfusion data acquired after an acute ischemic stroke [71]: brain tissue with reduced and delayed perfusion is selected using similarity-based brushing in the function graph view. The related tissue at risk is shown in the spatial context (inset).

Kehrer et al. [21] use brushing of derived temporal characteristics such as linear trends and signal-to-noise ratios for the steered generation of hypotheses in climate research. Spatio-temporal regions in the atmosphere are identified which can act as sensitive and robust indicators for climate change. This work is based on an extension of SimVis, which enables the interactive depiction of large amounts of time series as function graphs together with advanced brushing techniques [71]. Function graphs that are similar to a pattern sketched by the user can be interactively selected (see Fig. 5). Also, transfer functions are applied for visual clutter reduction by mapping the number of function graphs per pixel to the pixel's luminance (compare to Johansson et al. [68]). Using aggregation techniques (frequency binmaps [69]), the responsiveness of the system can be maintained, even when interacting with large amounts of time series.

5 Multi-variate Scientific Data

The multi-variate characteristics of scientific data are often of special interest, typically in combination with their spatial and/or temporal reference. When investigating, for instance, the fronts of a storm [114] or environmental phenomena such as the El Niño [109], multiple data attributes and their interrelation need to be considered. Johnson [2] identifies the visualization of multi-variate scientific data (also referred to as multi-field data) as one of the top challenges in scientific visualization. Comprehensive surveys on the topic are given by Wong and Bergeron [22] as well as Fuchs and Hauser [23].

In the following subsections, the representation, computational analysis, and interactive methods for multi-variate scientific data are discussed.

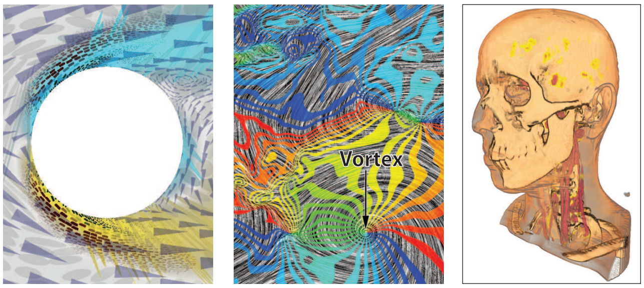
5.1 Representation of Multi-variate Scientific Data

Multi-variate patterns such as correlations or outliers can often be directly perceived when plotting the data in attribute space, for instance, using scatterplot matrices or parallel coordinates [22]. Such attribute views, however, are less able to convey spatial relationships of the data. Another challenge is which of the many data variables to show in order to not miss important patterns (e.g., using quality metrics [48] as discussed in Sec. 5.2). Alternative methods for spatial data such as direct volume rendering typically have difficulties encoding multi-variate characteristics. When fusing multiple scalar fields in a single visualization (e.g., using glyphs or layering techniques), one often has to cope with cluttering and occlusion. Different portions of the data can be represented using a set of visual styles (e.g., focus+context or illustrative visualization [53], [115], [116]). Such feature-based approaches, however, typically rely on segmentation information, which can be specified, for instance, interactively via brushing or transfer functions [74], [76], [114], [115].

Multiple data values can be simultaneously represented in an image using *preattentive visual stimuli* such as position, width, size, orientation, curvature, color (hue), or intensity [117], [118]. These features are rapidly processed by the low-level visual system and can thus be used for the effective visualization of large data. Special care is required, however, if several such stimuli are combined—the result may not be preattentive any more. Healey and Enns [119] propose simple texture patterns and color to visualize multi-variate spatial data. Different data attributes are encoded in the individual elements of a perceptual texture using equally distinguishable colors and texture dimensions such as element density, regularity and height. Groups of neighboring elements form texture patterns that can be analyzed visually.

Glyphs are a powerful way of encoding multi-variate data, which is often used in information visualization (e.g., star glyphs, stick figures, faces, see Ward [120] for an overview). Different data variables are represented by a glyph using a set of visual stimuli such as shape, size or color. Relations between the data variables can be directly perceived and compared, often also in the spatial context when using a hybrid visualization [23]. It should be noted that some visual cues and/or their relationships can be easier perceived than others [117], [118], [120]. An effective glyph visualization should, therefore, carefully chose and combine the utilized visual properties.

Ropinski and Preim [121] propose a perception-based glyph taxonomy for medical visualization. The authors categorize glyphs according to 1) preattentive stimuli such as glyph shape, color and placement, and 2) attentive visual processing, which is mainly related to the interactive exploration phase (e.g., changing the position or parameter mapping of a glyph). Additional usage guidelines are proposed, for instance, that glyph shapes should be perceivable unambiguously from different viewing directions. Kindlmann [122], for example, uses superquadric glyph shapes that fulfill this criterion. Ropinsky and Preim [121] also suggest that the mapping of data variables to glyph properties should focus the



(a) (b) (c)

Fig. 6. Visualizing multi-variate data with layering techniques: (a) Combining glyphs, elongated ellipses, and color to show a total of nine variables [124] (image courtesy of R. M. Kirby, © 1999 IEEE). (b) Filigree graphics encoding flow direction combined with a streamline texture [126] (image courtesy of P. C. Wong). (c) Different visual styles representing semantic information [115] (image courtesy of P. Rautek, © 2007 IEEE).

user's attention and emphasize important variables. Lie et al. [123] propose additional guidelines for glyph-based 3D visualization with respect to the different stages of the visualization pipeline. It should, for instance, be possible to perceive each visual glyph property independently (or to mentally reconstruct the depicted data values [121]). The authors discuss further design aspects of glyph-based 3D visualization such as depth perception and visual cluttering (e.g., using halos to discriminate overlapping glyphs).

Since glyphs are typically not placed in a dense way, the space between them can encode additional information. Kirby et al. [124], for example, use concepts from painting for visualizing 2D flow. They combine different image layers with glyphs, elongated ellipses, and color (see Fig. 6a). Treinish [125] visualizes multi-variate weather data using color contouring on vertical slices and isosurfaces that represent cloud boundaries. At user-defined locations, the wind velocities are represented by a set of arrow glyphs. Additional streamlines following the wind direction are seeded at each arrow.

Multiple scalar fields can be fused in a visualization by using *2D or 3D layering*. Wong et al. [126], for example, encode different climate variables by overlaying multiple see-through layers using opacity modulation, filigree graphics, or 2D height maps. Flow features such as critical points or vortices are highlighted using enhanced color maps (see Fig. 6b). Two-level volume rendering [127], [128] considers segmentation information when visualizing 3D medical data. Different rendering techniques such as maximum intensity projection, direct volume rendering, or non-photorealistic techniques are combined, based on the segmentation. Viola et al. [116] propose similar focus+context techniques, which integrate dense as well as sparse rendering styles. Illustrative visualizations such as cut-away and ghosted views can thus be generated automatically. In later work, Rautek et al. [115] propose semantic layers for illustrative volume rendering, where the mapping of data properties to visual styles can be specified using natural domain language. In Fig. 6c, for instance, contours represent areas of high density; yellow and red highlight regions with low and very low distances to vessels, respectively.

5.2 Analysis in Multi-variate Data Visualization

Finding interesting structures in multi-variate data is a typical challenge for computational analysis, especially in cases of many variables. Such methods, however, often neglect the semantic meaning of the independent dimensions of space and time. Example methods are aggregation techniques, clustering, regression and outlier analysis [55]. Dimensionality reduction such as PCA or MDS is often applied when analyzing multi-variate data. The data are projected to a lower-dimensional space while preserving their meaningful structures and relationships (see Sec. 3.2). Jänicke et al. [129], for instance, transform multi-variate data onto a 2D point cloud, where data items with similar characteristics are located close to each other. The authors compute a tree where the Euclidean distance between multi-variate data items is minimal. The tree structure is then utilized when transforming the data to 2D. Additional information is encoded using color and point size, and interesting structures can be selected via brushing.

Another analysis challenge with multi-variate data is finding those attributes that represent the most important data characteristics. The grand tour method [47], for instance, automatically generates a sequence of orthogonal projections onto a 2D subspace, which can be used in an animation. Seo and Shneiderman [130] introduce the rank-by-feature framework, where low-dimensional projections such as scatterplots or histograms are ranked based on user-selected criteria (e.g., correlation or entropy). A triangular matrix represents the possible combinations of data variables in a scatterplot and encodes the corresponding ranking score in color. This supports the user to select interesting views on the data. Scagnostics [131] are measures that characterize the point distribution in 2D scatterplots and can be used to detect anomalies in shape, density and trend. Tatu et al. [132] recently propose further quality measures for scatterplots and parallel coordinates that are utilized for ranking these views. Quality metrics can also be used for reordering axes in parallel coordinates in order to find visual structures such as correlations or clusters [48]. In this context, Ward [120] discusses measures for ordering the data attributes that are represented by a glyph.

In order to deal with visual cluttering in parallel coordinates, Johansson et al. [68] utilize clustering and high-precision textures. The number of primitives per pixel are mapped to the pixels' luminance by applying user-defined transfer functions. Each cluster is encoded in color and local outliers are visually enhanced. Novotný and Hauser [69] propose an interactive focus+context visualization for parallel coordinates. The data between each pair of adjacent axes is aggregated in a 2D binmap. Clustering and outlier detection are then applied on the aggregated data in order to show general data trends while preserving outliers. Since these methods are applied in image space (2D binmaps) instead of the original data space, the approach is suitable for interactively rendering larger amounts of data.

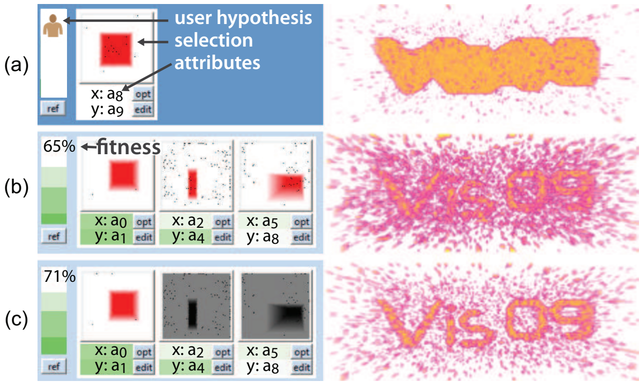


Fig. 7. Combining visual analysis with machine learning [136]: (a) The user specifies an initial hypothesis as a selection on attributes a_8 and a_9. (b) A search algorithm finds alternative hypotheses that explain the same feature but are described by selections on different attributes. The best resulting hypothesis has 65% fitness, where attributes a_0 and a_1 are very important (dark green). (c) The fitness increases when the user deactivates selections (gray) on less important attributes (images courtesy of R. Fuchs, © 2009 IEEE).

Other methods for volume visualization highlight the computed differences or correlations between multiple data variables. Sauber et al. [133], for example, introduce multifield-graphs that give an overview of the correlation between different scalar fields. The user is guided to interesting correlations, which can then be inspected in detail using direct volume rendering. Woodring and Shen [134] propose volume shaders to compare multiple time-dependent scalar volumes by using consecutive algebraic set operators as well as numerical or statistical operators. For interaction and visualization of the resulting volume tree they utilize image spreadsheets (compare to Jankun-Kelly and Ma [19]).

5.3 Interactive Methods for Multi-variate Data

As discussed in section 3.1, multi-variate data are often analyzed in coordinated multiple views. When combining brushing in attribute views with linked 3D volume visualizations, the specified features can be explored in their spatial context too [74], [76], [77]. In the following, we discuss the combination of interactive feature specification with supervised machine learning.

Kniss et al. [114] propose transfer functions for specifying multi-variate features in meteorological data. Interesting data subsets can be selected both in volume and transfer function space using a set of direct manipulation widgets. Tzeng et al. [135] propose an intelligent painting interface that supports the higher dimensional classification of volume data. Regions can be marked directly on sample slices in the volume space, and the data are then classified automatically using a supervised machine learning approach. The training data can then, for instance, be used for classifying other data with similar characteristics. Ma [91] discusses further applications of machine learning and visualization such as flow feature extraction and feature tracking. Fuchs et al. [136] combine interactive feature specification via brushing with machine learning. Using a heuristic search algorithm, the most suitable hypotheses for a user-specified feature can be identified out of a large search space according to different fitness criteria (see Fig. 7).

6 Multi-Modal Data

Data stemming from different acquisition modalities are common in many physical sciences including climate research, geology, and astronomy [137]. A simulation model can be validated, for instance, by comparing it to the output of another model or measurement data [30]. While multi-variate scientific data are typically sampled or computed for the same spatio-temporal locations, this need not be the case with multi-modal data. Such data can be given on various types of grids (e.g., 2D/3D, unstructured or hybrid) with different time steps and/or spatial resolutions. Accordingly, the different modalities often need to be fused in the visualization, for instance, by resampling them to a common grid [138]. In the medical domain as well, data increasingly often stem from different measurement techniques such as CT, MRI, or ultrasound data. Combining such modalities in a visualization can account for the strengths and weaknesses of the individual ones [139]. The different modalities often need to be registered and normalized to each other in order to make them comparable (see Ardeshir Goshtasby [27] for an overview).

6.1 Representation of Multi-Modal Data

Typical challenges for multi-modal data are the rendering and registration of multiple intersecting scalar volumes, which are possibly sampled at different locations. Similar to multi-variate data, such data can be fused at different stages of the visualization pipeline [23]: 1) during data filtering and visualization mapping, for instance, by reducing the data to relevant features or by resampling to a common grid; 2) during accumulation in the rendering stage; or 3) in image stage, for example, using layering techniques.

Multi-block flow visualization is an example where simulations are performed on multiple grid types with different resolutions [140]. When visualizing the data, these blocks are commonly intermixed at the data level by constructing a common grid. As a result, multi-variate visualization techniques as described in section 5 can be applied. We find, for instance, multi-variate rendering approaches for non-uniform grids [141] or hybrid and non-structured grids [81]. Treinish [138] discusses the data fusion of scattered meteorological observations, for example, by constructing a common grid using Delauney triangulation or resampling to a regular grid. The same author proposes a function-based data model [142] that provides uniform access to different modalities. The model adjusts to the data structure and the way data are processed. Consequently, the same operations can be applied to multi-modal data without resampling to a common mesh or unnecessary interpolation.

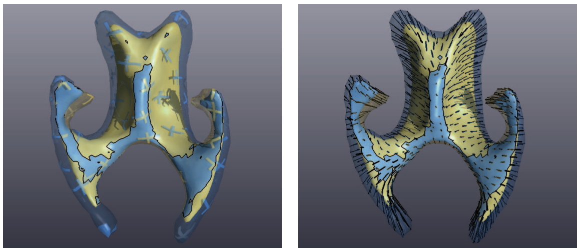
Cai and Sakas [143] use the different data modalities as parameters to a multi-volume illumination model (in the visualization mapping stage). As an alternative, the same authors combine color and opacity from different volumes during accumulation, where each volume has it's own transfer function [143]. Similar to that, Grimm et al. [144] fuse multiple intersecting volumes during the rendering by using V-objects, which represent different visual properties of the individual volumes (e.g., illumination, transfer function, region of interest, and transformation). The data are rendered efficiently in software using multi-threading and a brick-wise ray traversal scheme as well as mono-volume rendering for non-intersecting areas. Plate et al. [145] present a framework for rendering large, arbitrarily oriented volumes using slice-based rendering on the graphics hardware. Their approach supports out-of-core techniques and volumes given at multiple resolutions. Lindholm et al. [146] more recently introduce a region-based scene description for GPU-based volume rendering. Using binary space partitioning, the depth information of the intersecting geometry is stored in a view-independent way and time-consuming depth sorting can be avoided.

Beyer et al. [139] present a system for preoperative planning of neurosurgical interventions. Similar to two-level volume rendering [128], the authors render segmented multi-modal data directly on the GPU. Burns et al. [147] combine tracked 2D ultrasound data with illustrative techniques for volume visualization such as flexible cutaways and importance-driven shading. Context information occluding the object of interest can thus be removed and features can be enhanced (compare to importance-driven rendering by Viola et al. [116]).

6.2 Analysis in Multi-Modal Data Visualization

Registration [27] is a typical first step when working with multi-modal data (e.g., using mutual information [28]). A common analysis task is the comparison of multiple data modalities for similarities and differences [30]. According to Verma and Pang [44], scientific data can be compared at the image, data or feature level. Gleicher et al. [45] proposes a complementary taxonomy for InfoVis. The authors distinguish between methods that use juxtaposition, overlay or explicit encoding of differences.

Comparison at the *image level* is the most frequent one. It does not directly operate on the data but on 2D images that result, for example, from a visualization method or from experiments [148]. Examples include side-by-side visualizations (with synchronized viewing conditions) where the user has to mentally relate different images [9], [19], [29], [61]. Other approaches overlay co-registered images, for example, based on a checkerboard pattern or using transparency. Alternatively, per-pixel differences can be directly represent by subtracting the 2D representations from another [107]. For the latter, the selection of an appropriate color map is highly important, for instance, using a diverging map to discriminate positive and negative differences [149]. Zhou et al. [148] present a study of different comparison metrics that numerically quantify image differences between experiments and visualizations. It should be noted that image level comparison usually operates on 2D representations where the intermediate information about how the images were



(a) (b)

Fig. 8. Comparison of nested surfaces [154]: (a) Transparent surfaces, shadow-casting glyphs, coloring, and contouring are used for comparison. (b) Corresponding points on the surfaces are connected by glyphs (images courtesy of S. Busking).

created is lost. Deviations can, therefore, also result from different visualization settings (e.g., transfer functions, points-of-view, or lighting conditions) and need not necessarily represent data differences [44].

Data level comparison utilizes the raw data as a starting point and often incorporates intermediate information from the rendering process [150]. Sahasrabudhe et al. [151] propose methods for measuring the differences between scalar datasets including spatial and perceptual metrics. Kim et al. [152] propose metrics for data level comparison of direct volume rendering and also incorporate intermediate rendering information in their comparison approach. Malik et al. [153] recently propose the multi-image view that supports the comparison of series of scans from the same specimen.[3] Such approaches are usually superior to pure image level comparison since they include more information and flexibility.

Finally, *feature level comparison* is an extension of data level comparison and is based on extracted features of the data. For flow data, such features can be shock waves, vortices, streamlines, or isosurfaces (see the work of Verma and Pang [44] and Pagendarm and Post [155], for instance). Features from different modalities can be depicted as nested surfaces. Weigle and Taylor II [156], for example, use coloring, transparent textures, and shadow-casting glyphs for visually comparing two nested surfaces. Busking et al. [154] propose an image-based implementation of this approach and add contours where the surfaces intersect as well as local distance cues (see Fig. 8).

6.3 Interactive Methods for Multi-Modal Data

Rendering techniques for multi-modal data often rely on interaction, for example, specifying a transfer function for each data modality which is then combined during rendering [143], [144]. Another challenge with volume rendering is finding a good viewpoint where features are not occluded. This can be done interactively (e.g., by rotating the visualization or using clipping planes) or computationally assisted. Viola et al. [49], for example, utilize mutual information to automatically determine the most expressive viewpoint for a feature, which can be picked from a pre-defined list. The viewpoint then smoothly changes to give a clear view on the object of interest using a focus+context visualization [116].

3. To a certain degree, such dataset series resulting from multiple scans can be considered as multi-run data.

7 Multi-run Simulation Data

The previous two sections discuss approaches for a relatively small number of co-located volumes. For comparing such data, for instance, juxtaposed views or isosurfaces can be used [31]. However, the visual analysis of a larger number of concurrent data volumes requires more sophisticated methods. Such data commonly results from multi-run (or ensemble) simulations, which are performed increasingly often in automotive engineering [32], [51] or climate research [12], [31].

Multi-run simulations are an important step in the development of simulation models, where one aims to identify model parameters that have the most influence on the simulation output. In such a *sensitivity analysis* [34], the values of certain model parameters are changed systematically and multiple simulation runs are computed, accordingly. In the resulting data, a distribution of values is given for the same data attribute at each position in space and time (one value for each run). The visualization of multi-run data is especially interesting since it is an alternative approach for representing uncertainty [38], [43]. General approaches for uncertainty visualization are discussed by Pang et al. [157], Johnson and Sanderson [158], and Griethe and Schumann [159]. MacEachren et al. [160], moreover, review approaches for geospatial uncertainty visualization.

7.1 Representation of Multi-run Data

The representation of multi-run data is rather new to the visualization community [33]. It is especially challenging since the data are often higher-dimensional, multivariate, and large at the same time [38]. A direct depiction of many co-located and time-varying volumes of data is often not feasible. Accordingly, the distributions of multi-run values need to be aggregated, for example, by computing statistical summaries [33], [43]. The resulting data can then be visualized using box plots or glyphs, for example. Alternatively, InfoVis techniques such as parallel coordinates or scatterplot matrices can be combined with statistics [31], [161].

Box plots [162] encode important characteristics of data distributions such as minimum and maximum values, mean, median, and other quartile information. Kao et al. [163], [164] extend this approach to 2D multi-run data. In certain cases, the distribution can be represented adequately by statistical parameters such as mean, standard deviation, interquartile range, skewness or kurtosis. The computed statistics are visualized on 2D surfaces using colorcoding and bar glyphs. For other cases, the same authors propose a *shape descriptor* approach. A 3D volume is constructed where the data range is handled as a third dimension and the probability density function (PDF) of the multi-run data is used as voxel values. The peaks in the PDF are then described by a set of shape descriptors

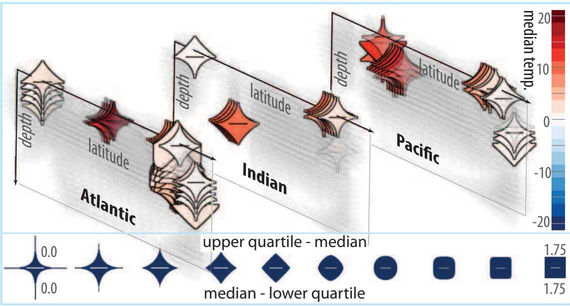
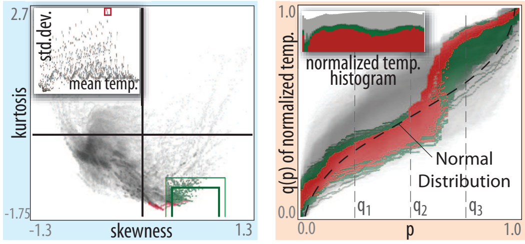


Fig. 9. Glyph-based 3D visualization of multi-run data [37]: Both upper and lower shape represent quartile information by changing from a star (small value), to a diamond, a circle, and a box representing a large value. Glyph size encodes the interquartile range and color shows median temperatures.



(a) (b)

Fig. 10. Relating aggregated and multi-run data of a climate simulation [36], [37]: (a) Interesting statistical properties are brushed and refined in the inset. (b) The corresponding multi-run distributions are emphasized in a quantile plot.

(e.g., number of peaks, height, width, and location), which are displayed on orthogonal 2D slices [164].

Spaghetti plots [42] are utilized by meteorologists to investigate multi-run data, where a contour line is visualized for each run at a selected time step (resembling a pile of spaghetti noodles). Sanyal et al. [165] combine spaghetti plots with a ribbon- and glyph-based uncertainty visualization. The uncertainty glyphs consist of a number of concentric colored circles that represent the standard deviation, interquartile range, and the width of the 95% confidence interval. Potter et al. [35] present a framework for analyzing multi-run data, which consists of overview and statistical visualizations such as trend charts or spaghetti plots. The same authors propose another extension of box plots. The so-called summary plot [43] includes additional statistics of the multi-run data such as skewness, kurtosis and tailing information. These plots, however, cannot be placed in a dense spatial context. Kehrer et al. [37] depict aggregated properties of multi-run data using 2D billboard glyphs that are based on super ellipses (see Fig. 9). The glyphs are carefully designed in order to be placed in a 3D context [123]. Using a focus+context visualization and brushing of aggregated statistics, glyphs that encode certain data characteristics can be interactively explored (see Sec. 7.2).

Chan et al. [166] augment 2D scatterplots by visualizing sensitivity information, which they considered similar to velocities in a flow field. Sensitivities are then represented as tangent lines on the individual points in the flow-based scatterplot. The assumed flow field can also be visualized using streamlines, and data points can be clustered by proximity to these lines. The proposed approach allows the analyst, for instance, to correlate changes in one variable with respect to another one.

7.2 Analysis in Multi-run Data Visualization

As mentioned earlier, statistical methods can be used to reduce the data dimensionality. Kehrer et al. [36], for example, integrate statistical moments (mean, variance, skewness, and kurtosis) into the visual analysis of multi-run data. Traditional and robust estimates of moments as well as measures of *outlyingness* are computed. A moment-based model for the visual analysis is proposed, which provides guidelines to the multitude of opportunities during such an analysis. Traditional estimates of moments can, for instance, be replaced by robust ones, or the scale of a data attribute can be changed by applying a normalization. For depicting the multi-run data, *quantile plots* that are common in statistics are adapted to enable a focus+context style (see Fig. 10b).

In the visual analysis, multi-run data and aggregated properties are related via an interface [37], which transfers selection information between the data parts.[4] This enables the analyst to work with both data representations simultaneously. Interesting multi-run distributions can then be selected, for instance, by brushing certain aggregated statistics (see Fig. 10a). For the investigated cases with multi-run data and aggregated statistics, the analysis usually starts at the aggregated level [36], [37]. Here, certain data characteristics can be specified via brushing. The feature can then be refined and investigated in detail in the related multi-run data. The analysis can then go back and forth between the data parts, where features are iteratively refined.

As an alternative, mathematical and procedural operators [33] can be applied, which transform the multi-run data into a form where existing visualization techniques are again applicable, for example, streamlines, isosurfaces or pseudo-coloring. The multi-run distributions can be compared against a reference distribution or a single threshold value when drawing contour lines or isosurfaces, for instance. This approach is very promising due to its flexibility. However, the usage of the operators and the interpretation of the resulting visualizations require additional training and care from the user.

Bordoloi et al. [167] apply hierarchical clustering techniques on multi-run data. Data can either be clustered along the spatial dimensions by grouping locations with similar statistical properties and probability density functions of multi-run values—this approach helps to identifying spatial structures and patterns, which may result from the simulated phenomenon. Alternatively, the runs can be clustered base on their similarity. Such an approach supports the comparison of different groups of simulation outcomes, where each group

4. The proposed interface concept is also applicable to other scenarios with heterogeneous scientific data such as multi-physics simulations discussed in section 8.

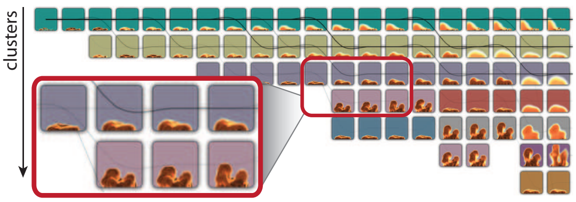


Fig. 11. Exploration of a multi-run simulation of a visual fire effect [168]: Each run is first split into similar time segments, which are then clustered across the runs. In an overview visualization, each row depicts one cluster with respect to a common time line. The simulation runs are drawn as paths between the clusters (image courtesy of S. Bruckner, © 2010 IEEE).

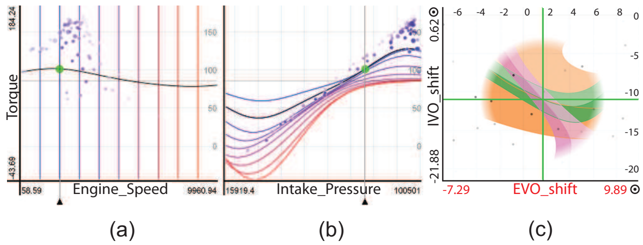


Fig. 12. Navigating input and output parameters of a multi-run simulation [51], [52]: (a) Variations of an input parameter are indicated with lines. (b) The corresponding changes in the simulation output are predicted and shown as function graphs. (c) The local neighborhood in three output parameters is mapped back to the input parameters, which indicates the sensitivity of the simulation with respect to its input parameters (images courtesy of H. Piringer and W. Berger).

can be represented [167]. In recent work, Bruckner and Möller [168] present a result-driven exploration approach for physically-based multi-run simulations. Each volumetric time sequence is first split into similar segments over time and thereafter grouped across different runs using a density-based clustering algorithm. This approach supports the user in identifying similar behavior in different simulation runs (see Fig. 11).

Correa et al. [169] propose a framework for uncertainty-aware visual analysis. Statistical methods such as uncertainty modeling are incorporated as well as uncertainty propagation and aggregation during data transformations. Approaches for data transformation such as regression, PCA, and k-means clustering are adopted in order to account for uncertainty. A number of views are presented that combine summarized and detailed visualizations of uncertainty. Dependent on the analysis task, uncertain data can be enhanced or de-emphasized.

7.3 Interactive Methods for Multi-run Data

A challenge with multi-run data is the relation of input to output variables of a simulation and vice versa. Nocke et al. [31] utilize coordinated multiple views for analyzing a large number of runs of climate simulations. Statistical aggregations are computed from the runs and visualized using linked scatterplots, graphical tables, or parallel coordinates. The sensitivity of the model to certain input parameters can be explored via brushing, and the related model runs can be compared in detail (compare to a similar approach on injection systems simulations [32]).

Certain methods that were originally designed for multi-dimensional data can be used for multi-run data as well. HyperSlice [170], for example, represents a higher dimensional function as a matrix of orthogonal 2D slices around a user-controlled n-dimensional focal point. The Prosection Matrix [171] extends this concept by projecting also the local neighborhood of the slices to 2D scatterplots. The approach supports also filtering via brushing. HyperMoVal [51] builds upon these concepts and enables the interactive visual validation of surrogate models. Such models are based on statistical regression and approximate the output of a more time-consuming simulation. HyperMoVal utilizes 2D and 3D projections of multi-run data around a user-controlled focal point. Model predictions of variations of one input parameter are represented as families of function graphs (see Fig. 12b). The predictions can then be compared to known results of the multi-run simulation (shown as points in the Figs. 12a and 12b). The approach supports, for example, the identification of regions with bad fit.

Berger et al. [52] extend HyperMoVal for exploring the continuous space of input and output variables of the simulation. The local neighborhood around the focal point in the input parameters is mapped to the output domain using k-nearest neighbor estimators or linear regression models. Since a direct mapping from output to input parameters of a simulation is not possible, a "spyhole" approach is proposed. In Figure 12c, the local area around the focal point is shown where variations of input parameters do not affect the predicted output by more than a certain threshold. Also, the uncertainty of the prediction can be visualized using box plots.

Matković et al. [41] visualize multi-run data as families of data surfaces with respect to pairs of independent data dimensions. Using multiple linked views and brushing, the authors analyze projections and aggregations of the data surfaces at different levels (e.g., a 1D profile or single aggregated value per surface). The same authors propose a visual steering approach [172] where new simulation runs are triggered by interactively narrowing down the control parameters in the visualization via brushing. This approach realizes a tight combination of interactive visualization and computational simulation.

In later work, Matković et al. [173] propose the simulation model view which is directly integrated in their coordinated multiple views framework. The view represents the building blocks of the utilized simulation process and model at three different levels of detail (using a histogram, scatterplot or curve view). The approach aims at bridging the gap between the simulation model and resulting multi-run data. Unger and Schumann [174] present a similar framework that facilitates the understanding of simulation processes at different levels. The user can compare and simultaneously explore the underlying data model, different parameter settings for the simulation, as well as individual runs or aggregated multi-run values.

We clearly see a lot of potential for future research on multi-run data. This kind of data is gaining increasing importance due to the technological developments in climate research, engineering, and other fields. Visualization must deal with multi-run data that are also multivariate and spatio-temporal. It is not at all straightforward to visualize an overview of several hundred runs of time-dependent 3D data. Advanced data abstraction and aggregation techniques are required that are aware of data trends and outliers.

8 Multi-Model Simulation Data

Data from multi-model simulations have been rarely addressed in the visualization community so far. Such simulations, however, are gaining importance in areas such as multi-physics or climate research [11], [12]. In the climate system, for instance, different compartments such as ocean, ice, surface, and atmosphere are interacting with each other. Ocean and atmosphere exchange through thermal absorption, precipitation, and evaporation, also ice and air are interacting. Accordingly, ocean and atmosphere models are often coupled in the simulation [12]. The models are often not computed on the same types of grid, or for the same time steps. When analyzing feedback between these models, statistical aggregates are usually investigated. Fluid–structure interactions (FSIs), to address another example, are interactions of a deformable or movable structure with an internal or surrounding flow [11]. They are among the most important and—with respect to both modeling and computational issues—the most challenging multi-physics problems, and therefore currently a hot topic in simulation research itself. The variety of FSI occurrences is abundant and ranges from bridges, flexible roofs, or off-shore platforms to micropumps and injection systems, from parachutes via airbags to blood flow in arteries or artificial heart valves, to name just a few [11].

For visualization research it is very challenging to generate a coherent representation from such data, for instance, when one model is simulated on a 2D grid and the other one on a 3D grid. How can different attributes given in the different models be compared to each other? How can data be represented, where there are values missing (e.g., an attribute is simulated in one model but not in the other, or the data are not uniformly available for a spatial dimension). Also, how can selections and features be communicated between different models, especially when these are given on non-overlapping grids or different time steps? One is, for example, interested in the areas of an ocean model that are influenced by adjacent atmospheric regions exhibiting certain characteristics such as high temperatures. How can such a feature from the atmosphere be propagated to the ocean part?

Kehrer et al. [37] recently propose a concept that integrates and relates two parts of scientific data in the visual analysis. The fractional degree-of-interest (DOI) attribution of the data, resulting from smooth brushing [80], is utilized as a common level of data abstraction between the related parts. As an example, a fluid–structure interaction of warm water flow though a cooler aluminum foam is investigated. Similar to the simulation, an *interface* is created that relates individual

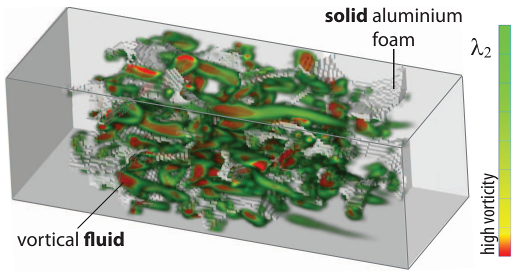


Fig. 13. Visual analysis of heat transfer between the fluid and solid part of a multi-physics simulation [37]: Both data parts are related via an interface that enables a bidirectional transfer of features. Vortical flow is selected via brushing. The feature is automatically transferred to the solid part, where it is further analyzed and related to other features.

grid cells between the two data parts (see Fig. 2). The interface enables the bidirectional transfer of DOI information. For understanding flow characteristics such as heat exchange, vortices are highly important. Vortical regions are thus selected via brushing in the fluid part of the data (see Fig. 13). The corresponding feature is instantly transferred to the foam part via the interface. Here, it can be related to other features specified in the solid part. The interface enables the investigation of a direct relation between turbulent flow around the foam structure and a corresponding heating in the foam.

To the best of the authors' knowledge, this work [37] is the first step into the direction of visual analysis of multi-model scenarios. Since the related simulation are becoming increasingly popular in the application domains, we see a great potential for future visualization research here.

9 Discussion and Outlook

The majority of the approaches discussed in this survey specifically address one or two facets of scientific data. What is often missing are general concepts for handling the heterogeneity of multi-faceted data (e.g., multi-run data are often spatio-temporal and multi-variate as well). One possible solution are coordinated multiple views, which combine and link well-known visualizations for different kinds of data. In such a framework, for example, function graphs can be used to analyze time-varying data, volume rendering for spatial data, 2D scatterplots or parallel coordinates for multi-variate data, or glyphs and box plots for encoding summary statistics of multi-run data. A challenge in this context is the relation of multiple views, allowing the investigation of features across data facets, datasets, as well as levels of data abstraction [9], [37], [74], [75], [115]. Context-preserving visual links [175], for example, interconnect related pieces of information across views and applications.

Another challenge is the extraction of meaningful information from heterogeneous scientific data. Such data can be fused on the data level, for example, by resampling or by using a data model that provides unified access to different modalities [142]. Another option is to fuse multi-faceted data on the feature level, for example, by exchanging selection information across different data parts [37], [75]. The data markups then represent the first level of semantic abstraction, ranging from raw data to knowledge [56]. This abstraction to the feature level enables the joint integration of heterogeneous data parts. Features can be exchanged between parts that are given on different levels of aggregation (e.g., multi-run and aggregated data) or on various types of grids [37]. Such a semantic abstraction is especially useful, since domain scientists usually think in application terms instead of data terms (e.g., objects or phenomena).

One interesting observation from our study was that many overview articles discuss approaches according to the different stages in the visualization pipeline (e.g., multi-variate data fusion [23], comparative visualization [44], [45], or quality metrics [48]). Analytical methods can, for example, be applied before the visualization mapping (in data space). Alternatively, they can control the visualization mapping or measure the quality of the resulting image [48]. Also, the user can interactively control the settings at different stages of the analysis pipeline [59]. Especially the combination of computational analysis and interactive visualization methodology—as proposed in the visual analytics agenda [4]—is a promising direction, and we expect to see a lot of more interesting work in this area. Examples are feature-based approaches that (semi)automatically extract interesting patterns from the data [74], [75], [129]. Recently, May [176] presented a thoroughly structured overview of different opportunities for integrating interactive and computational means in visual analytics. One important step in this context is the integration of machine learning methods that can learn from previous user input and data, and configure the parameters of the visualization based on the acquired knowledge [56], [91].

As another observation, we see a gap between the techniques used by domain scientists and the approaches available from visualization research. Recent advances in visualization are rarely used in application domains such as climate research (compare to Nocke et al. [177]). One reason for this may be that systems are complex to use and can overwhelm the user with a multitude of options and parameters. Also, it is not always obvious how such methods integrate into the typical workflow of the domain. A major challenge for future developments is thus to further bridge this gap, for example, by including knowledge from domain experts when designing visualization solutions [2]. Visualizations should follow guidelines from perception research and human–computer interaction [39], providing simple graphical user interfaces and advanced visualizations [59].

In general, current approaches rarely address the heterogeneity of multi-faceted scientific data. We see here a definite need for novel concepts and methods and thus see it as a promising research direction in visualization.

10 Conclusions

The visualization and interactive visual analysis of multi-faceted scientific data are gaining increased importance in areas such as engineering, medicine or climate research. This is due to the fact that computational power increases rapidly, measurements are getting more accurate and detailed, and multi-modal data are becoming more common. Accordingly, also model and data scenarios are getting more complex. Data are often multi-variate, spatio-temporal and stem from multi-modal, multi-run, and/or multi-model scenarios. Visualization has been well established to explore and analyze single facets of such data and to communicate results from data analysis. With respect to multi-faceted scientific data, however, we see a variety of interesting challenges that require advanced visualization technology. In this survey, the related state of the art has been discussed. A categorization of approaches has been proposed that is based on common visualization, interaction and analysis methods. What is largely missing are approaches that address a multitude of facets of scientific data.

We identify, in particular, the visualization and analysis of data stemming from multi-run simulations and interacting simulation models (e.g., coupled climate models or multi-physics simulations) as rewarding directions for future research, as well as multi-modal visualization. A challenge is to jointly integrate larger amounts of concurrent data volumes in the visual analysis, possibly given on different grids and/or with different data dimensionality [37]. Another challenge is how to investigate feedback between interacting compartments of the simulation. For multi-variate and time-dependent data, we can find a lot of related work that brings up good solutions. The visualization and analysis of these kinds of data belong to the top challenges in current visualization research [2].

Acknowledgments

The authors thank Heidrun Schumann (Univ. of Rostock) and Min Chen (Univ. of Oxford) for their valuable comments on an earlier version of this paper [178]. We are indebted to Thomas Nocke from the Potsdam Institute for Climate Impact Research (PIK), Germany, for fruitful discussions and input to earlier work on climate data that was the basis for this survey. Parts of this work have been supported by the Austrian Funding Agency (FFG) within the scope of the COMET K1 program.

References

[1] J. van Wijk, "The value of visualization," *Proc. IEEE Conf. Visualization*, pp. 79–86, 2005.

[2] C. Johnson, "Top scientific visualization research problems," *IEEE Computer Graphics and Applications*, vol. 24, no. 4, pp. 13–17, 2004.

[3] H. Hauser, "Interactive visual analysis – an opportunity for industrial simulation," *Proc. Simulation and Visualization (SimVis)*, pp. 1–6, 2006.

[4] J. Thomas and K. Cook, "A visual analytics agenda," *IEEE Computer Graphics and Applications*, vol. 26, no. 1, pp. 10–13, 2006.

[5] D. Keim, G. Andrienko, J.-D. Fekete, C. Görg, J. Kohlhammer, and G. Melançon, "Visual analytics: Definition, process, and challenges," *Information Visualization*, ser. Lecture Notes in Computer Science, vol. 4950, pp. 154–175, 2008.

[6] D. Keim, J. Kohlhammer, G. Ellis, and F. Mansmann, Eds., *Mastering the Information Age: Solving Problems with Visual Analytics*. Eurographics Assoc., 2010.

[7] B. Shneiderman, "The eyes have it: A task by data type taxonomy for information visualizations," *Proc. IEEE Symp. Visual Languages*, pp. 336–343, 1996.

[8] C. North, N. Conklin, K. Indukuri, and V. Saini, "Visualization schemas and a web-based architecture for custom multiple-view visualization of multiple-table databases," *Information Visualization*, vol. 1, no. 3–4, pp. 211–228, 2002.

[9] C. Stolte, D. Tang, and P. Hanrahan, "Polaris: A system for query, analysis, and visualization of multidimensional relational databases," *IEEE Trans. Visualization and Computer Graphics*, vol. 8, no. 1, pp. 52–65, 2002.

[10] M. Cammarano, X. Dong, B. Chan, J. Klingner, J. Talbot, A. Halevy, and P. Hanrahan, "Visualization of heterogeneous data," *IEEE Trans. Visualization and Computer Graphics*, vol. 13, no. 6, pp. 1200–1207, 2007.

[11] H.-J. Bungartz and M. Schäfer, Eds., *Fluid-Structure Interaction: Modelling, Simulation, Optimisation*, ser. Lecture Notes in Computational Science and Engineering. Springer, 2006.

[12] B. Hibbard, M. Böttinger, M. Schultz, and J. Biercamp, "Visualization in Earth System Science," *SIGGRAPH Comput. Graph.*, vol. 36, no. 4, pp. 5–9, 2002.

[13] J. Dykes, A. MacEachren, and M.-J. Kraak, Eds., *Exploring Geovisualization*. Elsevier, 2005.

[14] G. Andrienko, N. Andrienko, J. Dykes, S. Fabrikant, and M. Wachowicz, "Geovisualization of dynamics, movement and change: key issues and developing approaches in visualization research," *Information Visualization*, vol. 7, no. 3, pp. 173–180, 2008.

[15] N. Andrienko and G. Andrienko, *Exploratory analysis of spatial and temporal data: A systematic approach*. Springer, 2006.

[16] K.-L. Ma and E. Lum, "Techniques for visualizing time-varying volume data," *Visualization Handbook*, C. Hansen and C. Johnson, Eds., pp. 511–531. Academic Press, 2004.

[17] W. Aigner, S. Miksch, W. Müller, H. Schumann, and C. Tominski, "Visual methods for analyzing time-oriented data," *IEEE Trans. Visualization and Computer Graphics*, vol. 14, no. 1, pp. 47–60, 2008.

[18] ——, "Visualizing time-oriented data: A systematic view," *Computers & Graphics*, vol. 31, no. 3, pp. 401–409, 2007.

[19] T. Jankun-Kelly and K.-L. Ma, "Visualization exploration and encapsulation via a spreadsheet-like interface," *IEEE Trans. Visualization and Computer Graphics*, vol. 7, no. 3, pp. 275–287, 2001.

[20] T. Saito, H. N. Miyamura, M. Yamamoto, H. Saito, Y. Hoshiya, and T. Kaseda, "Two-tone pseudo coloring: Compact visualization for one-dimensional data," *Proc. IEEE Symp. Information Visualization*, pp. 173–180, 2005.

[21] J. Kehrer, F. Ladstädter, P. Muigg, H. Doleisch, A. Steiner, and H. Hauser, "Hypothesis generation in climate research with interactive visual data exploration," *IEEE Trans. Visualization and Computer Graphics*, vol. 14, no. 6, pp. 1579–1586, 2008.

[22] P. C. Wong and R. D. Bergeron, "30 years of multidimensional multivariate visualization," *Scientific Visualization: Overviews, Methodologies, and Techniques*, pp. 3–33, 1997.

[23] R. Fuchs and H. Hauser, "Visualization of multi-variate scientific data," *Computer Graphics Forum*, vol. 28, no. 6, pp. 1670–1690, 2009.

[24] E. Bertini and D. Lalanne, "Investigating and reflecting on the integration of automatic data analysis and visualization in knowledge discovery," *ACM SIGKDD Explor. Newsl.*, vol. 11, pp. 9–18, 2009.

[25] D. Keim, W. Müller, and H. Schumann, "Visual data mining," *Eurographics 2002 State of the Art Reports*, pp. 49–68, 2002.

[26] J. C. Roberts, "State of the art: Coordinated & multiple views in exploratory visualization," *Proc. Coordinated & Multiple Views in Exploratory Visualization (CMV)*, pp. 61–71, 2007.

[27] A. Ardeshir Goshtasby, *2D and 3D image registration for medical, remote sensing, and industrial applications*. Wiley, 2005.

[28] J. Pluim, J. Antoine Maintz, and M. Viergever, "Mutual-information-based registration of medical images: A survey," *IEEE Trans. Medical Imaging*, vol. 22, no. 8, pp. 986–1004, 2003.

[29] L. Treinish, "Visual data fusion for applications of high-resolution numerical weather prediction," *Proc. IEEE Conf. Visualization*, pp. 477–480, 2000.

[30] H.-G. Pagendarm and F. Post, "Comparative visualization – approaches and examples," *Visualization in Scientific Computing*, pp. 95–108, 1995.

[31] T. Nocke, M. Flechsig, and U. Böhm, "Visual exploration and evaluation of climate-related simulation data," *Proc. Winter Simulation Conf.*, pp. 703–711, 2007.

[32] K. Matković, M. Jelović, J. Jurić, Z. Konyha, and D. Gračanin, "Interactive visual analysis and exploration of injection systems simulations," *Proc. IEEE Visualization*, pp. 391–398, 2005.

[33] A. Love, A. Pang, and D. Kao, "Visualizing spatial multivalue data," *IEEE Computer Graphics and Applications*, vol. 25, no. 3, pp. 69–79, 2005.

[34] D. M. Hamby, "A review of techniques for parameter sensitivity analysis of environmental models," *J. Environmental Monitoring & Assessment*, vol. 32, no. 2, pp. 135–154, 2004.

[35] K. Potter, A. Wilson, P.-T. Bremer, D. Williams, C. Doutriaux, V. Pascucci, and C. Johnson, "Ensemble-Vis: a framework for the statistical visualization of ensemble data," *Proc. IEEE Int'l. Conf. on Data Mining Workshops*, pp. 233–240, 2009.

[36] J. Kehrer, P. Filzmoser, and H. Hauser, "Brushing moments in interactive visual analysis," *Computer Graphics Forum*, vol. 29, no. 3, pp. 813–822, 2010.

[37] J. Kehrer, P. Muigg, H. Doleisch, and H. Hauser, "Interactive visual analysis of heterogeneous scientific data across an interface," *IEEE Trans. Visualization and Computer Graphics*, vol. 17, no. 7, pp. 934–946, 2011.

[38] A. Wilson and K. Potter, "Toward visual analysis of ensemble data sets," *Proc. Ultrascale Visualization Workshop*, pp. 48–53, 2009.

[39] M. Tory and T. Möller, "Human factors in visualization research," *IEEE Trans. Visualization and Computer Graphics*, vol. 10, no. 1, pp. 72–84, 2004.

[40] J. S. Yi, Y. a. Kang, J. T. Stasko, and J. A. Jacko, "Toward a deeper understanding of the role of interaction in information visualization," *IEEE Trans. Visualization and Computer Graphics*, vol. 13, no. 6, pp. 1224–1231, 2007.

[41] K. Matković, D. Gračanin, B. Klarin, and H. Hauser, "Interactive visual analysis of complex scientific data as families of data surfaces," *IEEE Trans. Visualization and Computer Graphics*, vol. 15, no. 6, pp. 1351–1358, 2009.

[42] P. Diggle, P. Heagerty, K.-Y. Liang, and S. Zeger, *Analysis of longitudinal data*. Oxford University Press, 2002.

[43] K. Potter, J. Kniss, R. Riesenfeld, and C. Johnson, "Visualizing summary statistics and uncertainty," *Computer Graphics Forum*, vol. 29, no. 3, pp. 823–832, 2010.

[44] V. Verma and A. Pang, "Comparative flow visualization," *IEEE Trans. Visualization and Computer Graphics*, vol. 10, no. 6, pp. 609–624, 2004.

[45] M. Gleicher, D. Albers, R. Walker, I. Jusufi, C. Hansen, and J. Roberts, "Visual comparison for information visualization," *Information Visualization*, vol. 10, no. 4, pp. 289–309, 2011.

[46] N. Elmqvist, P. Dragicevic, and J.-D. Fekete, "Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation," *IEEE Trans. Visualization and Computer Graphics*, vol. 14, no. 6, pp. 1539–1148, 2008.

[47] D. Asimov, "The grand tour: A tool for viewing multidimensional data," *SIAM J. Scientific and Statistical Computing*, vol. 6, pp. 128–143, 1985.

[48] E. Bertini, A. Tatu, and D. Keim, "Quality metrics in high-dimensional data visualization: An overview and systematization," *IEEE Trans. Visualization and Computer Graphics*, vol. 17, no. 12, pp. 2203–2212, 2011.

[49] I. Viola, M. Feixas, M. Sbert, and M. E. Gröller, "Importance-driven focus of attention," *IEEE Trans. Visualization and Computer Graphics*, vol. 12, pp. 933–940, 2006.

[50] R. Becker and W. Cleveland, "Brushing scatterplots," *Technometrics*, vol. 29, no. 2, pp. 127–142, 1987.

[51] H. Piringer, W. Berger, and J. Krasser, "HyperMoVal: interactive visual validation of regression models for real-time simulation," *Computer Graphics Forum*, vol. 29, no. 3, pp. 983–992, 2010.

[52] W. Berger, H. Piringer, P. Filzmoser, and M. E. Gröller, "Uncertainty-aware exploration of continuous parameter spaces using multivariate prediction," *Computer Graphics Forum*, vol. 30, no. 3, pp. 911–920, 2011.

[53] H. Hauser, "Generalizing focus+context visualization," *Scientific Visualization: The Visual Extraction of Knowledge from Data*, pp. 305–327, 2005.

[54] A. Cockburn, A. Karlson, and B. Bederson, "A review of overview+detail, zooming, and focus+context interfaces," *ACM Computing Surveys*, vol. 41, pp. 2:1–2:31, 2009.

[55] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, 2nd ed. Springer, 2009.

[56] M. Chen, D. Ebert, H. Hagen, R. Laramee, R. van Liere, K.-L. Ma, W. Ribarsky, G. Scheuermann, and D. Silver, "Data, information, and knowledge in visualization," *IEEE Computer Graphics and Applications*, vol. 29, pp. 12–19, 2009.

[57] P. Keller and M. Keller, *Visual Cues: Practical Data Visualization*. IEEE Press, 1993.

[58] R. Amar and J. Stasko, "A knowledge task-based framework for design and evaluation of information visualizations," *Proc. IEEE Symp. Information Visualization*, pp. 143–150, 2004.

[59] B. Shneiderman, "Inventing discovery tools: combining information visualization with data mining," *Information Visualization*, vol. 1, no. 1, pp. 5–12, 2002.

[60] J. Tukey, *Exploratory Data Analysis*. Addison-Wesley, 1977.

[61] E. Tufte, *The Visual Display of Quantitative Information*. Graphics Press, 1983.

[62] W. Cleveland, *Visualizing Data*. Hobart Press, 1993.

[63] S. Card, J. Mackinlay, and B. Shneiderman, *Readings in information visualization: using vision to think*. M. Kaufmann, 1999.

[64] D. Keim, "Information visualization and visual data mining," *IEEE Trans. Visualization and Computer Graphics*, vol. 8, no. 1, pp. 1–8, 2002.

[65] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "From data mining to knowledge discovery in databases," *AI Magazine*, vol. 17, pp. 37–54, 1996.

[66] M. de Oliveira and H. Levkowitz, "From visual data exploration to visual data mining: a survey," *IEEE Trans. Visualization and Computer Graphics*, vol. 9, no. 3, pp. 378–394, 2003.

[67] A. Inselberg, "The plane with parallel coordinates," *The Visual Computer*, vol. 1, no. 2, pp. 69–91, 1985.

[68] J. Johansson, P. Ljung, M. Jern, and M. Cooper, "Revealing structure within clustered parallel coordinates displays," *Proc. IEEE Symp. Information Visualization*, pp. 125–132, 2005.

[69] M. Novotný and H. Hauser, "Outlier-preserving focus+context visualization in parallel coordinates," *IEEE Trans. Visualization and Computer Graphics*, vol. 12, no. 5, pp. 893–900, 2006.

[70] Z. Konyha, K. Matković, D. Gračanin, M. Jelović, and H. Hauser, "Interactive visual analysis of families of function graphs," *IEEE Trans. Visualization and Computer Graphics*, vol. 12, no. 6, pp. 1373–1385, 2006.

[71] P. Muigg, J. Kehrer, S. Oeltze, H. Piringer, H. Doleisch, B. Preim, and H. Hauser, "A four-level focus+context approach to interactive visual analysis of temporal features in large scientific data," *Computer Graphics Forum*, vol. 27, no. 3, pp. 775–782, 2008.

[72] S. Havre, E. Hetzler, P. Whitney, and L. Nowell, "ThemeRiver: visualizing thematic changes in large document collections," *IEEE Trans. Visualization and Computer Graphics*, vol. 8, no. 1, pp. 9–20, 2002.

[73] M. Ward, "XmdvTool: Integrating multiple methods for visualizing multivariate data," *Proc. IEEE Conf. Visualization*, pp. 326–336, 1994.

[74] H. Doleisch, M. Gasser, and H. Hauser, "Interactive feature specification for focus+context visualization of complex simulation data," *Proc. Symp. Data Visualization (VisSym)*, pp. 239–248, 2003.

[75] C. Weaver, "Cross-filtered views for multidimensional visual analysis," *IEEE Trans. Visualization and Computer Graphics*, vol. 16, no. 2, pp. 192–204, 2010.

[76] D. Gresh, B. Rogowitz, R. Winslow, D. Scollan, and C. Yung, "WEAVE: a system for visually linking 3D and statistical visualizations applied to cardiac simulation and measurement data," *Proc. IEEE Conf. Visualization*, pp. 489–492, 2000.

[77] O. Rübel *et al.*, "PointCloudXplore: visual analysis of 3D gene expression data using physical views and parallel coordinates," *Proc. Symp. Data Visualization (EuroVis)*, pp. 203–210, 2006.

[78] S. Oeltze, H. Doleisch, H. Hauser, P. Muigg, and B. Preim, "Interactive visual analysis of perfusion data," *IEEE Trans. Visualization and Computer Graphics*, vol. 13, no. 6, pp. 1392–1399, 2007.

[79] G. W. Furnas, "Generalized fisheye views," *Proc. Human Factors in Computing Systems (CHI '86)*, pp. 16–23, 1986.

[80] H. Doleisch and H. Hauser, "Smooth brushing for focus+context visualization of simulation data in 3D," *J. WSCG*, vol. 10, no. 1, pp. 147–154, 2002.

[81] P. Muigg, M. Hadwiger, H. Doleisch, and H. Hauser, "Scalable hybrid unstructured and structured grid raycasting," *IEEE Trans. Visualization and Computer Graphics*, vol. 13, no. 6, pp. 1592–1599, 2007.

[82] F. Post, B. Vrolijk, H. Hauser, R. Laramee, and H. Doleisch, "The state of the art in flow visualisation: Feature extraction and tracking," *Computer Graphics Forum*, vol. 22, no. 4, pp. 775–792, 2003.

[83] T. Nocke, H. Schumann, and U. Böhm, "Methods for the visualization of clustered climate data," *Computational Statistics*, vol. 19, no. 1, pp. 75–94, 2004.

[84] J. van Wijk and E. van Selow, "Cluster and calendar based visualization of time series data," *Proc. IEEE Symp. Information Visualization*, pp. 4–9, 1999.

[85] W. Müller, T. Nocke, and H. Schumann, "Enhancing the visualization process with principal component analysis to support the exploration of trends," *Proc. Asia Pacific Symp. on Information Visualisation (APVIS '06)*, pp. 121–130, 2006.

[86] A. Buja, D. Swayne, M. Littman, N. Dean, and H. Hofmann, "XGvis: interactive data visualization with multidimensional scaling," *J. Computational and Graphical Statistics*, 2004.

[87] A. Morrison, G. Ross, and M. Chalmers, "Fast multidimensional scaling through sampling, springs and interpolation," *Information Visualization*, vol. 2, no. 1, pp. 68–77, 2003.

[88] T. Kohonen, *Self-organizing maps*, 2nd ed. Springer, 1997.

[89] P. Agarwal and A. Skupin, Eds., *Self-Organising Maps: Applications in Geographic Information Science*. Wiley, 2008.

[90] G. Andrienko, N. Andrienko, S. Bremm, T. Schreck, T. von Landesberger, P. Bak, and D. Keim, "Space-in-time and time-in-space self-organizing maps for exploring spatiotemporal patterns," *Computer Graphics Forum*, vol. 29, no. 3, pp. 913–922, 2010.

[91] K.-L. Ma, "Machine learning to boost the next generation of visualization technology," *IEEE Computer Graphics and Applications*, vol. 27, no. 5, pp. 6–9, 2007.

[92] A. MacEachren, *How Maps Work: Representation, Visualization, and Design*. Guilford Press, 2004.

[93] C. Tominski, P. Schulze-Wollgast, and H. Schumann, "3D information visualization for time-dependent data on maps," *Proc. Int'l. Conf. Information Visualization (IV)*, pp. 175–181, 2005.

[94] R. Kosara, F. Bendix, and H. Hauser, "Time Histograms for large, time-dependent data," *Proc. Symp. Data Visualization (VisSym)*, pp. 45–54, 2004.

[95] H. Hochheiser and B. Shneiderman, "Dynamic query tools for time series data sets: timebox widgets for interactive exploration," *Information Visualization*, vol. 3, no. 1, pp. 1–18, 2004.

[96] W. Tobler, "A computer movie simulating urban growth in the detroit region," *Economic Geography*, vol. 46, no. 2, pp. 234–240, 1970.

[97] T. Jankun-Kelly and K.-L. Ma, "A study of transfer functions generation for time-varying volume data," *Proc. EG/IEEE TCVG Workshop Volume Graphics*, pp. 51–68, 2001.

[98] J. Woodring and H.-W. Shen, "Chronovolumes: a direct rendering technique for visualizing time-varying data," *Proc. Symp. Data Visualization (VisSym)*, pp. 27–34, 2003.

[99] A. Joshi and P. Rheingans, "Illustration-inspired techniques for visualizing time-varying data," *Proc. IEEE Conf. Visualization*, pp. 679–686, 2005.

[100] R. Laramee, H. Hauser, H. Doleisch, B. Vrolijk, F. Post, and D. Weiskopf, "The state of the art in flow visualization: Dense and texture-based techniques," *Computer Graphics Forum*, vol. 23, no. 2, pp. 203–221, 2004.

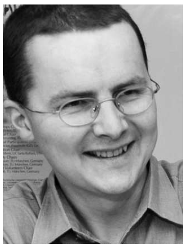
[101] T. Salzbrunn, H. Jänicke, T. Wischgoll, and G. Scheuermann, "The state of the art in flow visualization: Partition-based techniques," *Proc. Simulation and Visualization (SimVis)*, pp. 75–92, 2008.

[102] T.-C. Fu, "A review on time series data mining," *Eng. Appl. of Artificial Intelligence*, vol. 24, no. 1, pp. 164–181, 2011.

[103] I. López, R. Snodgrass, and B. Moon, "Spatiotemporal aggregate computation: A survey," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 2, pp. 271–286, 2005.
[104] M. Hao, D. Keim, U. Dayal, and T. Schreck, "Multi-resolution techniques for visual exploration of large time-series data," *Proc. Symp. Data Visualization (EuroVis)*, pp. 27–34, 2007.
[105] N. Andrienko and G. Andrienko, "Spatial generalization and aggregation of massive movement data," *IEEE Trans. Visualization and Computer Graphics*, vol. 17, no. 2, pp. 205–219, 2011.
[106] N. Willems, H. van de Wetering, and J. van Wijk, "Visualization of vessel movements," *Computer Graphics Forum*, pp. 959–966, 2009.
[107] O. Daae Lampe, J. Kehrer, and H. Hauser, "Visual analysis of multivariate movement data using interactive difference views," *Proc. Vision, Modeling, and Visualization (VMV)*, pp. 315–322, 2010.
[108] H. Jänicke, A. Wiebel, G. Scheuermann, and W. Kollmann, "Multifield visualization using local statistical complexity," *IEEE Trans. Visualization and Computer Graphics*, vol. 13, no. 6, pp. 1384–1391, 2007.
[109] H. Jänicke, M. Böttinger, U. Mikolajewicz, and G. Scheuermann, "Visual exploration of climate variability changes using wavelet analysis," *IEEE Trans. Visualization and Computer Graphics*, vol. 15, no. 6, pp. 1375–1382, 2009.
[110] J. Woodring and H.-W. Shen, "Multiscale time activity data exploration via temporal clustering visualization spreadsheet," *IEEE Trans. Visualization and Computer Graphics*, vol. 15, no. 1, pp. 123–137, 2009.
[111] H. Akiba, N. Fout, and K.-L. Ma, "Simultaneous classification of time-varying volume data based on the Time Histogram," *Proc. Symp. Data Visualization (EuroVis)*, pp. 171–178, 2006.
[112] H. Hauser, F. Ledermann, and H. Doleisch, "Angular brushing of extended parallel coordinates," *Proc. IEEE Symp. Information Visualization*, pp. 127–130, 2002.
[113] M. Jern, T. Åström, and S. Johansson, "GeoAnalytics tools applied to large geospatial datasets," *Proc. Int'l. Conf. Information Visualization (IV)*, pp. 362–372, 2008.
[114] J. Kniss, C. Hansen, M. Grenier, and T. Robinson, "Volume rendering multivariate data to visualize meteorological simulations: a case study," *Proc. Symp. Data Visualization (VisSym)*, pp. 189–194, 2002.
[115] P. Rautek, S. Bruckner, and M. E. Gröller, "Semantic layers for illustrative volume rendering," *IEEE Trans. Visualization and Computer Graphics*, vol. 13, pp. 1336–1343, 2007.
[116] I. Viola, A. Kanitsar, and M. E. Gröller, "Importance-driven feature enhancement in volume visualization," *IEEE Trans. Visualization and Computer Graphics*, vol. 11, no. 4, pp. 408–418, 2005.
[117] W. Cleveland and R. McGill, "Graphical perception: Theory, experimentation, and application to the development of graphical methods," *J. American Statistical Association*, vol. 79, no. 387, pp. 531–554, 1984.
[118] C. Healey, K. Booth, and J. Enns, "High-speed visual estimation using preattentive processing," *ACM Trans. Computer-Human Interaction*, vol. 3, pp. 107–135, 1996.
[119] C. Healey and J. Enns, "Large datasets at a glance: Combining textures and colors in scientific visualization," *IEEE Trans. Visualization and Computer Graphics*, vol. 5, no. 2, pp. 145–167, 1999.
[120] M. Ward, "Multivariate data glyphs: Principles and practice," *Handbook of Data Visualization*, C.-H. Chen, W. Härdle, and A. Unwin, Eds., pp. 179–198. Springer, 2008.
[121] T. Ropinski and B. Preim, "Taxonomy and usage guidelines for glyph-based medical visualization," *Proc. Simulation and Visualization (SimVis)*, pp. 121–138, 2008.
[122] G. Kindlmann, "Superquadric tensor glyphs," *Proc. Symp. Data Visualization (VisSym)*, pp. 147–154, 2004.
[123] A. Lie, J. Kehrer, and H. Hauser, "Critical design and realization aspects of glyph-based 3D data visualization," *Proc. Spring Conf. Computer Graphics (SCCG)*, pp. 27–34, 2009.
[124] R. Kirby, H. Marmanis, and D. Laidlaw, "Visualizing multivalued data from 2D incompressible flows using concepts from painting," *Proc. IEEE Visualization*, pp. 333–340, 1999.
[125] L. Treinish, "Task-specific visualization design," *IEEE Computer Graphics and Applications*, vol. 19, no. 5, pp. 72–77, 1999.
[126] P. C. Wong, H. Foote, D. L. Kao, R. Leung, and J. Thomas, "Multivariate visualization with data fusion," *Information Visualization*, vol. 1, no. 3/4, pp. 182–193, 2002.
[127] H. Hauser, L. Mroz, G.-I. Bischi, and M. E. Gröller, "Two-level volume rendering," *IEEE Trans. Visualization and Computer Graphics*, vol. 7, no. 3, pp. 242–252, 2001.
[128] M. Hadwiger, C. Berger, and H. Hauser, "High-quality two-level volume rendering of segmented data sets on consumer graphics hardware," *Proc. IEEE Conf. Visualization*, pp. 301–308, 2003.
[129] H. Jänicke, M. Böttinger, and G. Scheuermann, "Brushing of attribute clouds for the visualization of multivariate data," *IEEE Trans. Visualization and Computer Graphics*, vol. 14, no. 6, pp. 1459–1466, 2008.
[130] J. Seo and B. Shneiderman, "A rank-by-feature framework for unsupervised multidimensional data exploration using low dimensional projections," *Proc. IEEE Symp. Information Visualization*, pp. 65–72, 2004.
[131] L. Wilkinson, A. Anand, and R. Grossman, "Graph-theoretic scagnostics," *Proc. IEEE Symp. Information Visualization*, pp. 157–164, 2005.
[132] A. Tatu, G. Albuquerque, M. Eisemann, P. Bak, H. Theisel, M. Magnor, and D. Keim, "Automated analytical methods to support visual exploration of high-dimensional data," *IEEE Trans. Visualization and Computer Graphics*, vol. 17, no. 5, pp. 584–597, 2011.
[133] N. Sauber, H. Theisel, and H.-P. Seidel, "Multifield-Graphs: an approach to visualizing correlations in multifield scalar data," *IEEE Trans. Visualization and Computer Graphics*, vol. 12, no. 5, pp. 917 –924, 2006.
[134] J. Woodring and H.-W. Shen, "Multi-variate, time-varying, and comparative visualization with contextual cues," *IEEE Trans. Visualization and Computer Graphics*, vol. 12, pp. 909–916, 2006.
[135] F.-Y. Tzeng, E. Lum, and K.-L. Ma, "An intelligent system approach to higher-dimensional classification of volume data," *IEEE Trans. Visualization and Computer Graphics*, vol. 11, no. 3, pp. 273–284, 2005.
[136] R. Fuchs, J. Waser, and M. E. Gröller, "Visual human+machine learning," *IEEE Trans. Visualization and Computer Graphics*, vol. 15, no. 6, pp. 1327–1334, 2009.
[137] D. Lipşa, R. Laramee, S. Cox, J. Roberts, and R. Walker, "Visualization for the physical sciences," *Eurographics 2011 State of the Art Reports*, pp. 49–73, 2011.
[138] L. Treinish, "Case study: Severe rainfall events in northwestern peru (visualization of scattered meteorological data)," *Proc. IEEE Conf. Visualization*, pp. 350–354, 1994.
[139] J. Beyer, M. Hadwiger, S. Wolfsberger, and K. Bühler, "High-quality multimodal volume rendering for preoperative planning of neurosurgical interventions," *IEEE Trans. Visualization and Computer Graphics*, vol. 13, no. 6, pp. 1696–1703, 2007.
[140] J. M. Favre, "Towards efficient visualization support for single-block and multi-block datasets," *Proc. IEEE Conf. Visualization*, pp. 425–428, 1997.
[141] Y. Song, J. Ye, N. Svakhine, S. Lasher-Trapp, M. Baldwin, and D. Ebert, "An atmospheric visual analysis and exploration system," *IEEE Trans. Visualization and Computer Graphics*, vol. 12, no. 5, pp. 1157–1164, 2006.
[142] L. Treinish, "A function-based data model for visualization," *Proc. IEEE Conf. Visualization*, pp. 73–76, 1998.
[143] W. Cai and G. Sakas, "Data intermixing and multivolume rendering," *Computer Graphics Forum*, vol. 18, no. 3, pp. 359–368, 1999.
[144] S. Grimm, S. Bruckner, A. Kanitsar, and M. E. Gröller, "Flexible direct multi-volume rendering in interactive scenes," *Proc. Vision, Modeling, and Visualization (VMV)*, pp. 386–379, 2004.
[145] J. Plate, T. Holtkämper, and B. Fröhlich, "A flexible multi-volume shader framework for arbitrarily intersecting multi-resolution datasets," *IEEE Trans. Visualization and Computer Graphics*, vol. 13, no. 6, pp. 1584–1591, 2007.
[146] S. Lindholm, P. Ljung, M. Hadwiger, and A. Ynnerman, "Fused multi-volume DVR using binary space partitioning," *Computer Graphics Forum*, vol. 28, no. 3, pp. 847–854, 2009.
[147] M. Burns, M. Haidacher, W. Wein, I. Viola, and M. E. Gröller, "Feature emphasis and contextual cutaways for multimodal medical visualization," *Proc. Symp. Data Visualization (EuroVis)*, pp. 275–282, 2007.

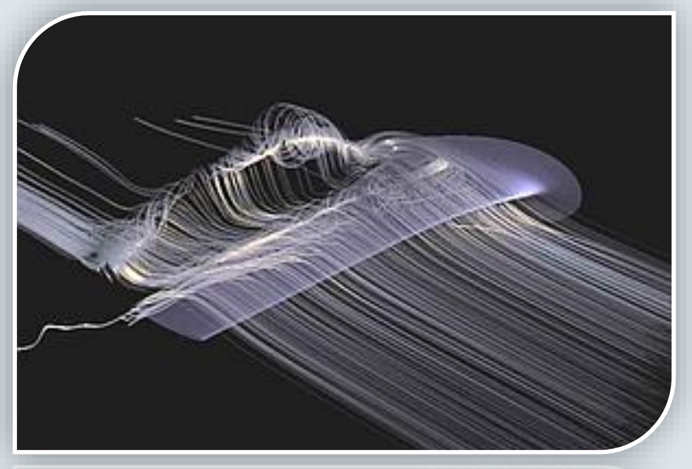
[148] H. Zhou, M. Chen, and M. Webster, "Comparative evaluation of visualization and experimental results using image comparison metrics," *Proc. IEEE Visualization*, pp. 315–322, 2002.
[149] C. Brewer, "Color use guidelines for data representation," *Proc. Section on Statistical Graphics*, pp. 55–60, 1999.
[150] J. Trapp and H.-G. Pagendarm, "Data level comparative visualization in aircraft design," *Proc. IEEE Conf. Visualization*, pp. 393–396, 1996.
[151] N. Sahasrabudhe, J. West, R. Machiraju, and M. Janus, "Structured spatial domain image and data comparison metrics," *Proc. IEEE Conf. Visualization*, pp. 97–515, 1999.
[152] K. Kim, C. Wittenbrink, and A. Pang, "Extended specifications and test data sets for data level comparisons of direct volume rendering algorithms," *IEEE Trans. Visualization and Computer Graphics*, vol. 7, no. 4, pp. 299–317, 2001.
[153] M. Malik, C. Heinzl, and M. E. Gröller, "Comparative visualization for parameter studies of dataset series," *IEEE Trans. Visualization and Computer Graphics*, vol. 16, no. 5, pp. 829–840, 2010.
[154] S. Busking, C. Botha, L. Ferrarini, J. Milles, and F. Post, "Image-based rendering of intersecting surfaces for dynamic comparative visualization," *The Visual Computer*, vol. 27, pp. 347–363, 2011.
[155] H.-G. Pagendarm and F. Post, "Studies in comparative visualization of flow features," *Scientific Visualization: Overviews, Methodologies, and Techniques*, pp. 211–227, 1997.
[156] C. Weigle and R. Taylor II, "Visualizing intersecting surfaces with nested-surface techniques," *Proc. IEEE Conf. Visualization*, pp. 503–510, 2005.
[157] A. Pang, C. Wittenbrink, and S. Lodha, "Approaches to uncertainty visualization," *The Visual Computer*, vol. 13, no. 8, pp. 370–390, 1997.
[158] C. Johnson and A. Sanderson, "A next step: Visualizing errors and uncertainty," *IEEE Computer Graphics and Applications*, vol. 23, no. 5, pp. 6–10, 2003.
[159] H. Griethe and H. Schumann, "The visualization of uncertain data: methods and problems," *Proc. Simulation and Visualization (SimVis)*, pp. 143–156, 2006.
[160] A. MacEachren, A. Robinson, S. Hopper, S. Gardner, R. Murray, M. Gahegan, and E. Hetzler, "Visualizing geospatial information uncertainty: What we know and what we need to know," *Cartography and Geographic Information Science*, vol. 32, no. 3, pp. 139–160, 2005.
[161] R. Cooke and J. van Noortwijk, "Graphical Methods for Uncertainty and Sensitivity Analysis," *Sensitivity Analysis*, Saltelli, Chan, and Scott, Eds., pp. 245–266. Wiley, 2000.
[162] R. McGill, J. Tukey, and W. Larsen, "Variations of box plots," *The American Statistician*, vol. 32, pp. 12–16, 1978.
[163] D. Kao, J. Dungan, and A. Pang, "Visualizing 2D probability distributions from EOS satellite image-derived data sets: a case study," *Proc. IEEE Conf. Visualization*, pp. 457–460, 2001.
[164] D. Kao, A. Luo, J. Dungan, and A. Pang, "Visualizing spatially varying distribution data," *Proc. Int'l. Conf. Information Visualization (IV)*, pp. 219–225, 2002.
[165] J. Sanyal, S. Zhang, J. Dyer, A. Mercer, P. Amburn, and R. Moorhead, "Noodles: A tool for visualization of numerical weather model ensemble uncertainty," *IEEE Trans. Visualization and Computer Graphics*, vol. 16, no. 6, pp. 1421–1430, 2010.
[166] Y.-H. Chan, C. Correa, and K.-L. Ma, "Flow-based scatterplots for sensitivity analysis," *Proc. IEEE Conf. Visual Analytics Science and Technology (VAST)*, pp. 43–50, 2010.
[167] U. D. Bordoloi, D. L. Kao, and H.-W. Shen, "Visualization techniques for spatial probability density function data," *Data Science J.*, vol. 3, pp. 153–162, 2004.
[168] S. Bruckner and T. Möller, "Result-driven exploration of simulation parameter spaces for visual effects design," *IEEE Trans. Visualization and Computer Graphics*, vol. 16, no. 6, pp. 1468–1476, 2010.
[169] C. Correa, Y.-H. Chan, and K.-L. Ma, "A framework for uncertainty-aware visual analytics," *Proc. IEEE Symp. Visual Analytics Science and Technology (VAST)*, pp. 51–58, 2009.
[170] J. van Wijk and R. van Liere, "HyperSlice: visualization of scalar functions of many variables," *Proc. IEEE Conf. Visualization*, pp. 119–125, 1993.
[171] L. Tweedie, R. Spence, H. Dawkes, and H. Su, "Externalising abstract mathematical models," *Proc. Human Factors in Computing Systems (CHI 96)*, pp. 406–412, 1996.
[172] K. Matković, D. Gračanin, M. Jelović, and H. Hauser, "Interactive visual steering – rapid visual prototyping of a common rail injection system," *IEEE Trans. Visualization and Computer Graphics*, vol. 14, no. 6, pp. 1699–1706, 2008.
[173] K. Matković, D. Gračanin, M. Jelović, A. Ammer, A. Lež, and H. Hauser, "Interactive visual analysis of multiple simulation runs using the simulation model view," *IEEE Trans. Visualization and Computer Graphics*, vol. 16, no. 6, pp. 1449–1457, 2010.
[174] A. Unger and H. Schumann, "Visual support for the understanding of simulation processes," *Proc. IEEE Pacific Visualization Symp.*, pp. 57–64, 2009.
[175] M. Steinberger, M. Waldner, M. Streit, A. Lex, and D. Schmalstieg, "Context-preserving visual links," *IEEE Trans. Visualization and Computer Graphics*, vol. 17, no. 12, pp. 2249–2258, 2011.
[176] T. May, "Modelle und Methoden für die Kopplung automatischer und visuell-interaktiver Verfahren für die Datenanalyse," PhD dissertation, Dept. of Informatics, TU Darmstadt, 2011, (in German).
[177] T. Nocke, T. Sterzel, M. Böttinger, and M. Wrobel, "Visualization of climate and climate change data: An overview," *Digital Earth Summit on Geoinformatics*, pp. 226–232, 2008.
[178] J. Kehrer, "Interactive visual analysis of multi-faceted scientific data," PhD dissertation, Dept. of Informatics, Univ. of Bergen, Norway, 2011.

Johannes Kehrer received the MSc degree in 2007 from the Vienna University of Technology, Austria, and the PhD degree in 2011 from the University of Bergen, Norway, where he worked at the Department of Informatics. He is currently a researcher at the VRVis Research Center in Vienna, Austria. His interests include the interactive visual analysis of multi-run, multi-variate and spatio-temporal data (mainly from the application fields of climate research and engineering). He has cooperated with several climate and meteorological institutes.



Helwig Hauser received the graduate and doctoral degree in 1995 and 1998, respectively, from the Vienna University of Technology, Austria, where he worked as an assistant professor at the Institute of Computer Graphics from 1994 until mid 2000. He then joined the newly founded VRVis Research Center in Vienna, Austria, as a key researcher in the field of visualization. In 2003, Dr. Hauser became the scientific director of VRVis (www.VRVis.at). Since 2007, he is a professor at the University of Bergen, Norway, within a newly established research group on visualization (www.ii.UiB.no/vis). His interests are diverse in visualization and computer graphics, including interactive visual analysis, illustrative visualization, the application of visualization to various domain problems, and the combination of scientific and information visualization. He is a member of the IEEE.

SIEMENS
Ingenuity for life



Visual Analysis of Scientific Data

Dr. Johannes Kehrer

Disclaimer

These lecture slides and materials are for personal use by students and are intended for educational purposes only. Distribution or reproduction of the material is prohibited.

The views and opinions expressed by any individual during the course of these lecturers are their own and do not necessarily represent the views, opinions, or positions of Siemens or the Technical University of Munich.

Although reasonable efforts have been made to ensure the information in these lecture slides is current and accurate, we do not assume any responsibility for the content of the posted material.

These slides also contain links to websites of third parties. As the content of these websites is not under our control, we cannot assume any liability for such external content.

Visualization – Major Areas

- Major areas
  - Volume Visualization
  - Flow Visualization
  - Information Visualization
  - Visual Analytics

Inherent spatial reference

Scientific Visualization

3D

nD

Usually no spatial reference

Visualization – Goals

SIEMENS
Ingenuity for life

Visualization is good for

– **Visual exploration**
  - find unknown/unexpected
  - generate new hypotheses

Nothing is known about the data

– **Visual analysis** (confirmative vis.)
  - verify or reject hypotheses
  - information drill-down

There are hypotheses

– **Presentation**
  - show/communicate results

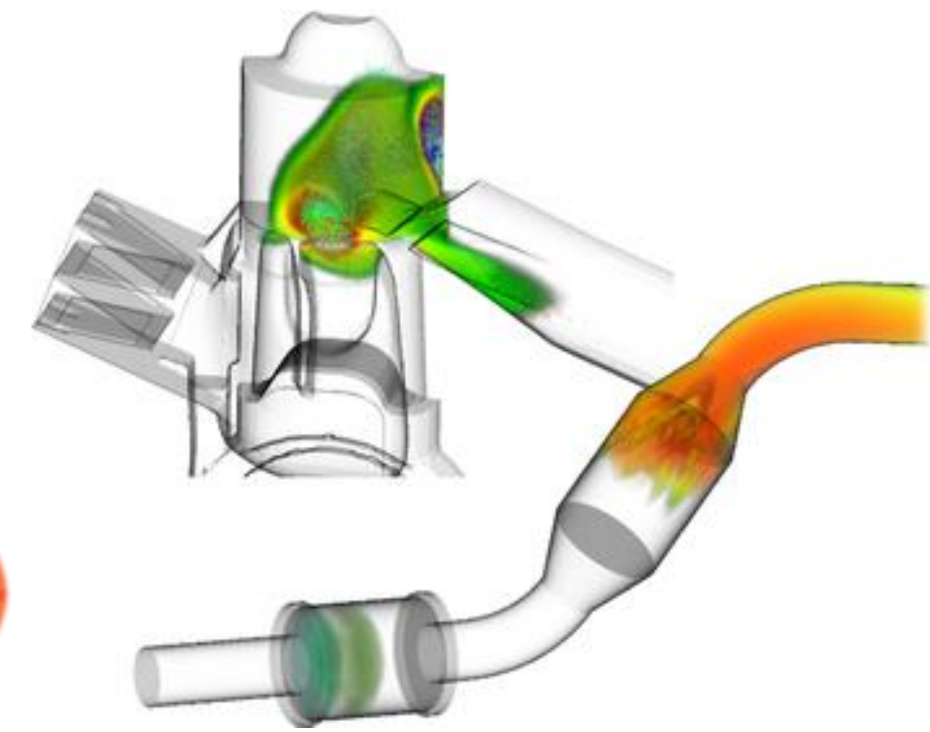
"Everything" is known

Motivation

- Large amounts of data are generated everywhere

medical scanners

global climate simulations



automotive engineering

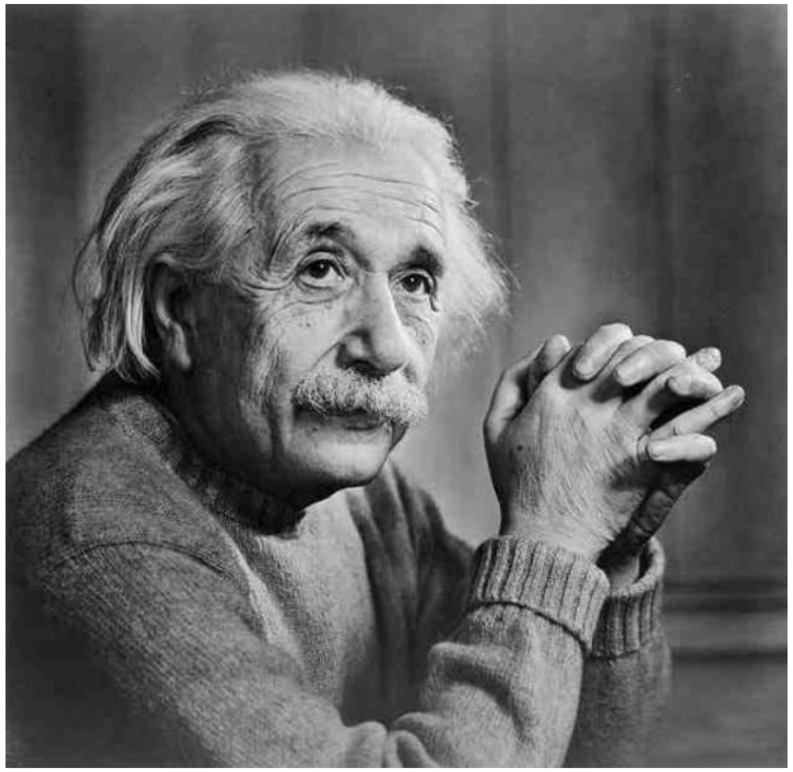
- Data are increasing in complexity & variability

Problems

- Data size / complexity / variable quality
- Various / heterogeneous data sources
- Data is explicit, but information/insight is often implicit
- Missing involvement of users and their tasks
- Pure visualization methods are not enough for large & complex data
- Pure automatic methods only work for well-defined & clearly specified problems

Humans & Computers

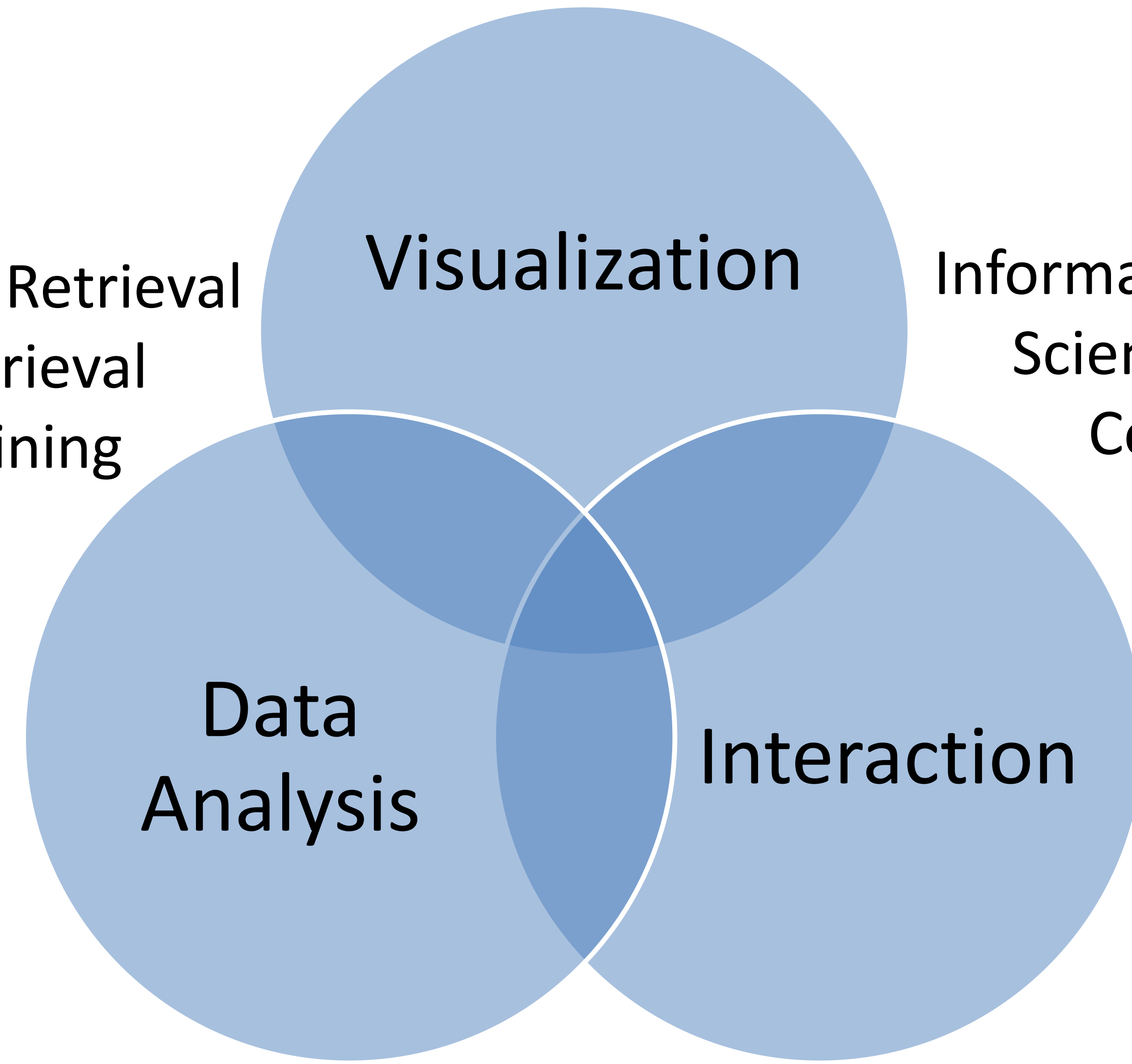
"Computers are incredibly fast, accurate, and stupid; humans are incredibly slow, inaccurate, and brilliant; together they are powerful beyond imagination."



Attributed to Albert Einstein

Visual Analytics / Analysis

SIEMENS
Ingenuity for life



Information Retrieval
Data Retrieval
Data Mining

Information Visualization
Scientific Visualization
Computer Graphics

Human-Computer Interaction
Cognitive Psychology
Perception

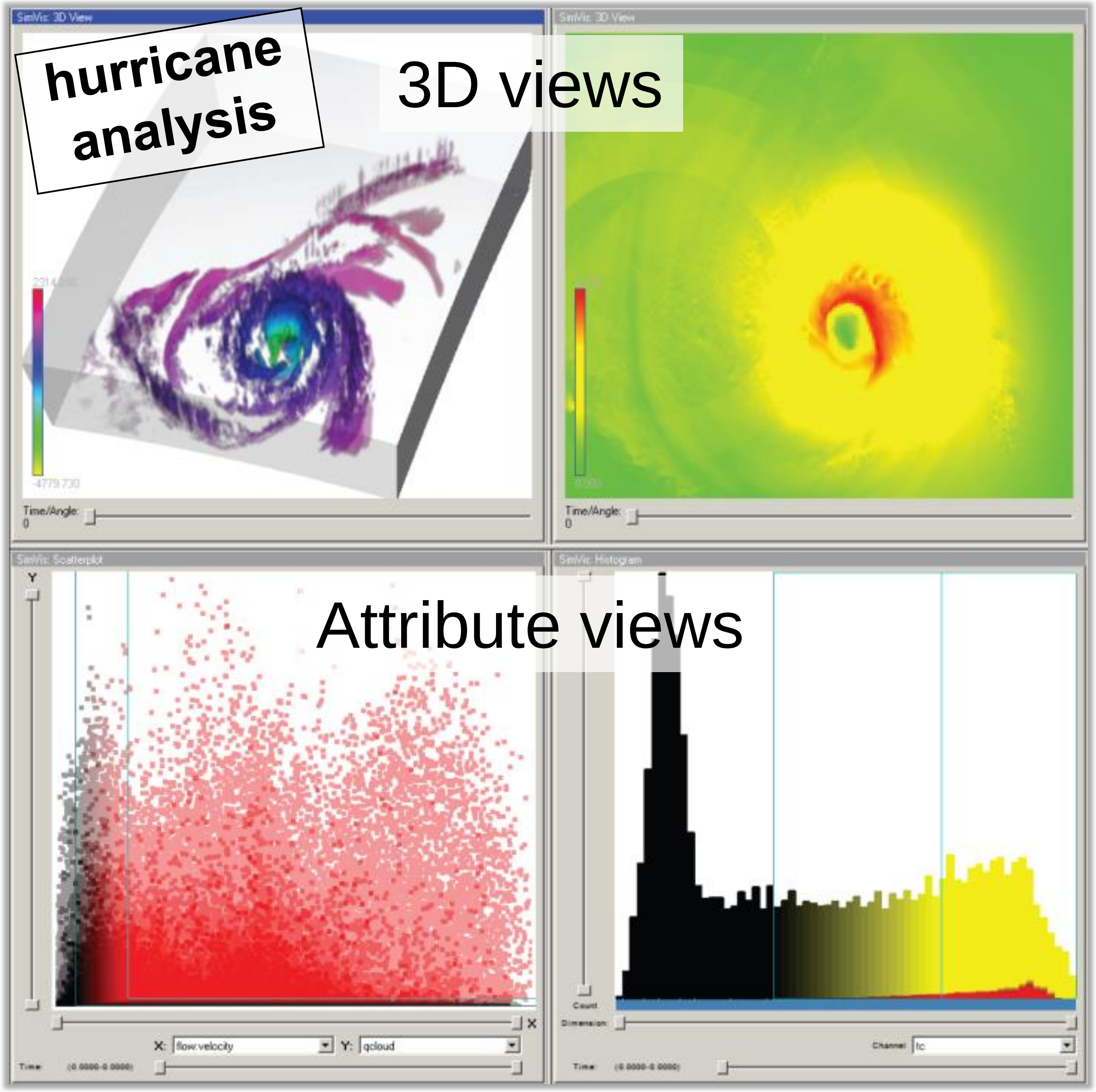
Adapted from Keim

Visual Analysis of Scientific Data

SIEMENS
Ingenuity for life

Combines **computational** & **interactive visual methods**

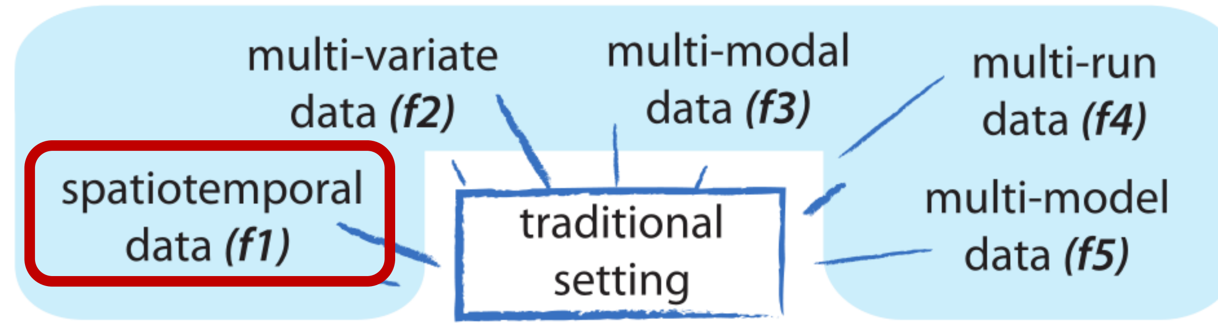
- **Multiple linked views**
- **Interpret** large & complex data
- **Drill-down** into information
- **Find relations**
  ("read between the lines")
- Detect **features/patterns** that are difficult to describe
- Integrate **expert knowledge**

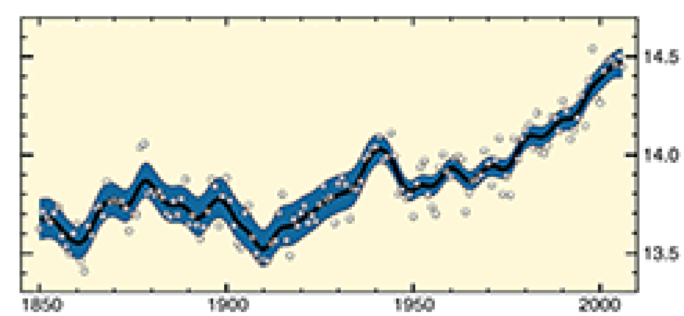


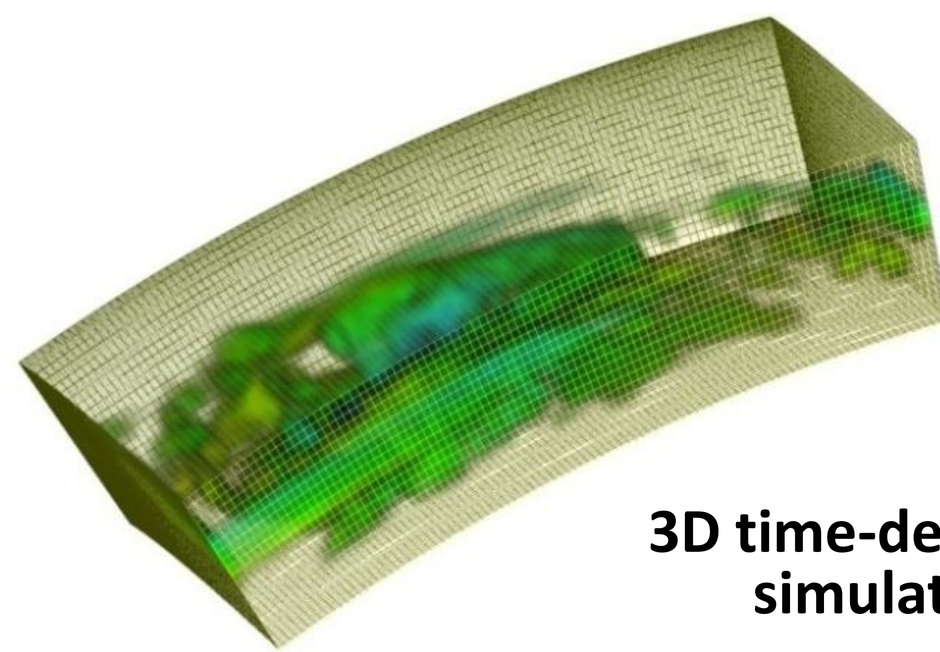
SimVis [Doleisch et al. 03]

Multi-faceted Scientific Data

- Spatiotemporal data
- Multi-variate/multi-field data
  (multiple data attributes, e.g., temperature or pressure)
- Multi-modal data
  (CT, MRI, large-scale measurements, simulations, etc.)
- Multi-run/ensemble
  simulations (repeated with varied parameter settings)
- Multi-model scenarios
  (e.g., coupled climate model)



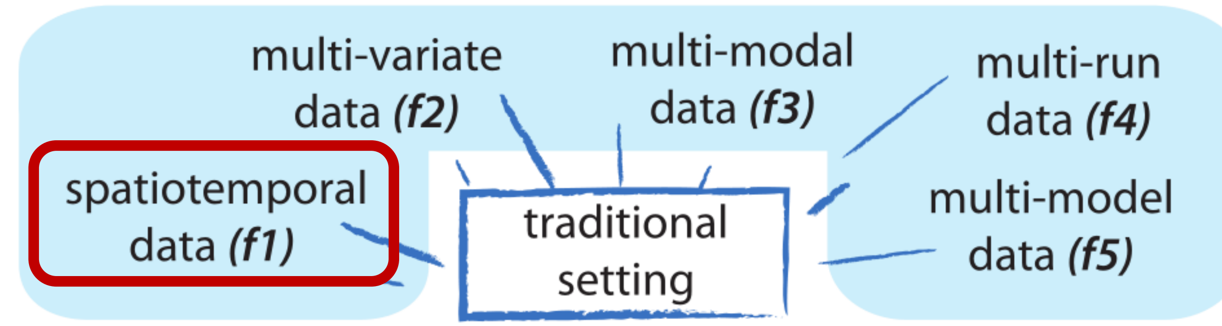


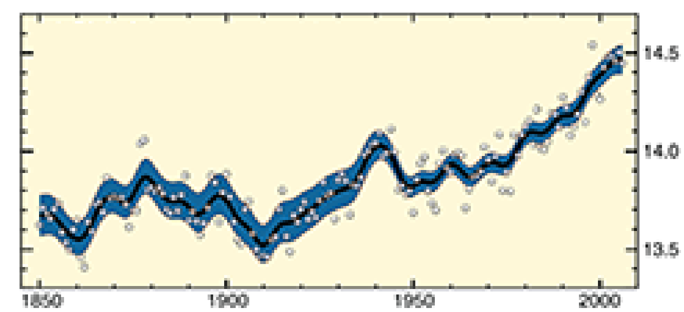


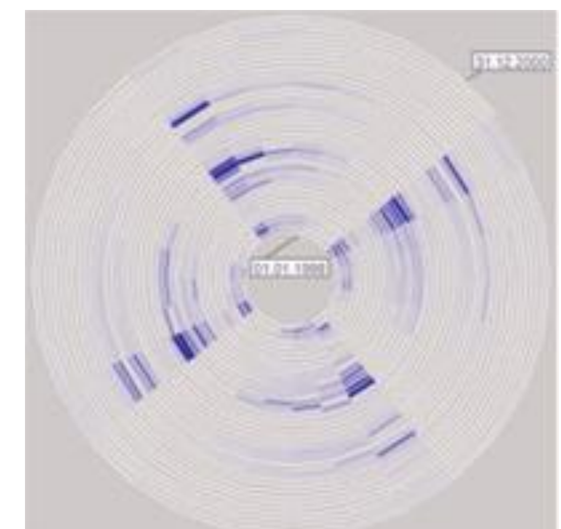
3D time-dependent simulation data

Multi-faceted Scientific Data

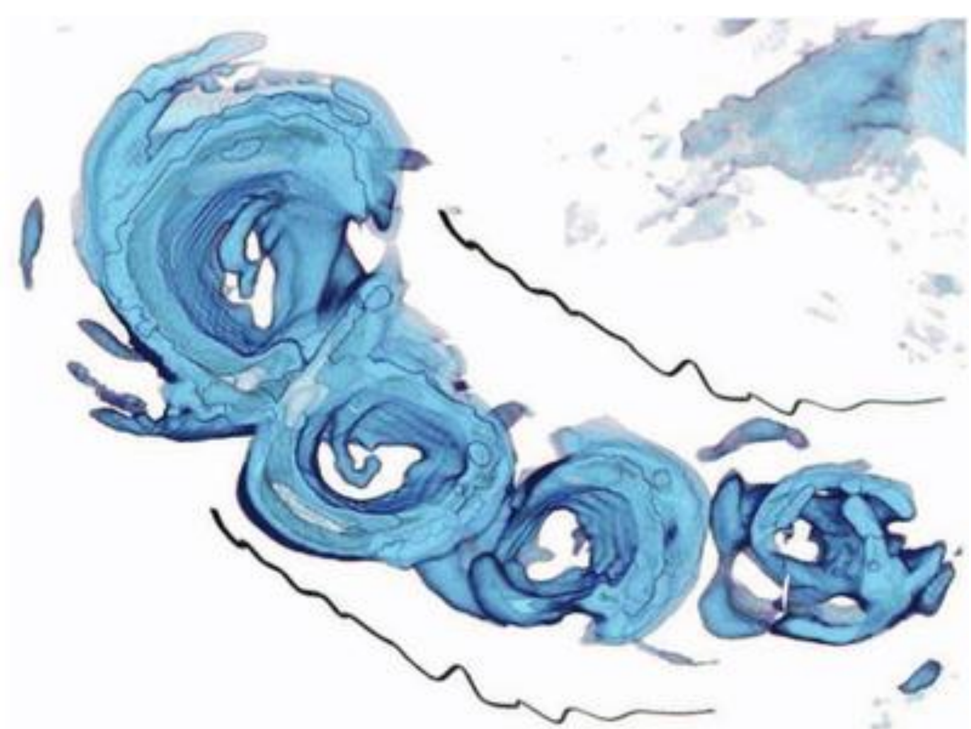
- Spatiotemporal data
  - Cartography, geovis, etc.
  - Linear vs. cyclic time
  - Automatic animations
  - Flow visualization
  - Visualize summary statistics







Spiral Graph
[Weber et al. 01]

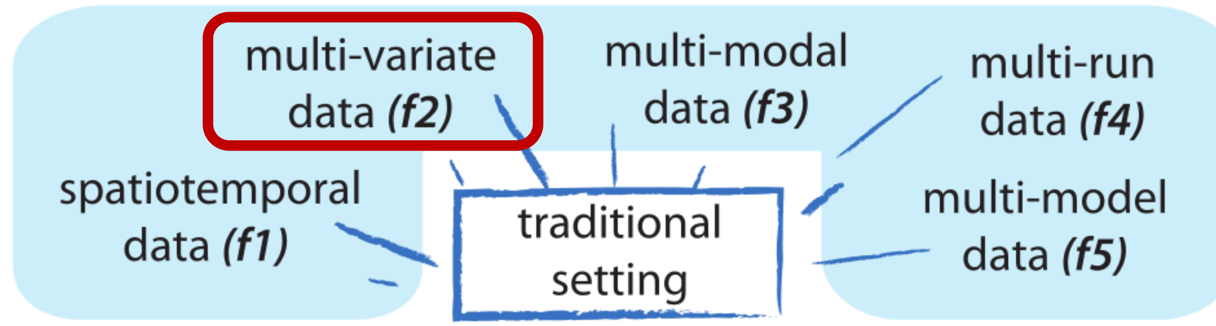


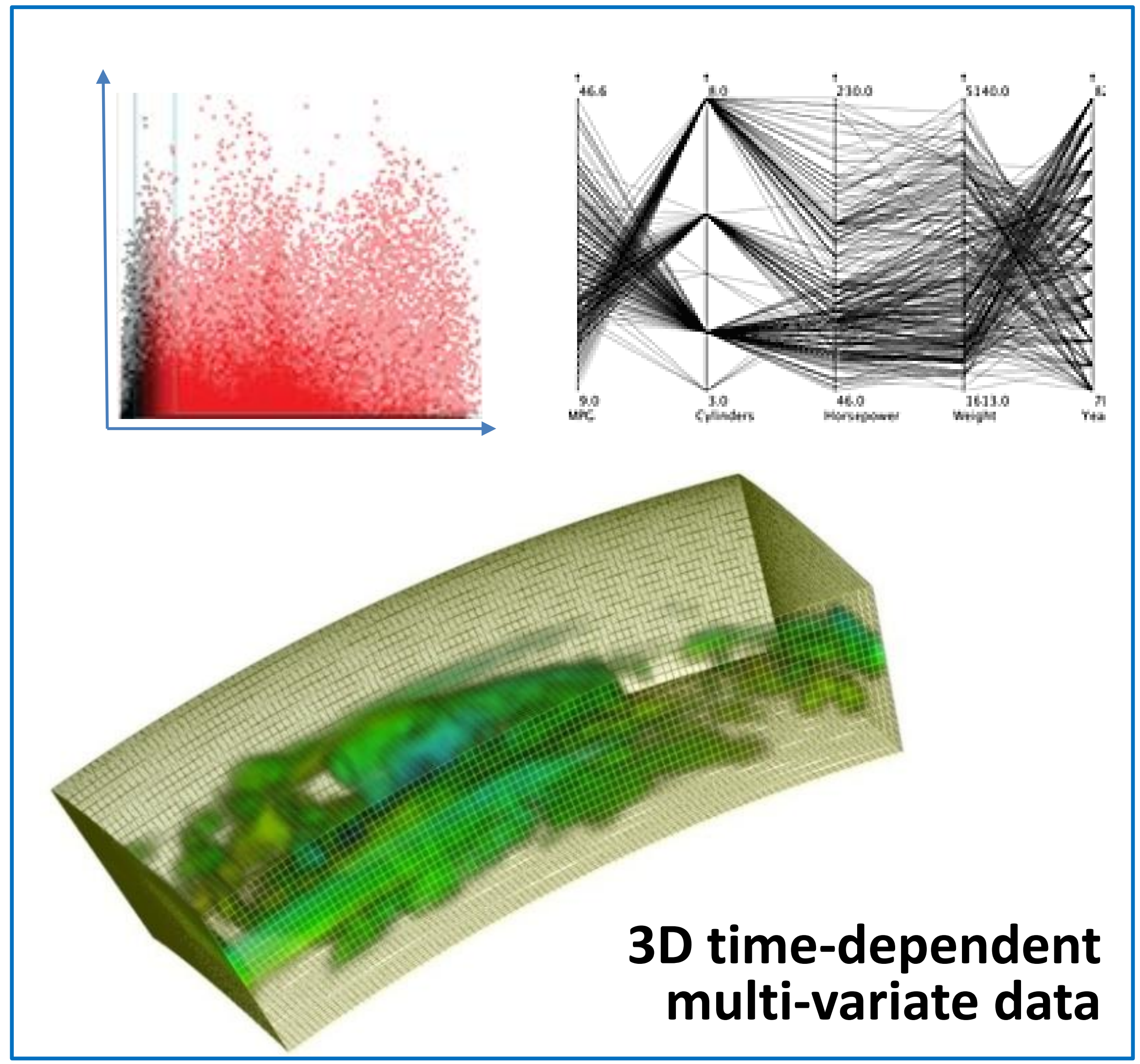
Illustrative techniques
[Joshi et al. 09]

[Hsu et al. 10]

Multi-faceted Scientific Data

- Spatiotemporal data
- Multi-variate/multi-field data
  (multiple data attributes, e.g., temperature or pressure)
- Multi-modal data
  (CT, MRI, large-scale measurements, simulations, etc.)
- Multi-run/ensemble simulations (repeated with varied parameter settings)
- Multi-model scenarios
  (e.g., coupled climate model)

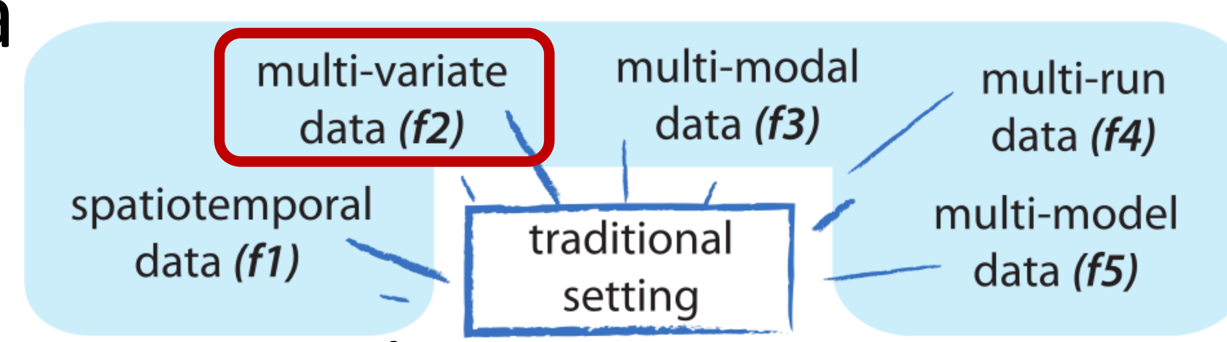




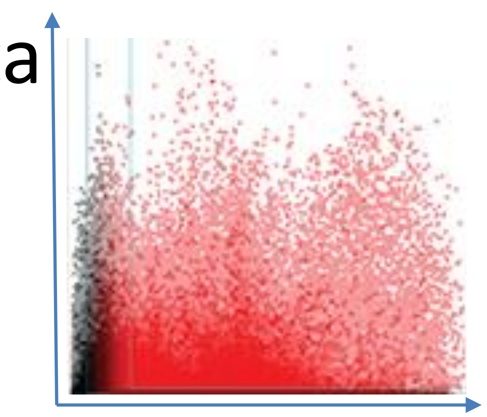
3D time-dependent multi-variate data

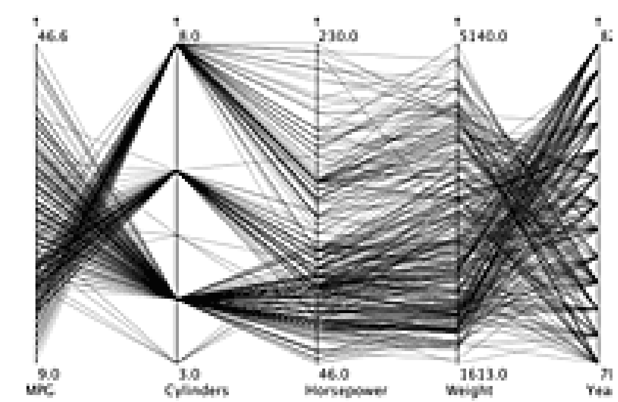
Multi-faceted Scientific Data

- Multi-variate/multi-field data

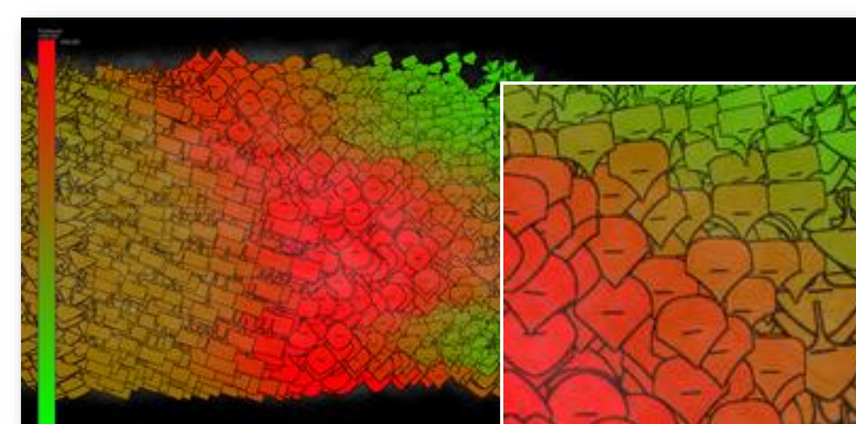


  – Attribute views (scatterplots, parallel coordinates, etc.)
    - Find patterns such as correlations or outliers
    - Lack spatial relationships of data
    - Which of the many data variables to show?





  – Volume rendering
    - Difficult to see multi-variate patterns
    - Layering & glyphs
    - Feature-based vis. (brushing, segmentation, …)

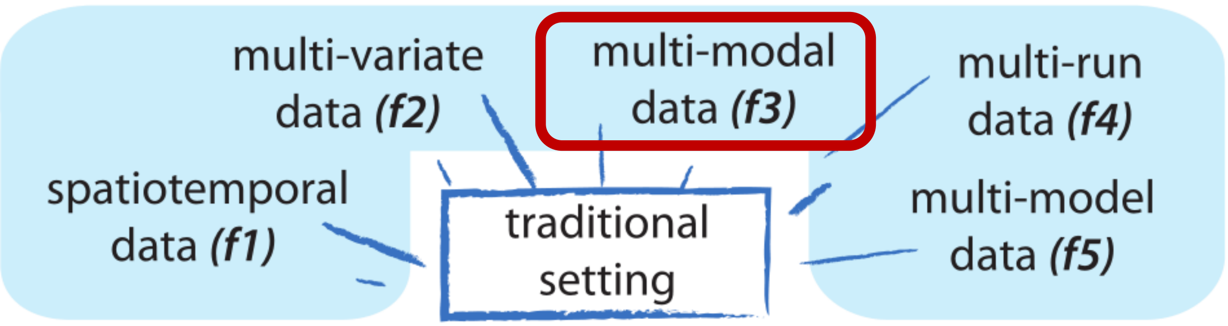


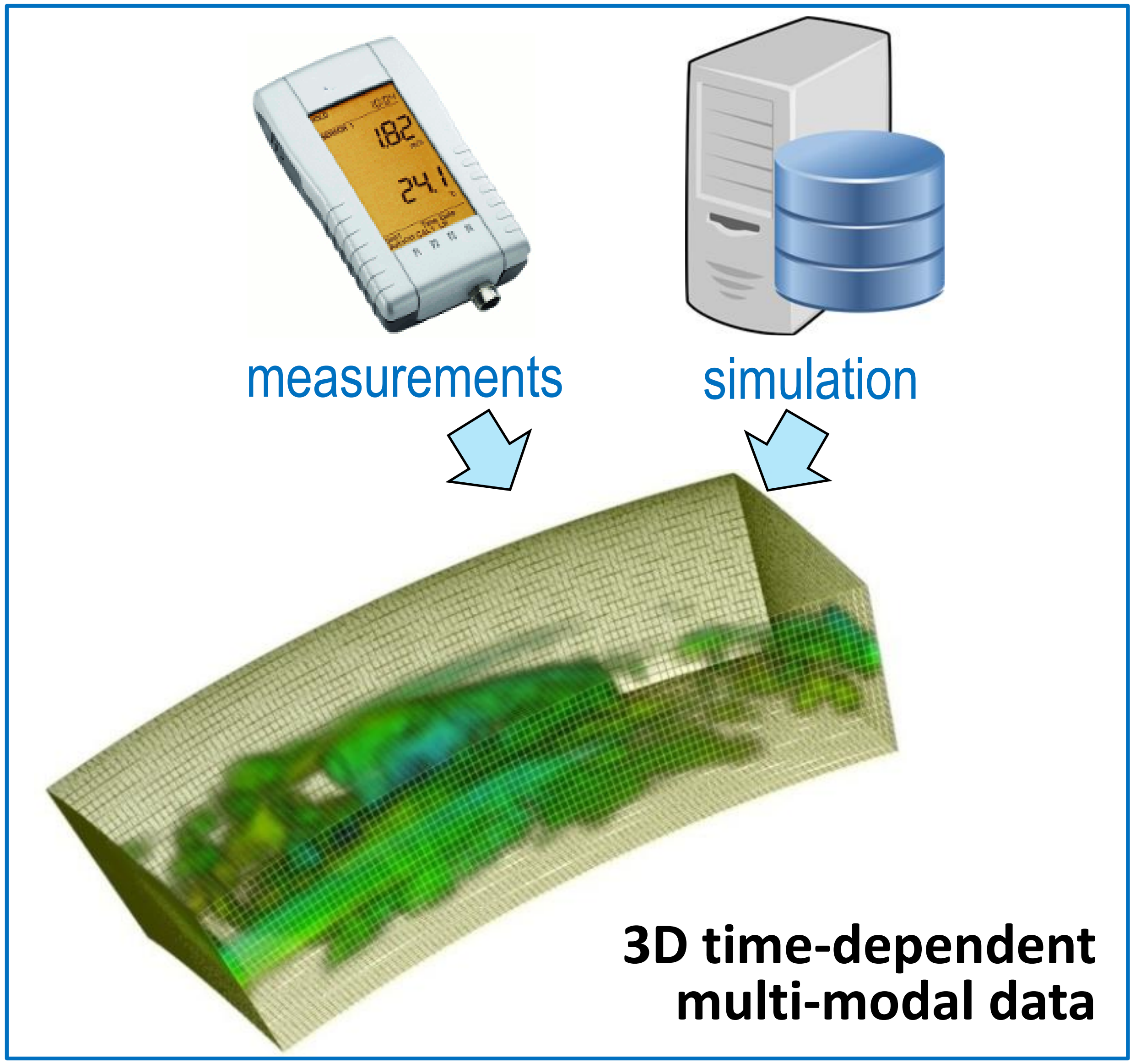
  – Clustering, dimensionality reduction, etc.

Multi-faceted Scientific Data

SIEMENS
Ingenuity for life

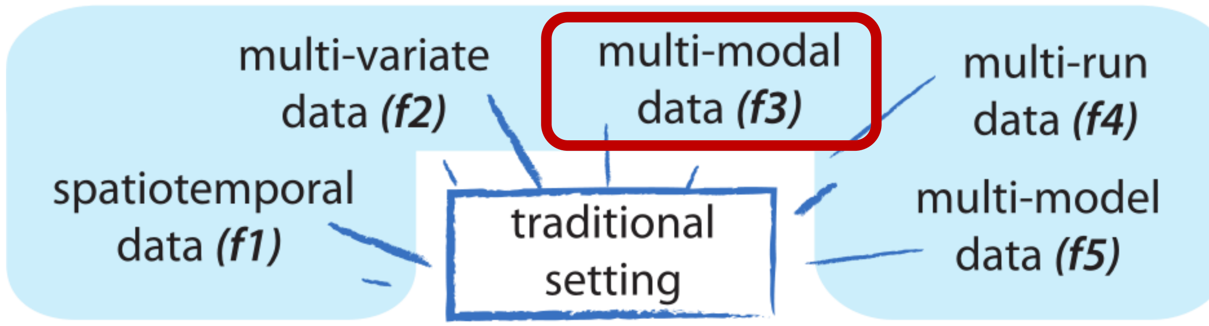
- Spatiotemporal data
- Multi-variate/multi-field data
  (multiple data attributes, e.g., temperature or pressure)
- Multi-modal data
  (CT, MRI, large-scale measurements, simulations, etc.)
- Multi-run/ensemble simulations (repeated with varied parameter settings)
- Multi-model scenarios
  (e.g., coupled climate model)

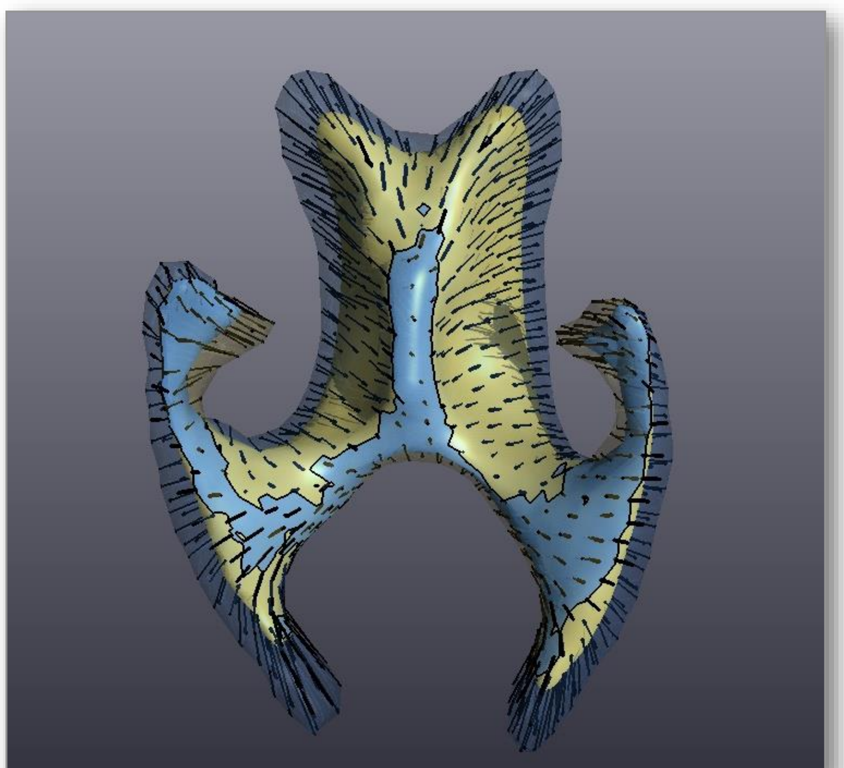




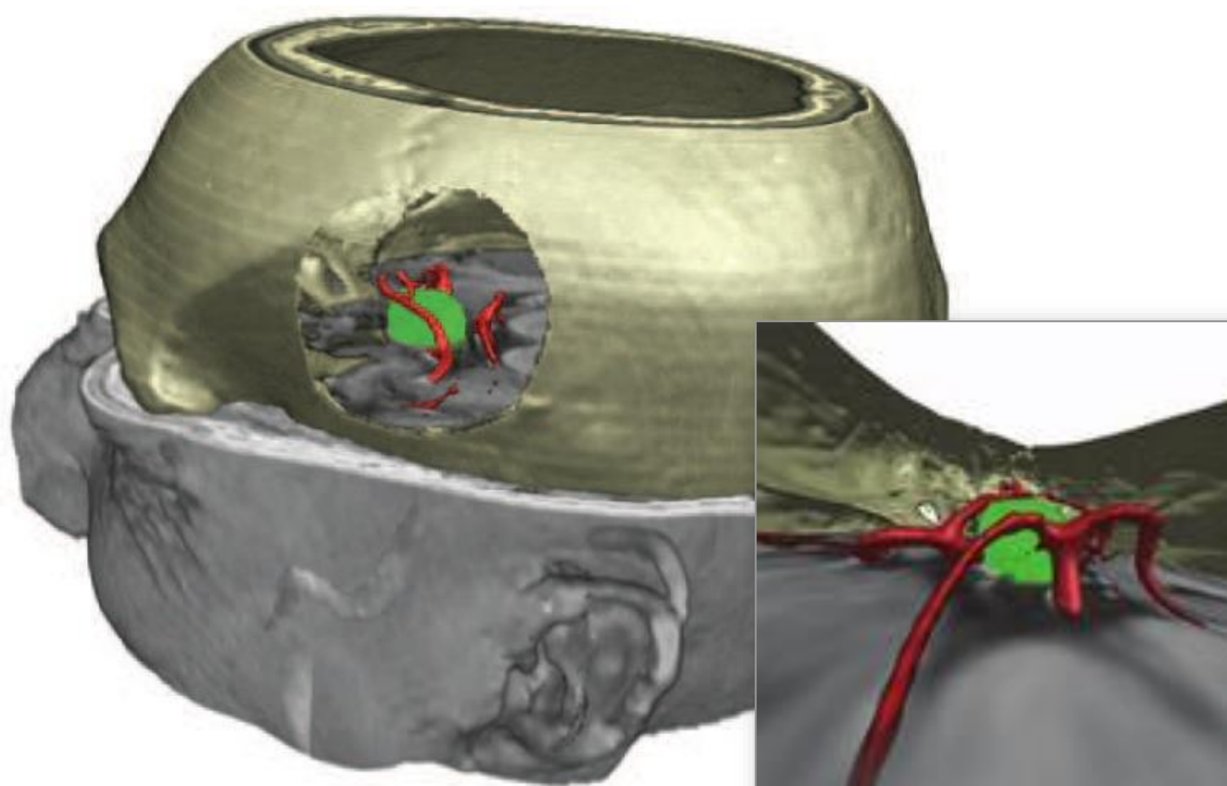
Multi-faceted Scientific Data

- Multi-modal data
  - Various types of grids with different resolution
  - Coregistration & normalization
  - Multi-volume rendering
  - Visual data fusion
  - Comparative visualization





Nested surfaces [Buskin et al. 11]



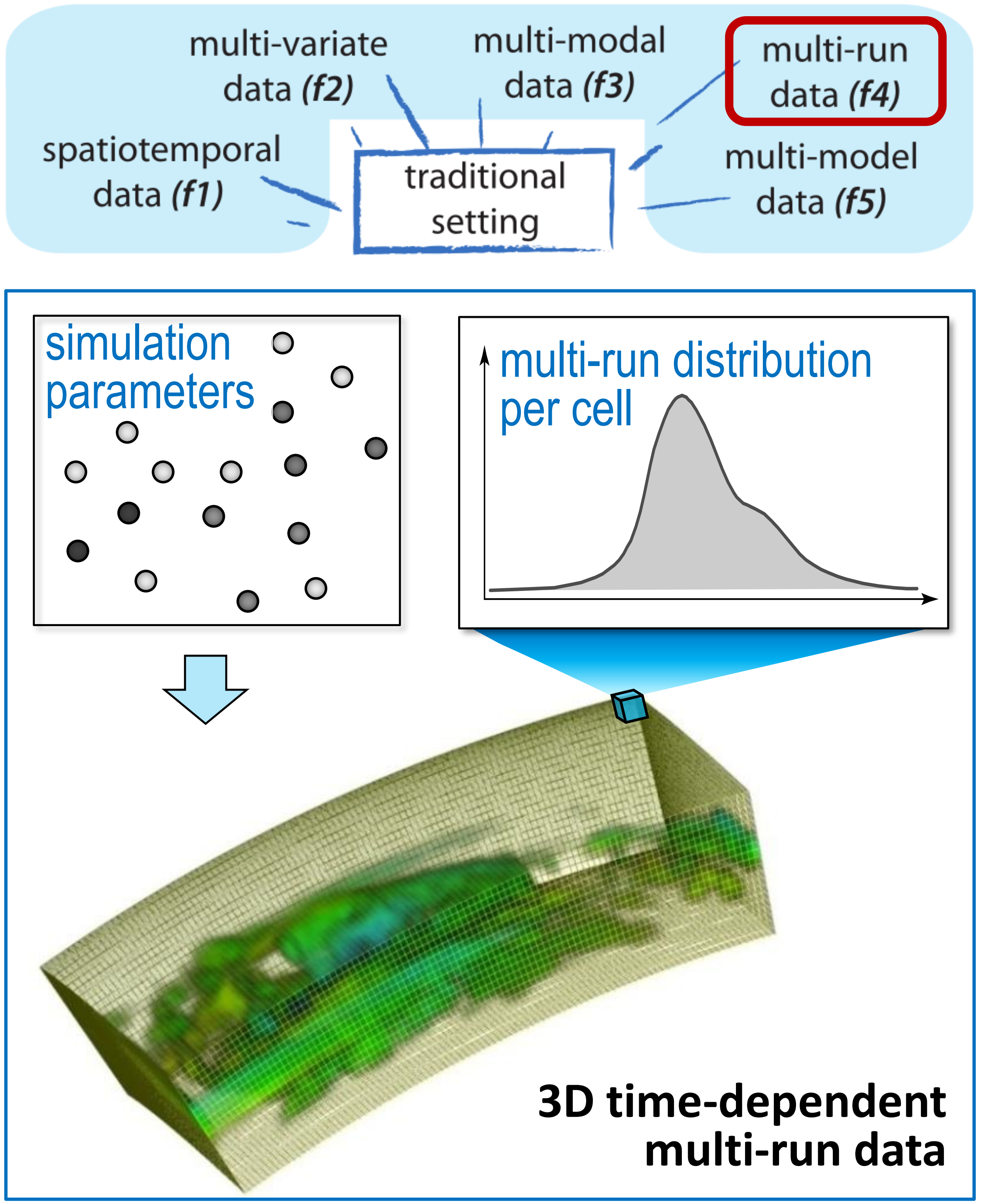
[Beyer et al. 07]

Multi-volume rendering of segmented data (green: tumor - MR, red: vessels – MRA, brown: skull - CT)

Multi-faceted Scientific Data

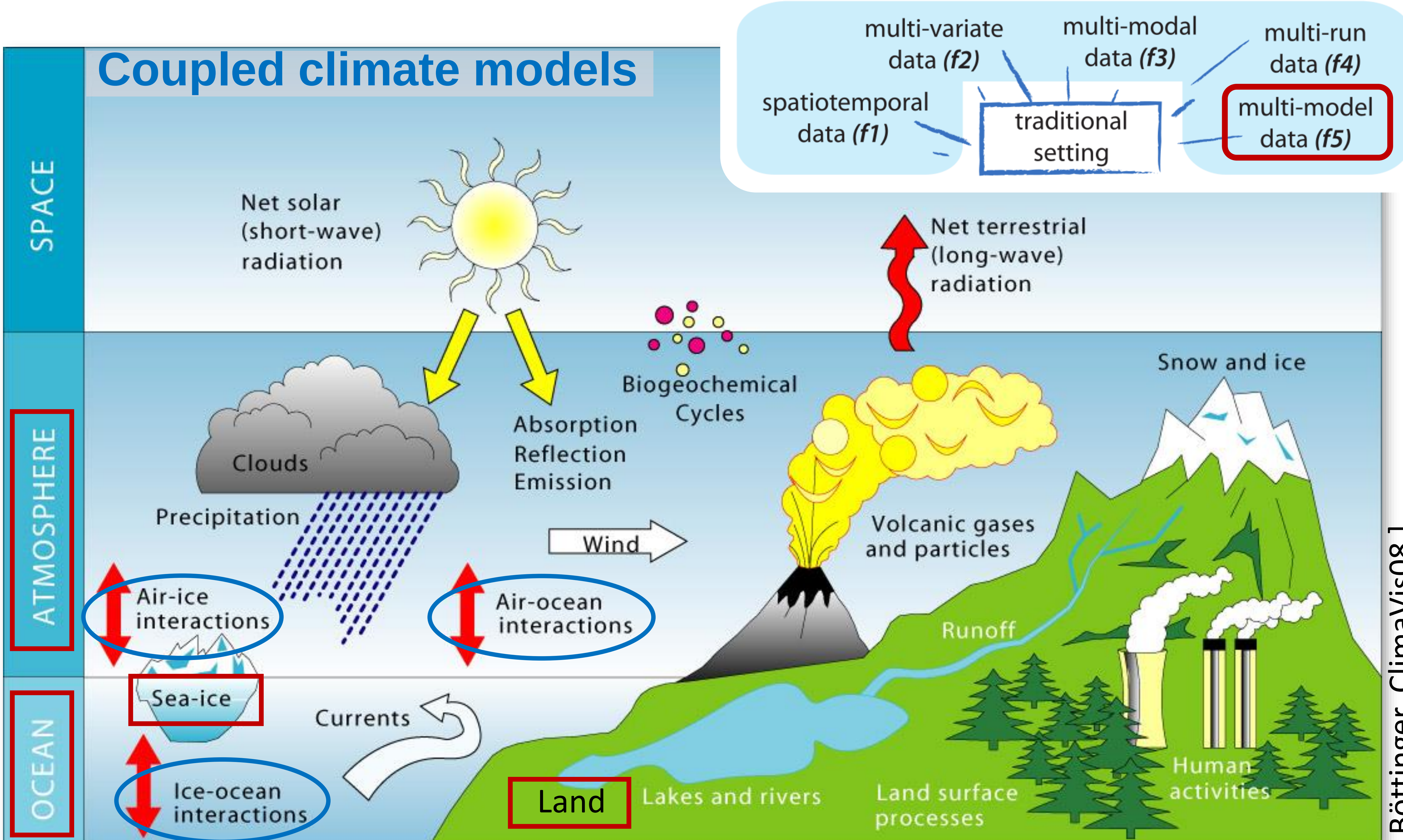
SIEMENS
Ingenuity for life

- Spatiotemporal data
- Multi-variate/multi-field data
  (multiple data attributes, e.g., temperature or pressure)
- Multi-modal data
  (CT, MRI, large-scale measurements, simulations, etc.)
- **Multi-run/ensemble data**
  (simulation repeated with varied parameter settings)
- Multi-model scenarios
  (e.g., coupled climate model)



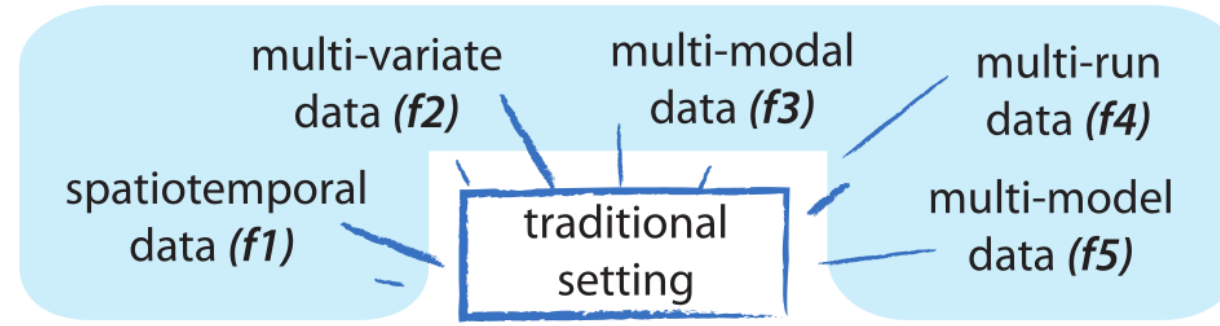
Multi-faceted Scientific Data

SIEMENS
Ingenuity for life

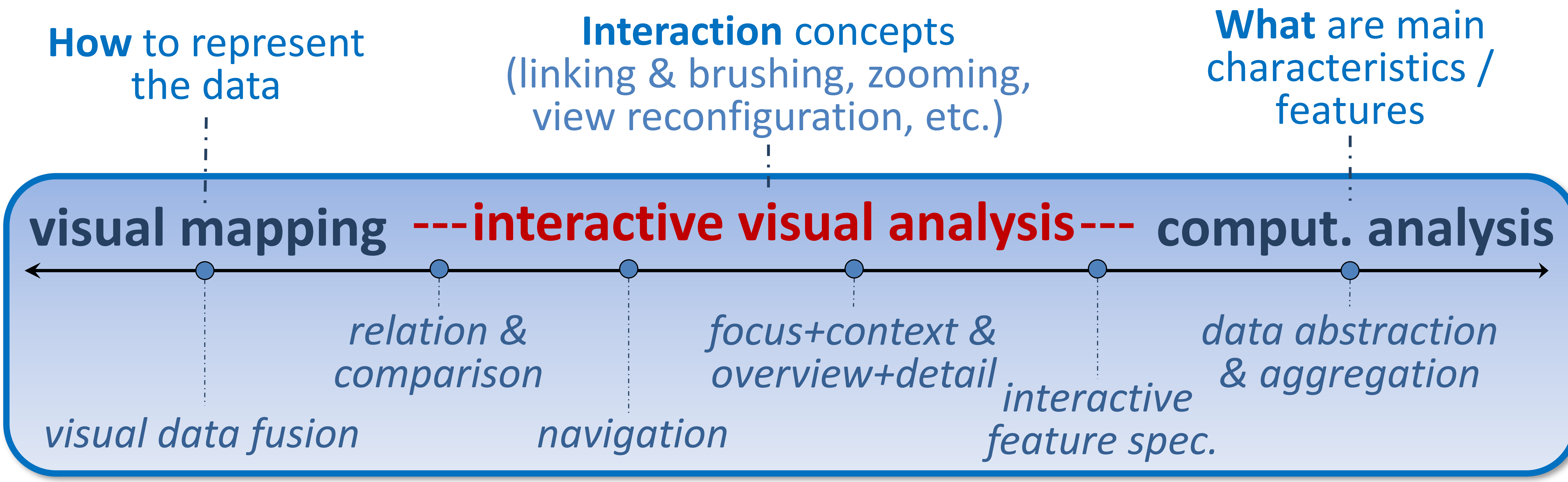


[Böttinger, ClimaVis08]

Categorization

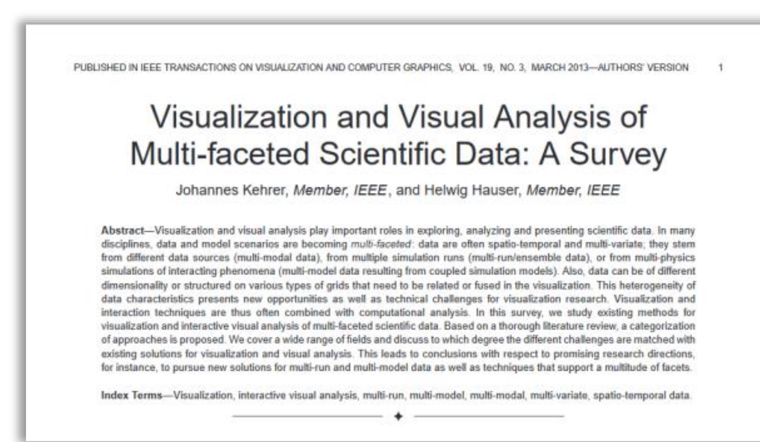


- Literature review of 200+ papers on scientific data
- How are visualization, interaction, & comput. analysis combined?



Compare to Keim et al. 09,
Bertine & Lalanne 09

[Kehrer & Hauser 13]

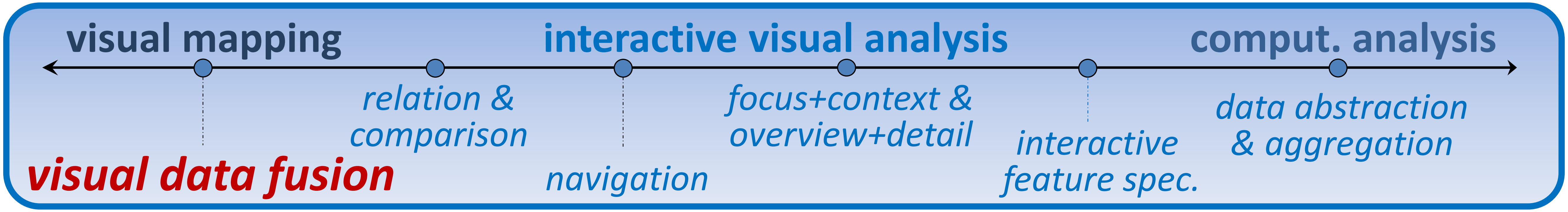

PUBLISHED IN IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS, VOL. 19, NO. 3, MARCH 2013—AUTHORS' VERSION 1

Visualization and Visual Analysis of Multi-faceted Scientific Data: A Survey

Johannes Kehrer, *Member, IEEE*, and Helwig Hauser, *Member, IEEE*

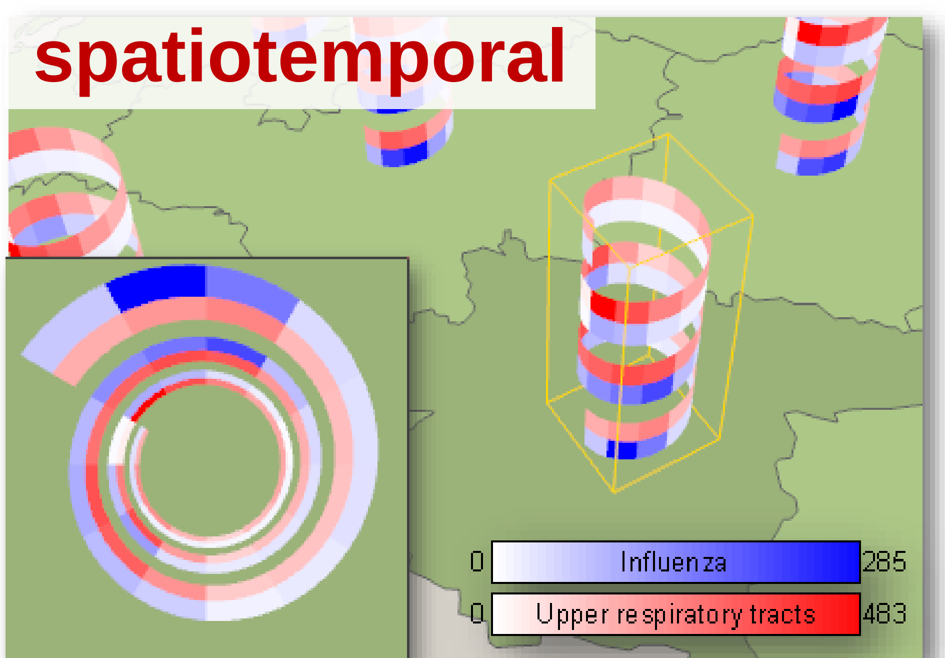
Abstract—Visualization and visual analysis play important roles in exploring, analyzing and presenting scientific data. In many disciplines, data and model scenarios are becoming *multi-faceted*: data are often spatio-temporal and multi-variate; they stem from different data sources (multi-modal data), from multiple simulation runs (multi-run/ensemble data), or from multi-physics simulations of interacting phenomena (multi-model data resulting from coupled simulation models). Also, data can be of different dimensionality or structured on various types of grids that need to be related or fused in the visualization. This heterogeneity of data characteristics presents new opportunities as well as technical challenges for visualization research. Visualization and interaction techniques are thus often combined with computational analysis. In this survey, we study existing methods for visualization and interactive visual analysis of multi-faceted scientific data. Based on a thorough literature review, a categorization of approaches is proposed. We cover a wide range of fields and discuss to which degree the different challenges are matched with existing solutions for visualization and visual analysis. This leads to conclusions with respect to promising research directions, for instance, to pursue new solutions for multi-run and multi-model data as well as techniques that support a multitude of facets.

Index Terms—Visualization, interactive visual analysis, multi-run, multi-model, multi-modal, multi-variate, spatio-temporal data.

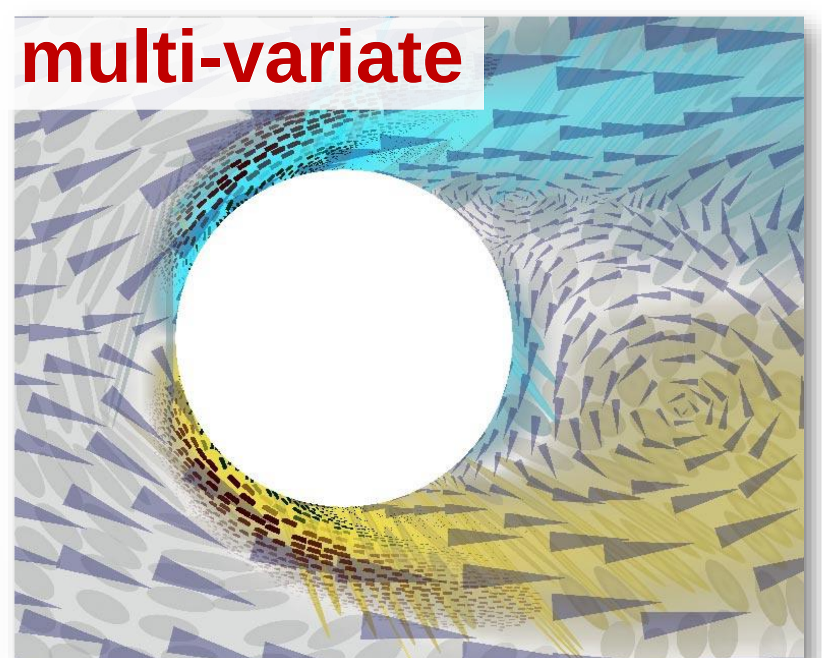


Fusion within a single visualization

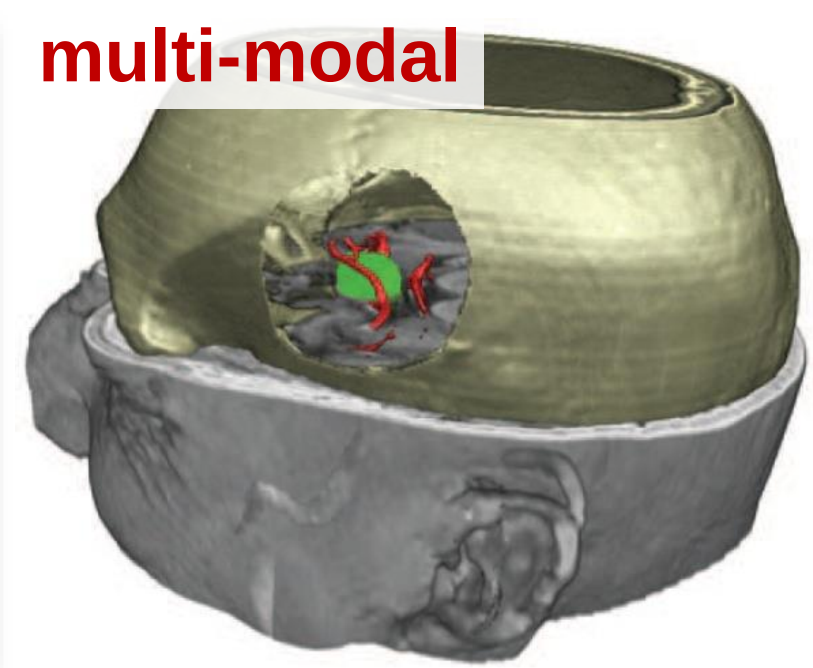
- Use a common frame of reference (e.g., axes)
- Layering techniques (e.g., glyphs, color, transparency)
- Multi-volume rendering (coregistration, segmentation)



Helix glyphs [Tominski et al. 05]

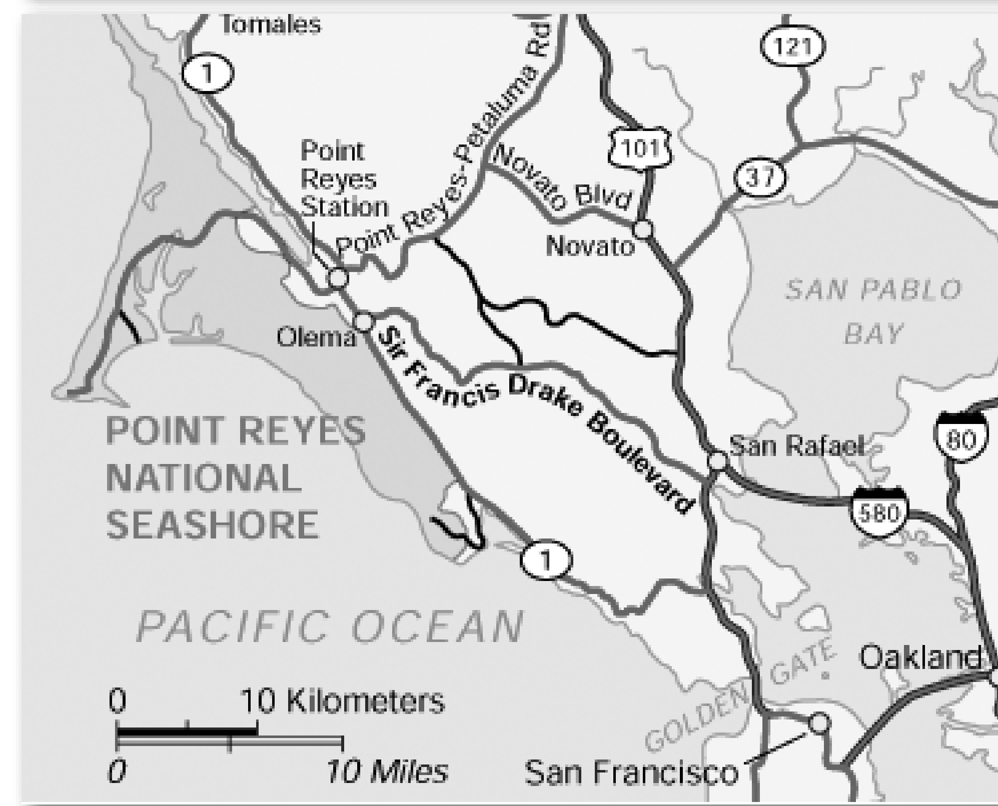


Layering [Kirby et al. 99]

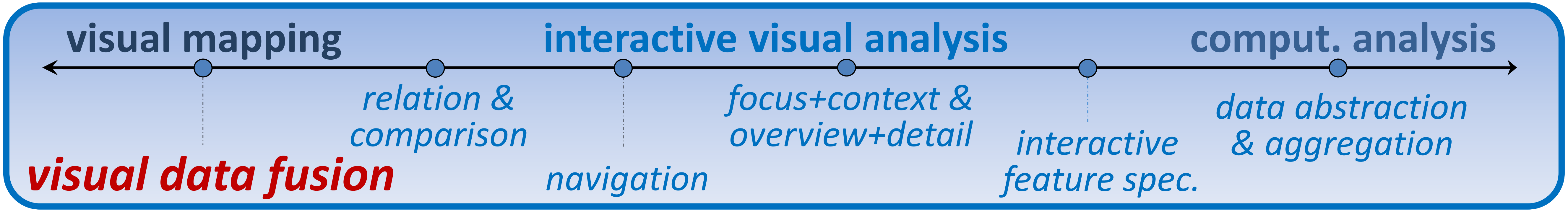


Multi-volume rendering [Beyer et al. 07]

- Layering techniques
  - How many layers?
  - How are layers distinguishable?
- Distinguishable layers
  - Encode with different, nonoverlapping visual channels
  - Foreground layer: roads
    - Hue, size → main from minor
    - Luminance contrast from background
  - Background layer: regions
    - Desaturated colors for water / land

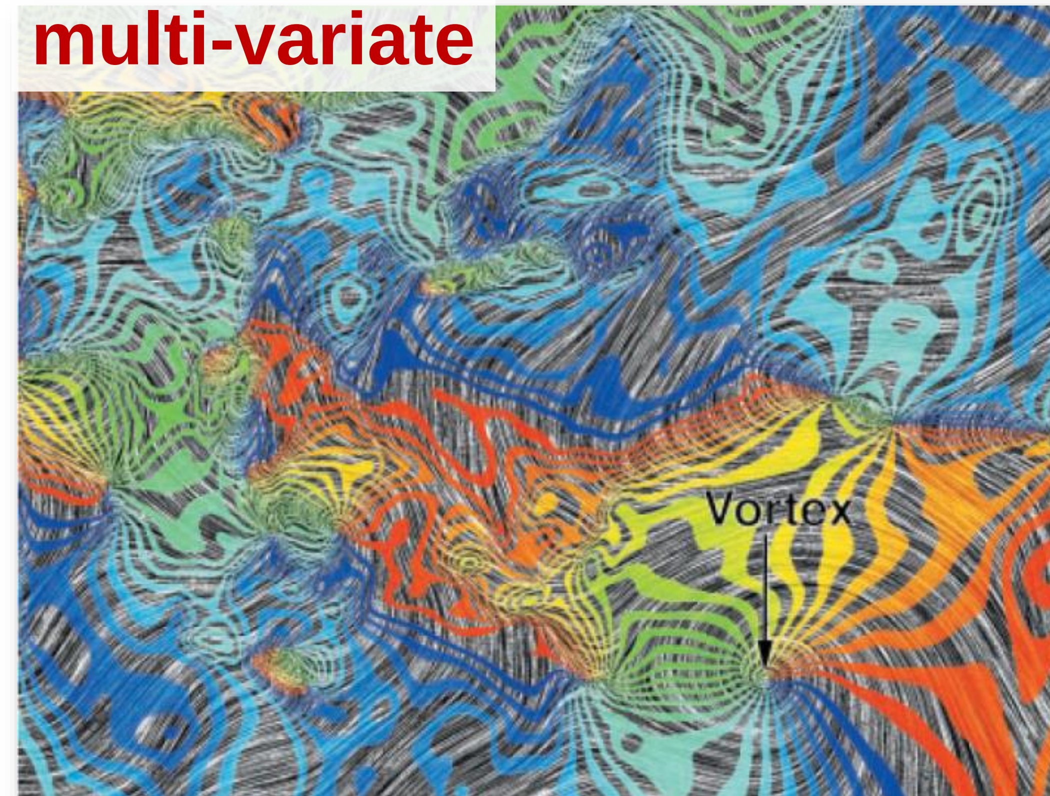


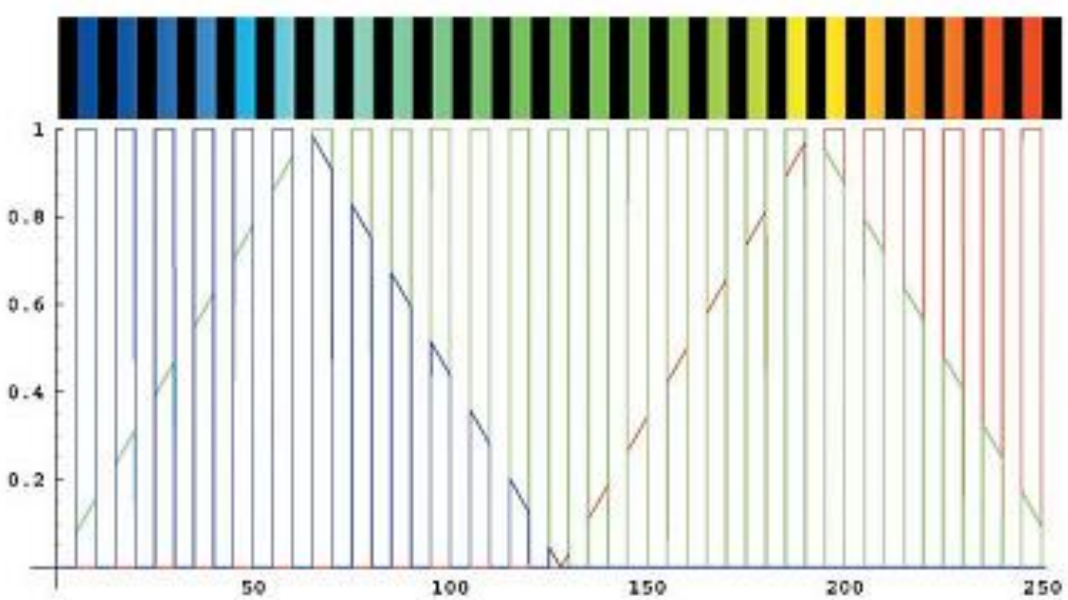
[Munzner / Stone]



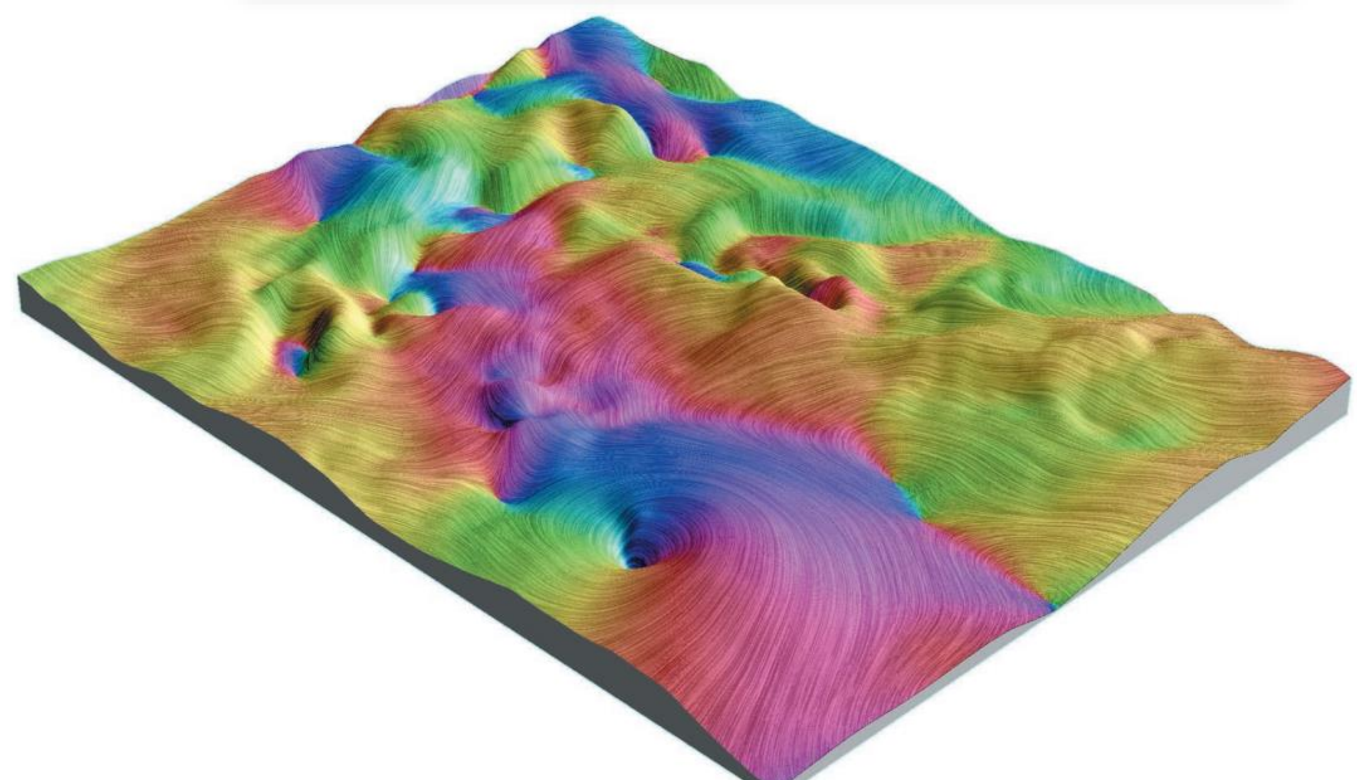
Layering techniques [Wong 02]

- Opacity modulation
- Filigreed
- Colormap enhancement
- 2D heightmap

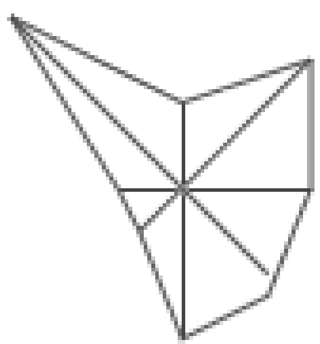




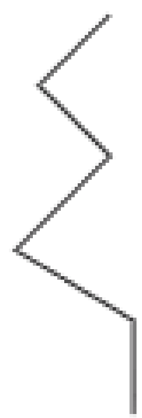
Colormap + square wave modulation



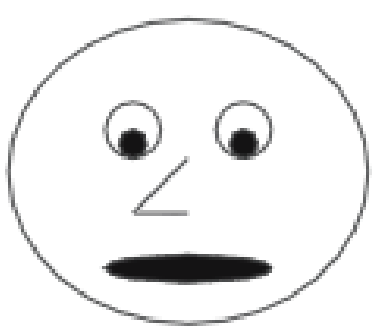
Glyphs: Small independent visual objects that depict attributes of a data record



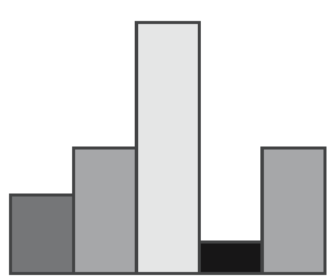
Star glyphs



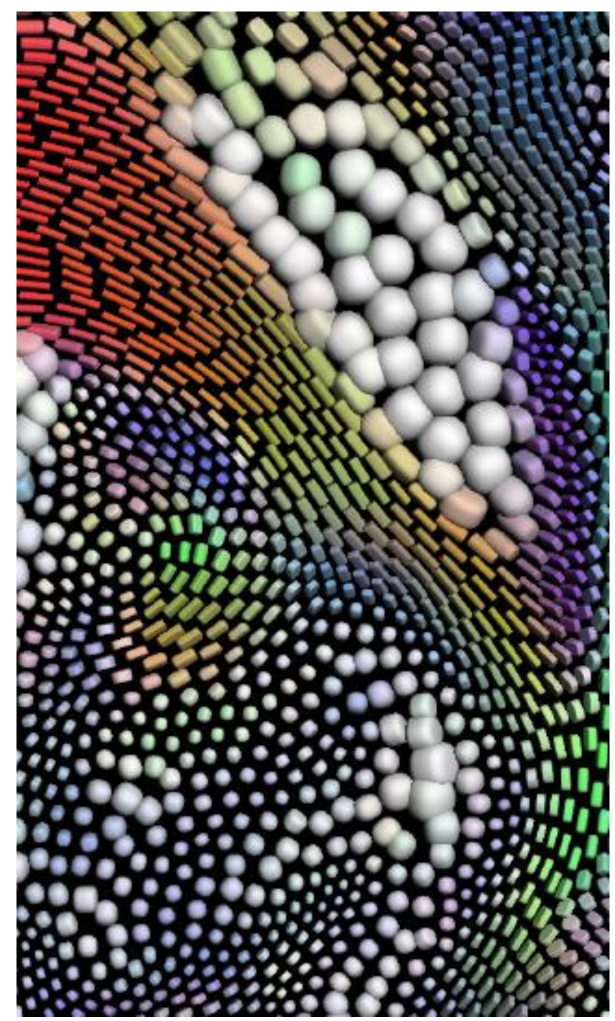
Stick figures



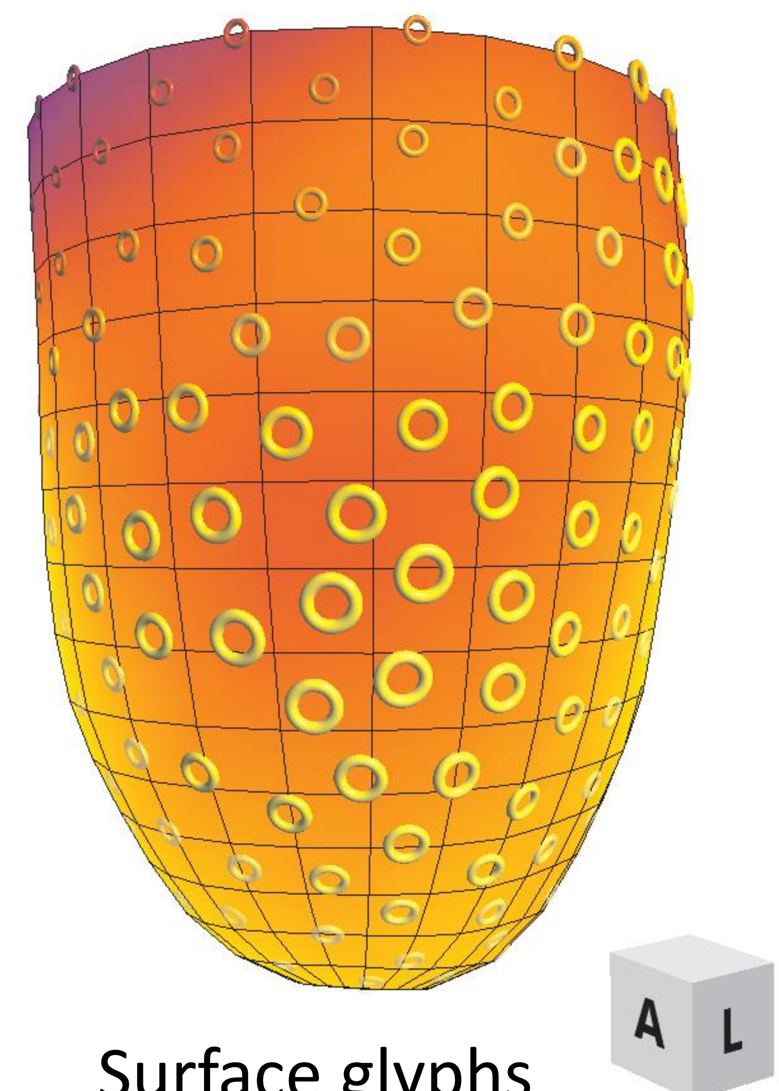
Chernoff faces



Profile glyphs



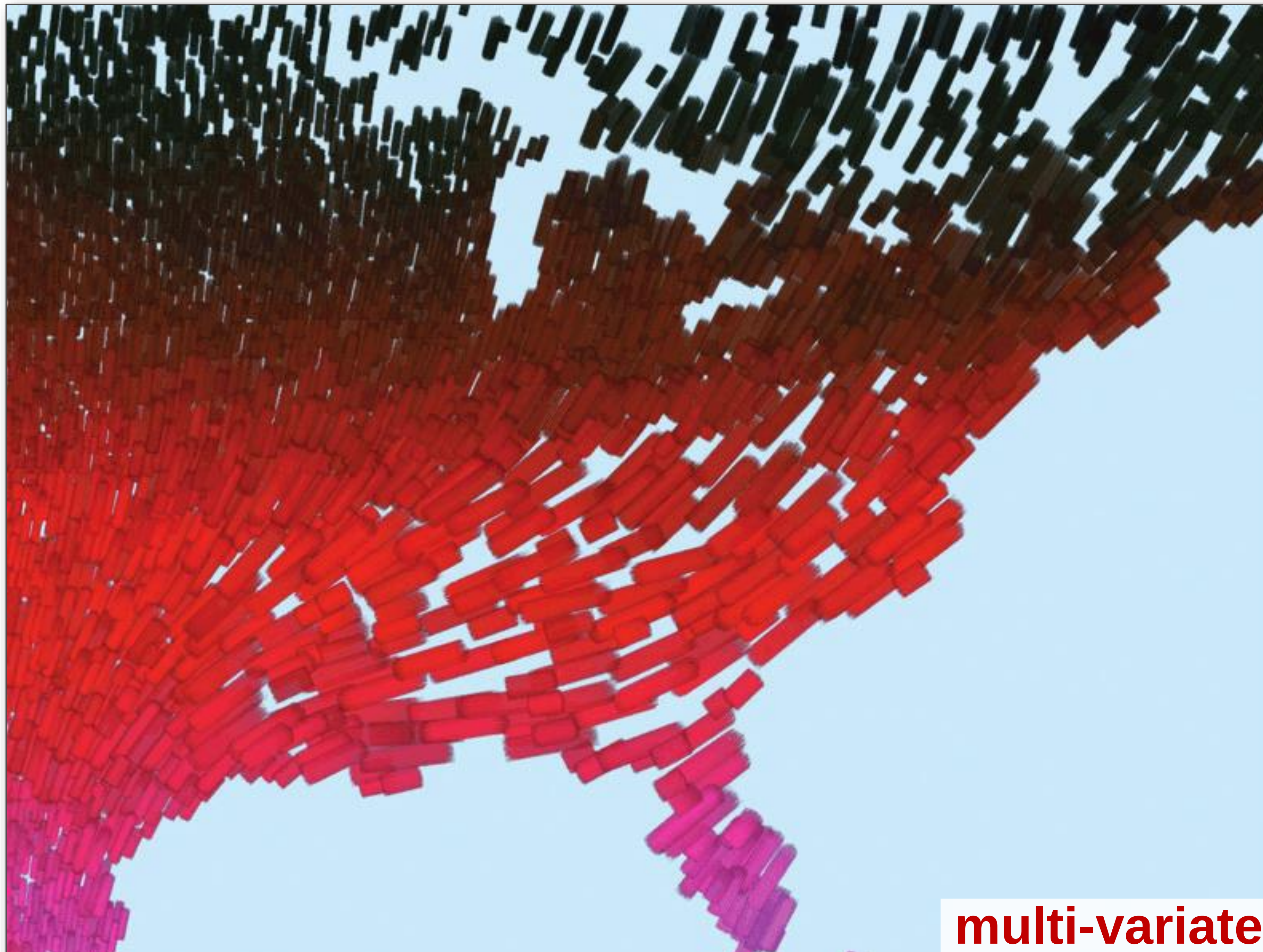
Glyph packing
[Kindlmann&Westin 06]



Surface glyphs
[Meyer-Spradow et al. 08]

Textures and Colors [Healey & Enns 02]

- Temperature
  → color
- Wind speed
  → coverage
- Pressure
  → size
- Precipitation
  → orientation



visual mapping — interactive visual analysis — comput. analysis

relation & comparison

focus+context & overview+detail

data abstraction & aggregation

visual data fusion

navigation

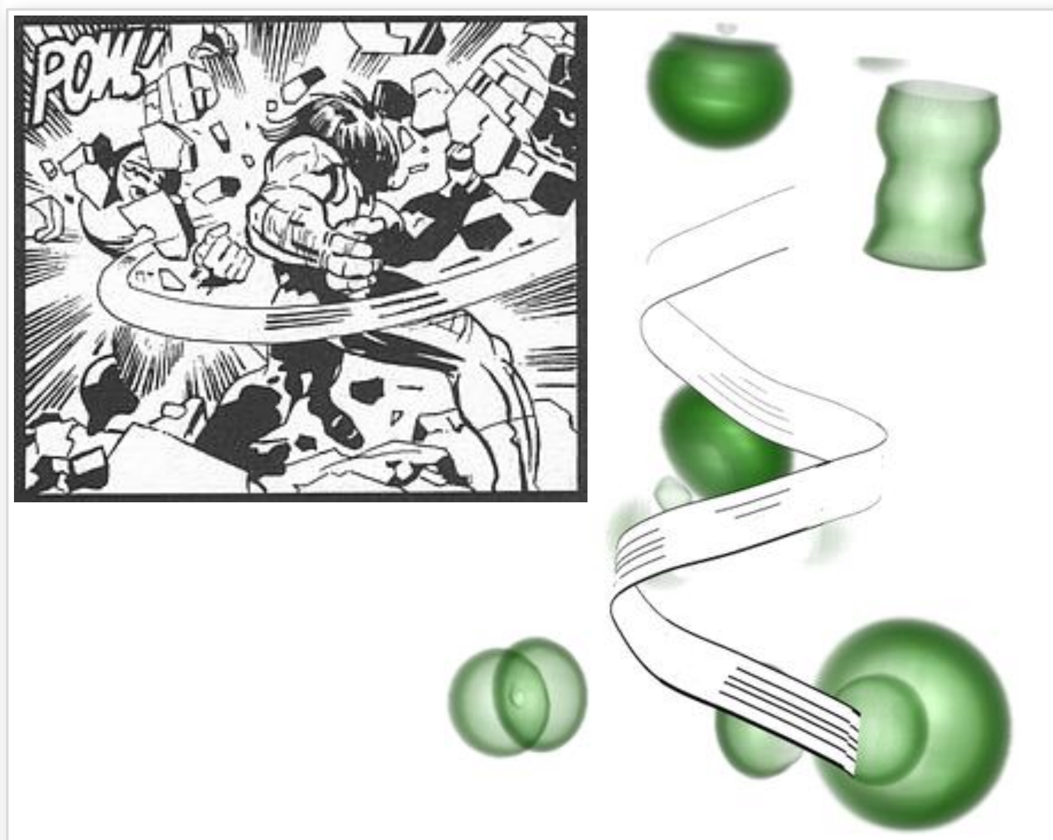
interactive feature spec.

Illustrative visualization techniques

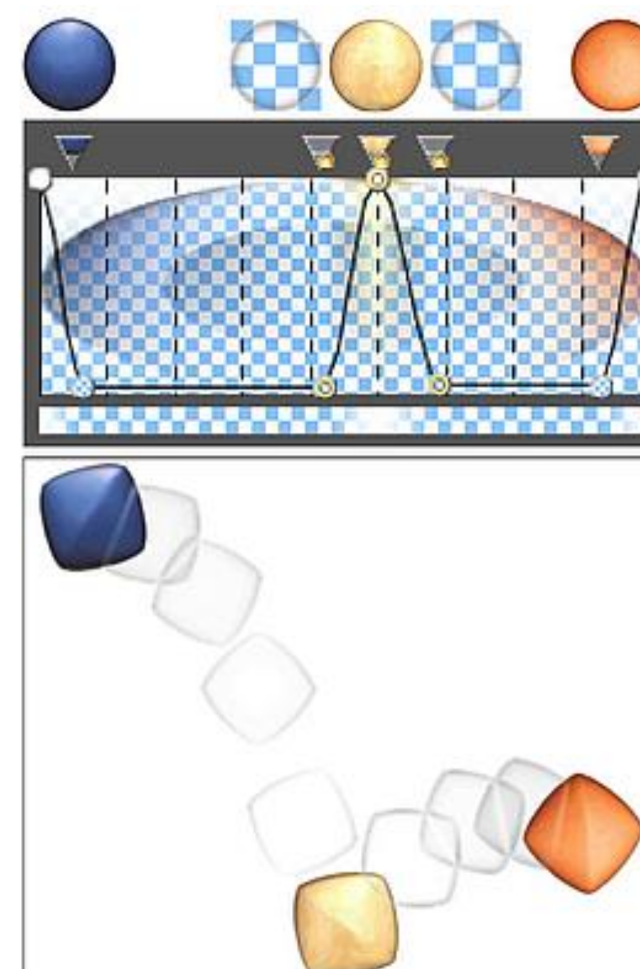
spatio-temporal

Opacity & speedlines [Joshi et al. 09]

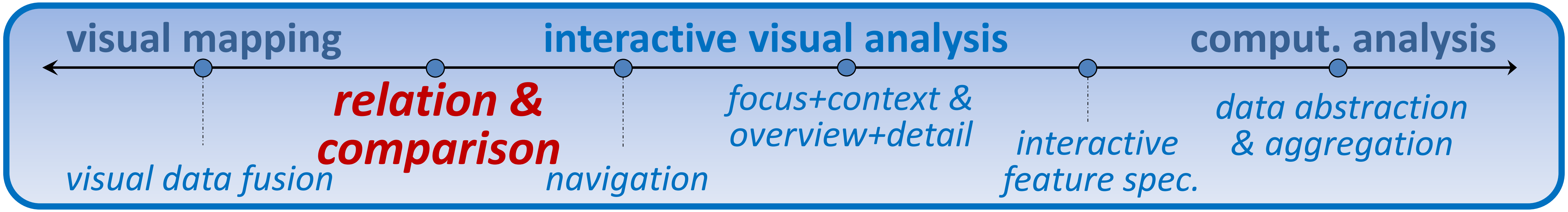
Saturation & silhouettes [Hsu et al. 10]



Flow ribbons [Joshi & Rheingans 05]



Temporal styles [Balabanian et al. 08]

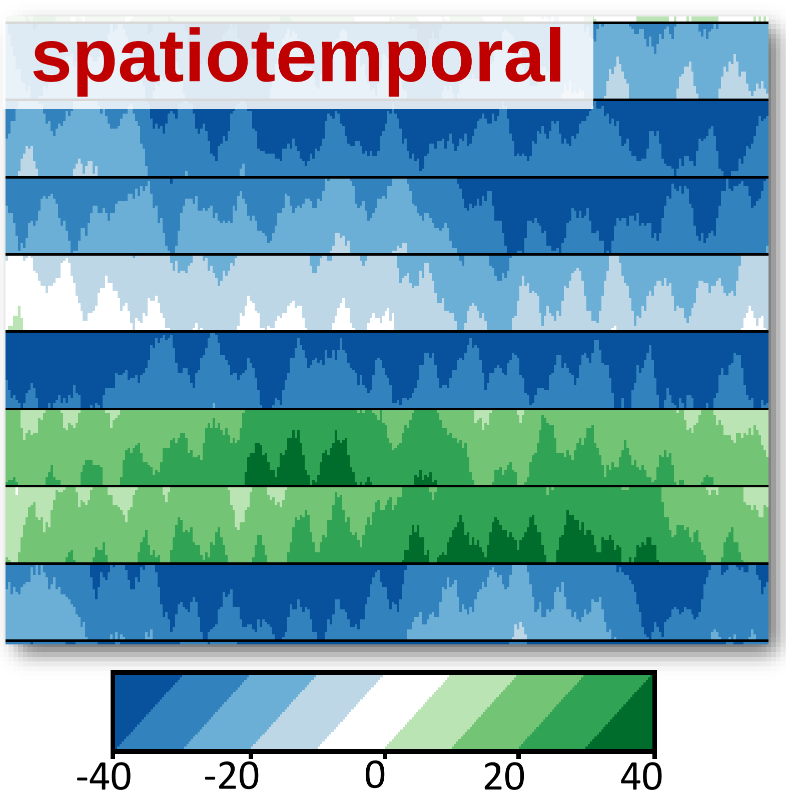


Comparative visualization taxonomy [Gleicher et al. 2011]

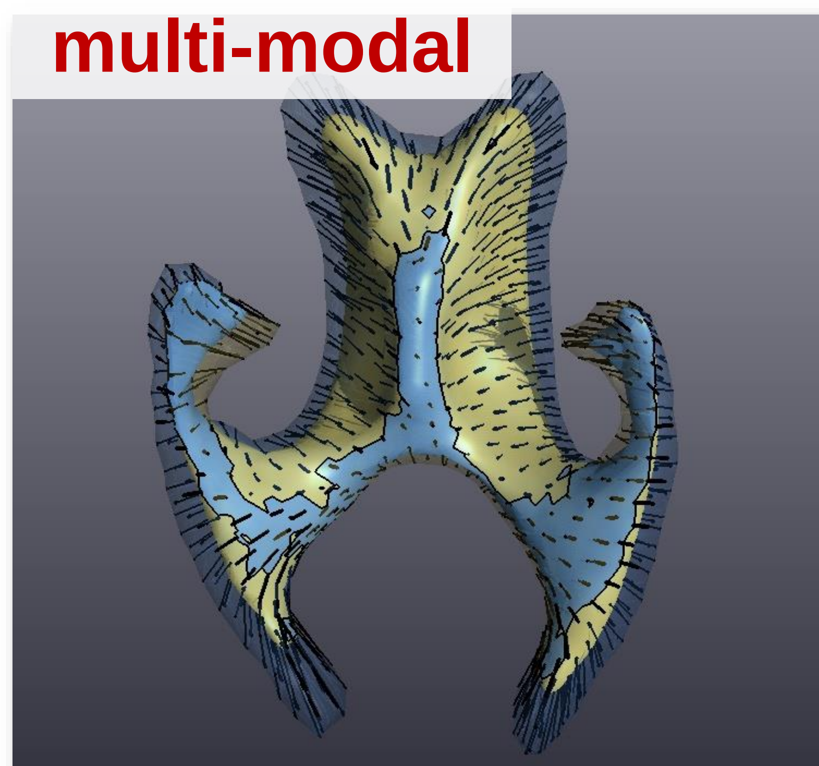
– Side-by-side comparison (juxtaposition)

– Overlay in same coordinate system (superposition)

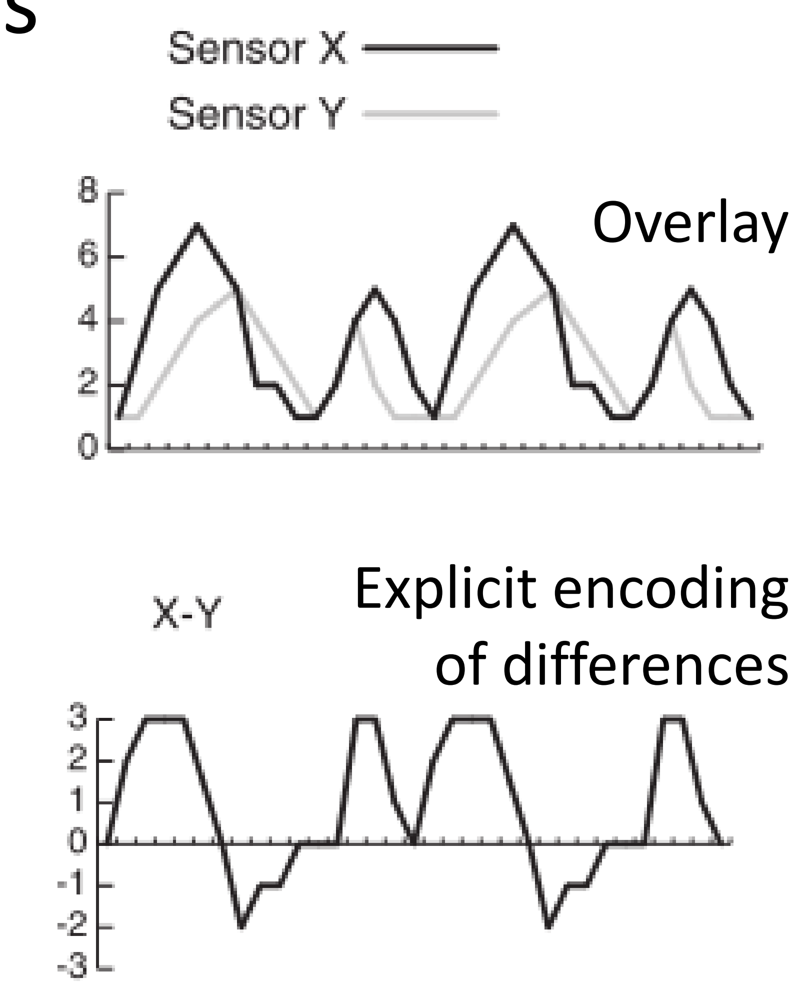
– Explicit encoding of differences / correlations



Horizon graphs [Saito 05, Heer 09]



Nested surfaces [Buskin et al. 11]

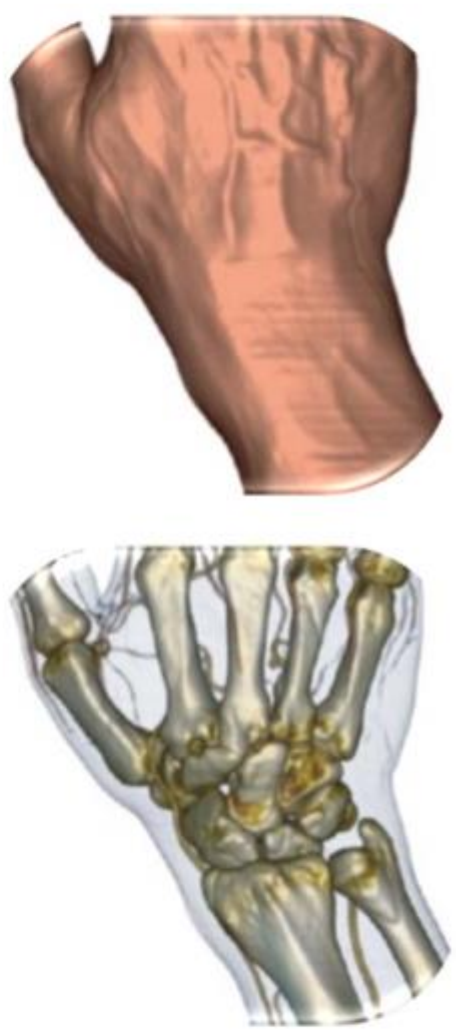


visual mapping — interactive visual analysis — comput. analysis

visual data fusion · **relation & comparison** · navigation · focus+context & overview+detail · interactive feature spec. · data abstraction & aggregation

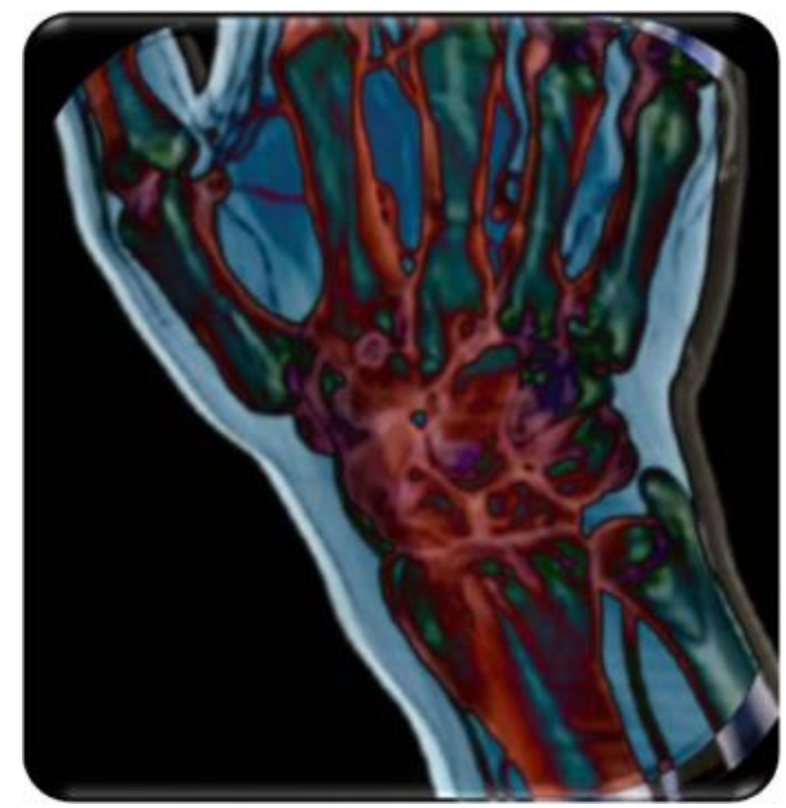
Comparative visualization taxonomy [Gleicher et al. 2011]

Juxtaposition

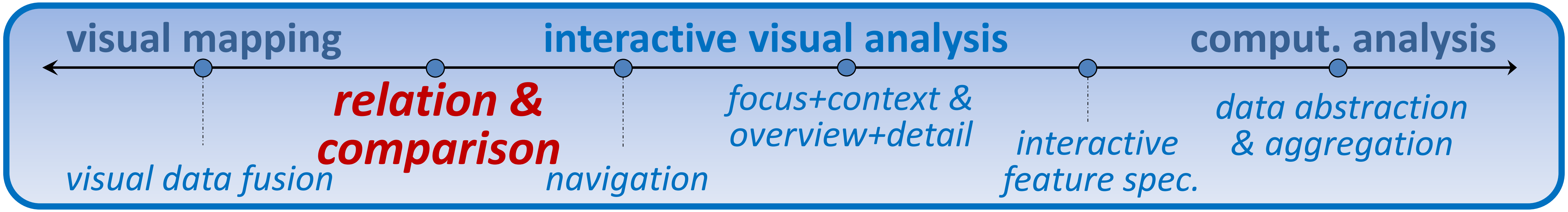


Overlay

Explicit encoding

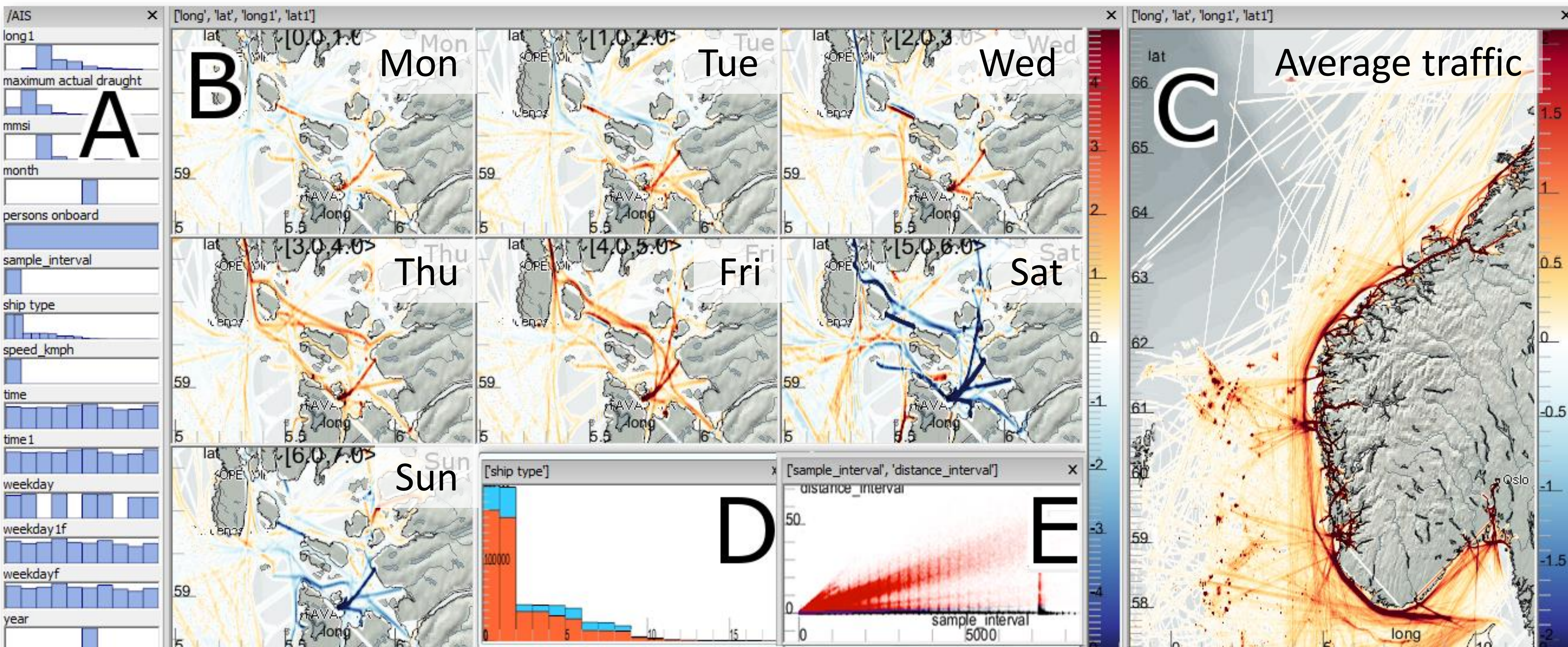


[Gröller & Schmidt]

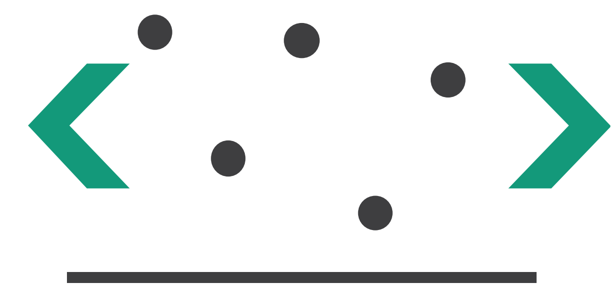


- Difference Views [Daae Lampe et al. 10, Kehrer et al. 13]
  - Side-by-side comparison + explicit encoding
  - Show difference from average per weekday

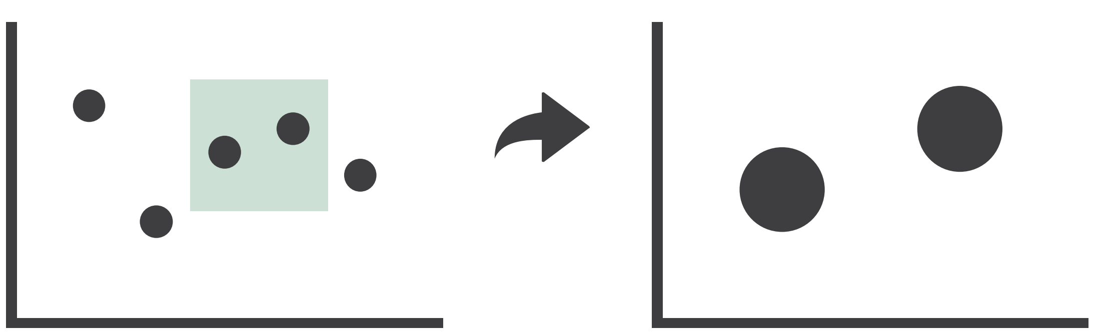
spatiotemporal



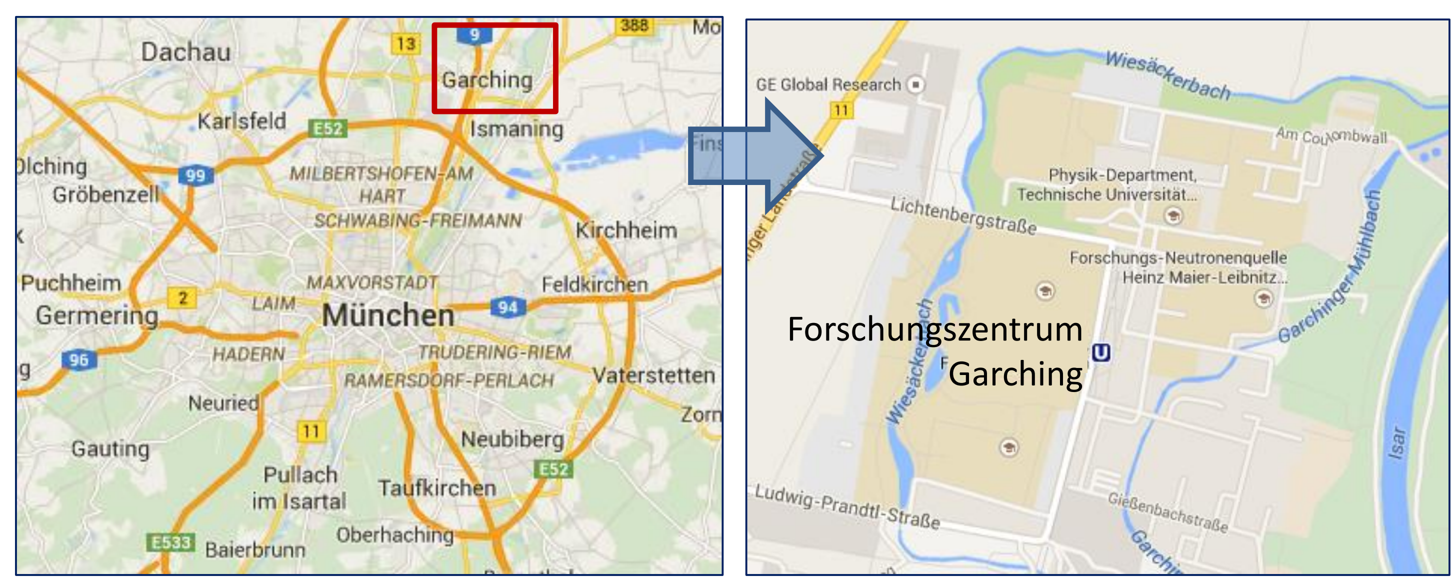
- Navigation: Change item visibility
  - Change which items are visible
  - Camera metaphor
  - Zoom, pan, rotate (3D)



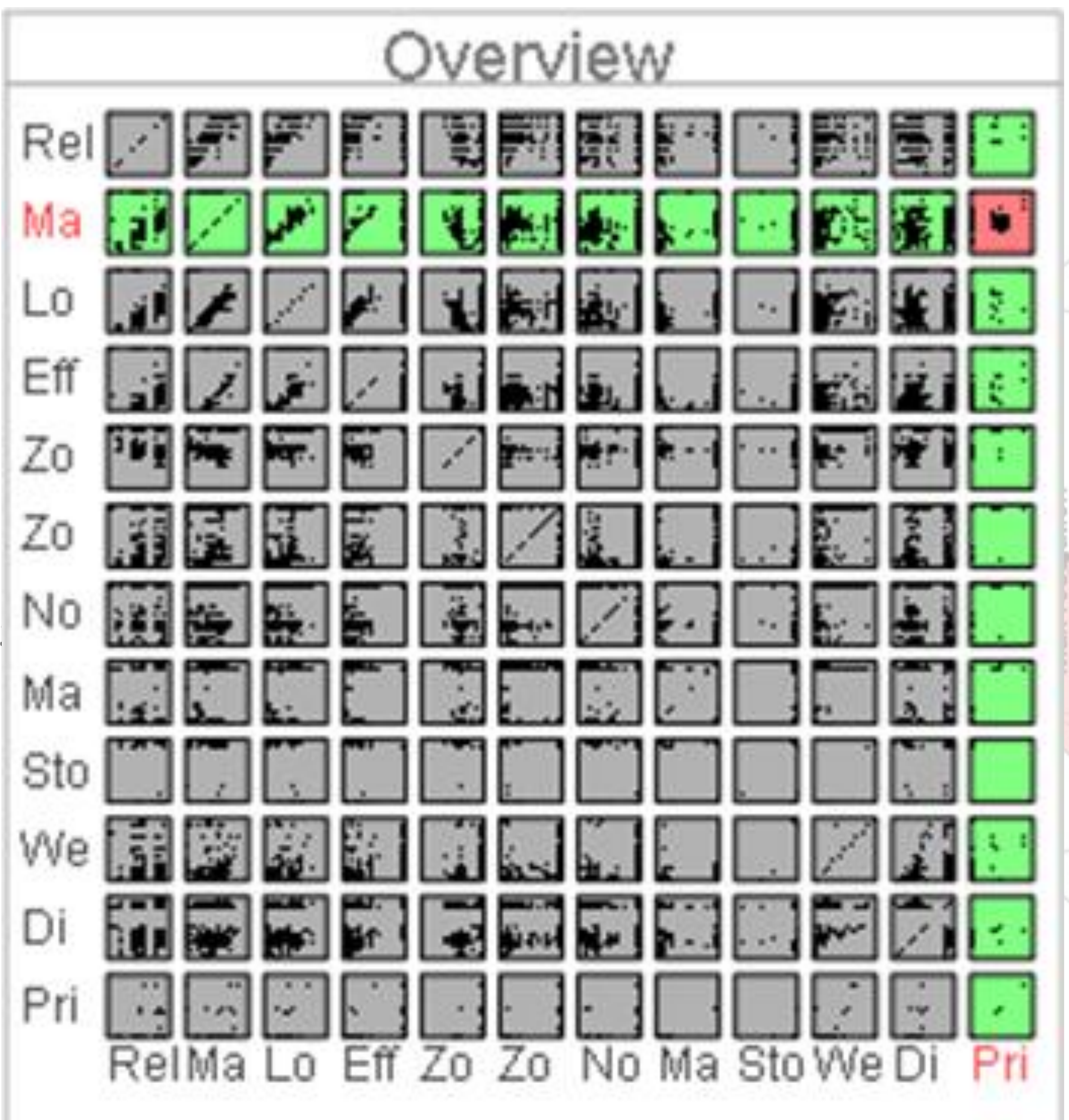
Pan/translate

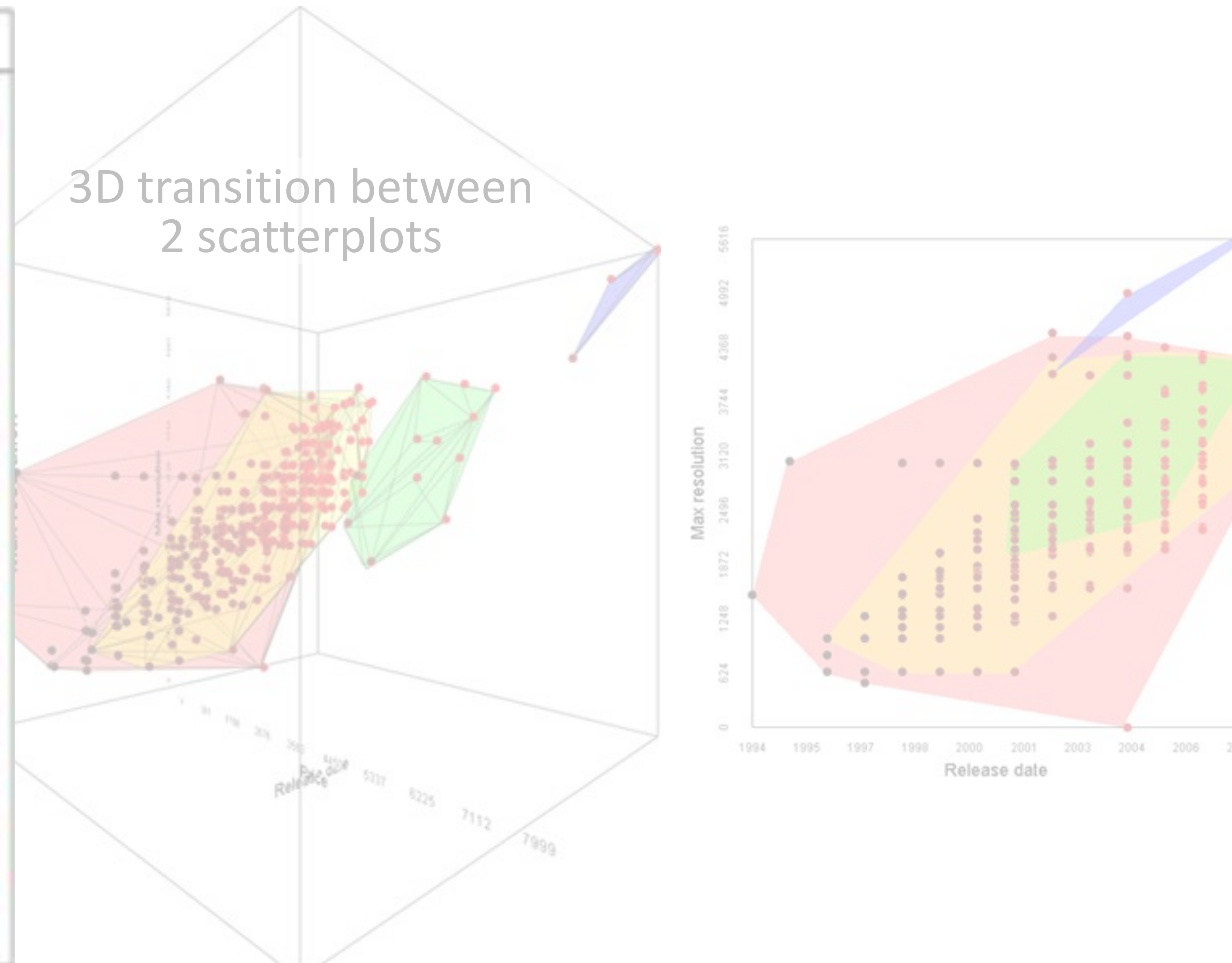


Zoom



- Scatterplot Matrix Navigation [Elmqvist et al. 2008]
  – How to navigate a large matrix with scatterplots?

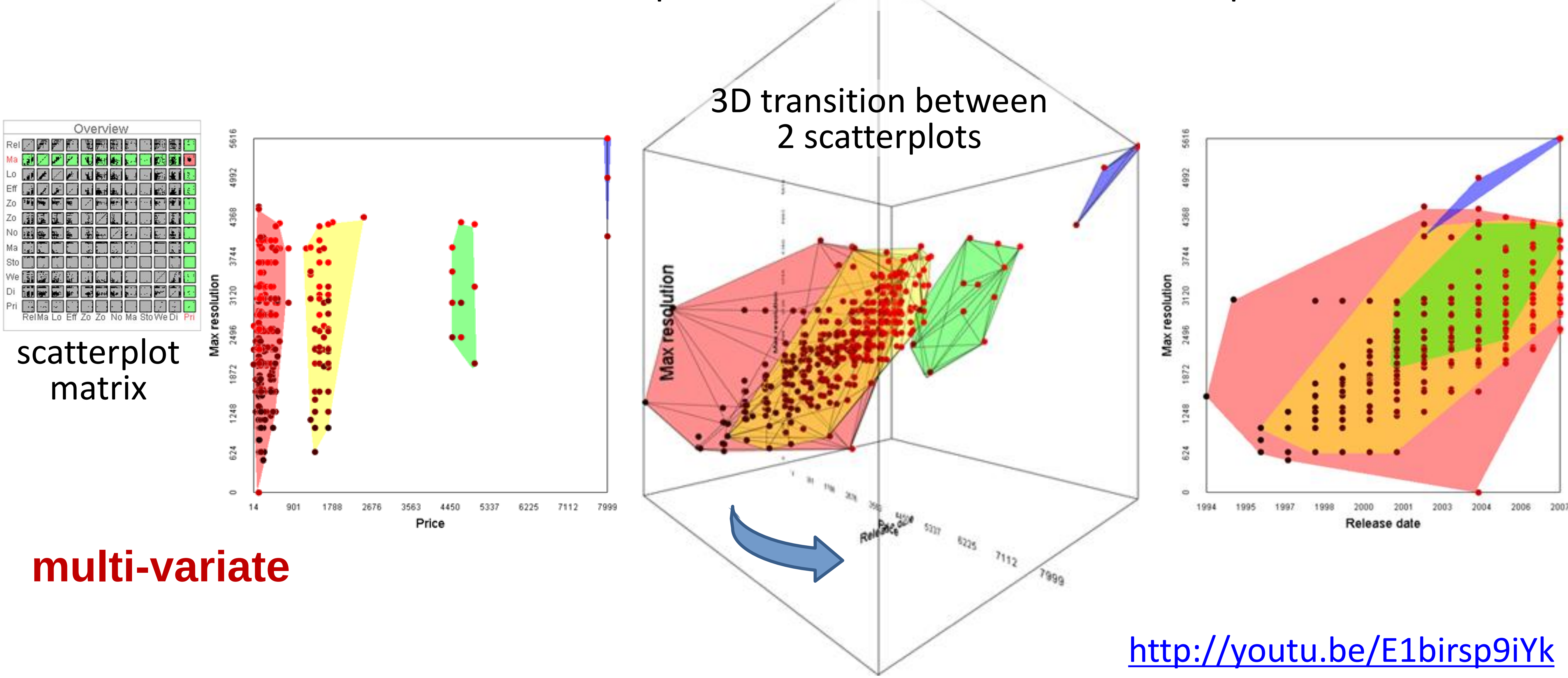




visual mapping — interactive visual analysis — comput. analysis

visual data fusion · relation & comparison · **navigation** · focus+context & overview+detail · interactive feature spec. · data abstraction & aggregation

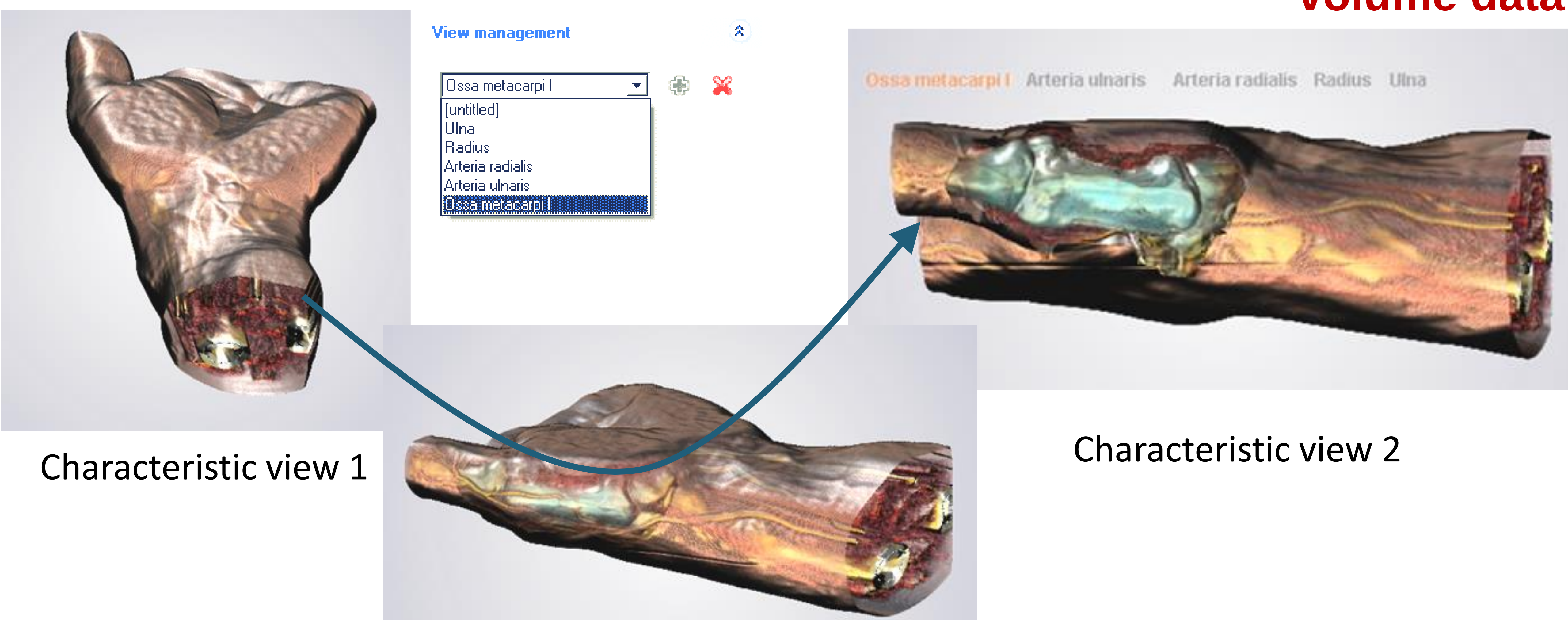
- Scatterplot Matrix Navigation [Elmqvist et al. 2008]
  - How to navigate a large matrix with scatterplots?
  - Transitions between scatterplots as animated rotations in 3D space



http://youtu.be/E1birsp9iYk

- Automated viewpoint selection
  - Guided navigation between characteristic views
  - Based on information-theoretic measures

segmented volume data



Characteristic view 1

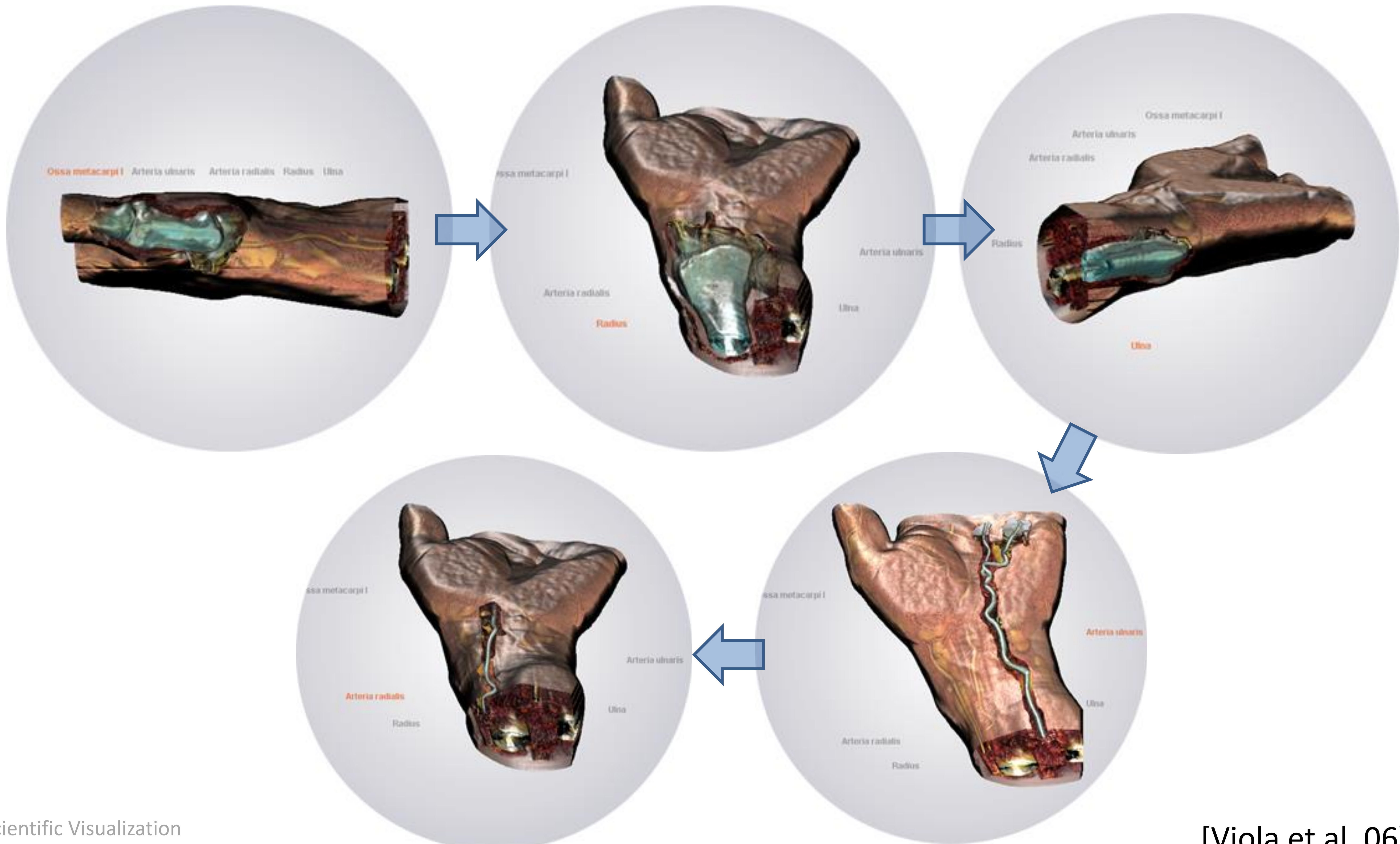
Animated transition

Characteristic view 2

[Viola et al. 06]

visual mapping | **interactive visual analysis** | **comput. analysis**

visual data fusion · *relation & comparison* · ***navigation*** · *focus+context & overview+detail* · *interactive feature spec.* · *data abstraction & aggregation*

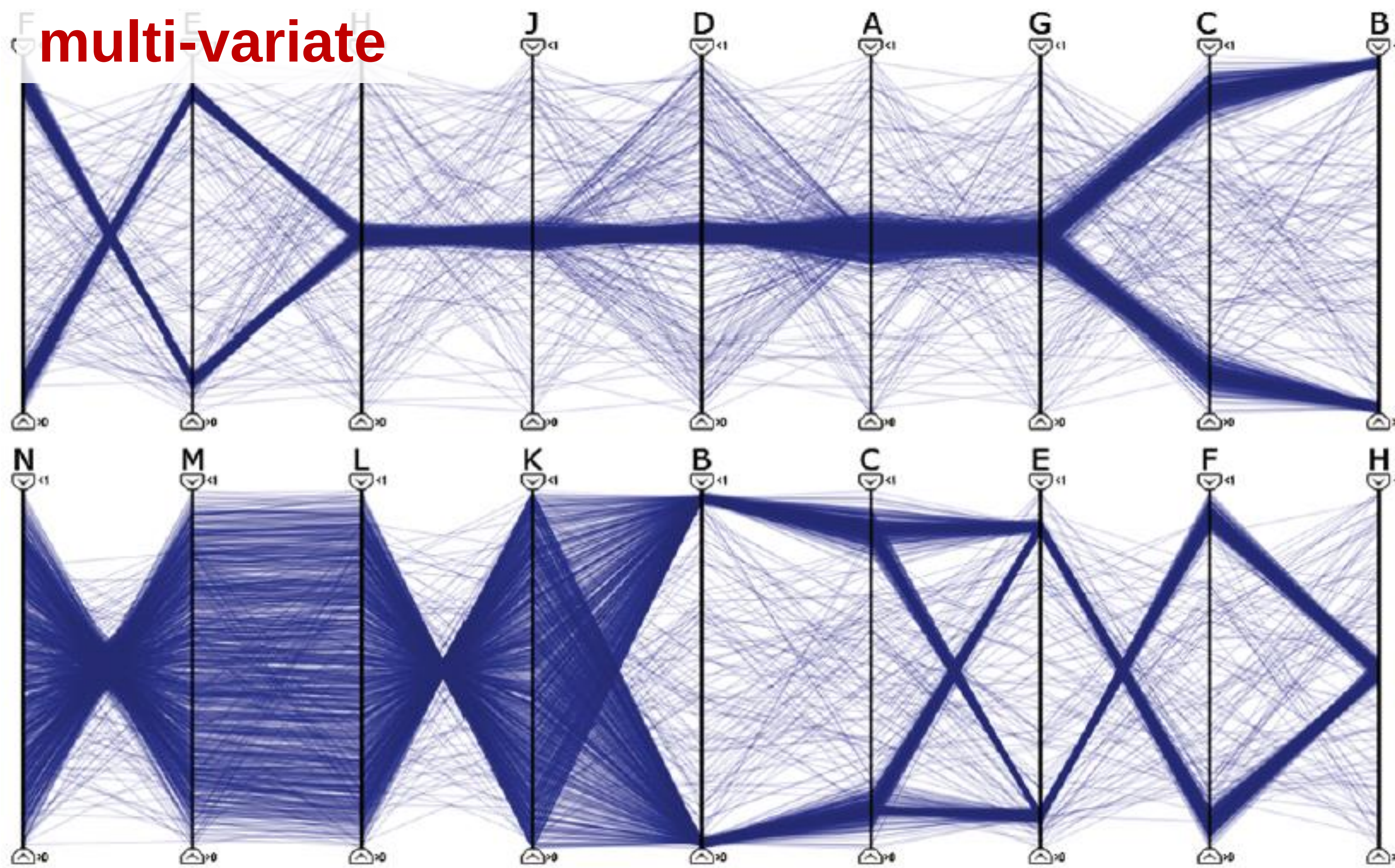


[Viola et al. 06]

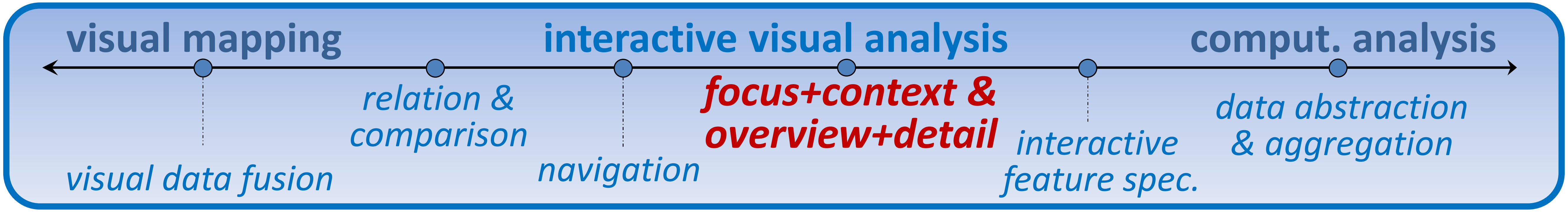
- Ranking/quality metrics [Bertini et al. 2011]
  - Automatically order views/axes by quality metrics
  - Enhance clustering, correlations, outliers, image quality, etc.

Enhance clustering

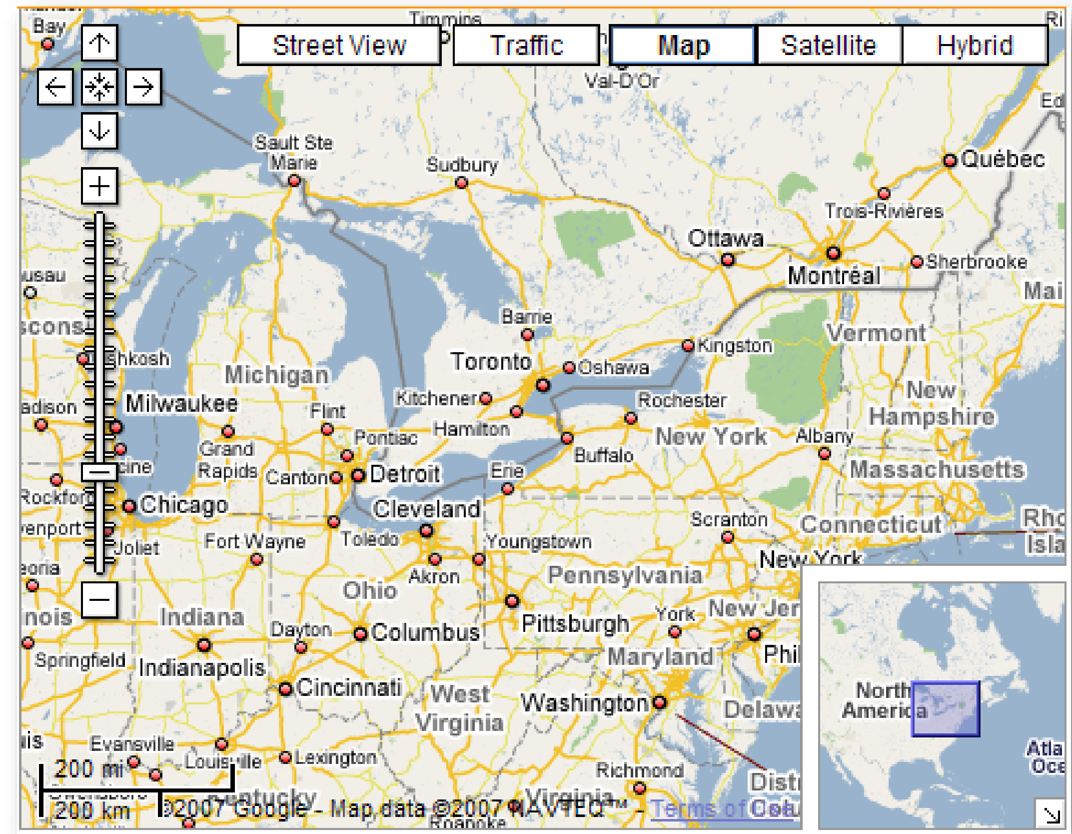
Enhance correlation



[Johansson & Johansson 09]



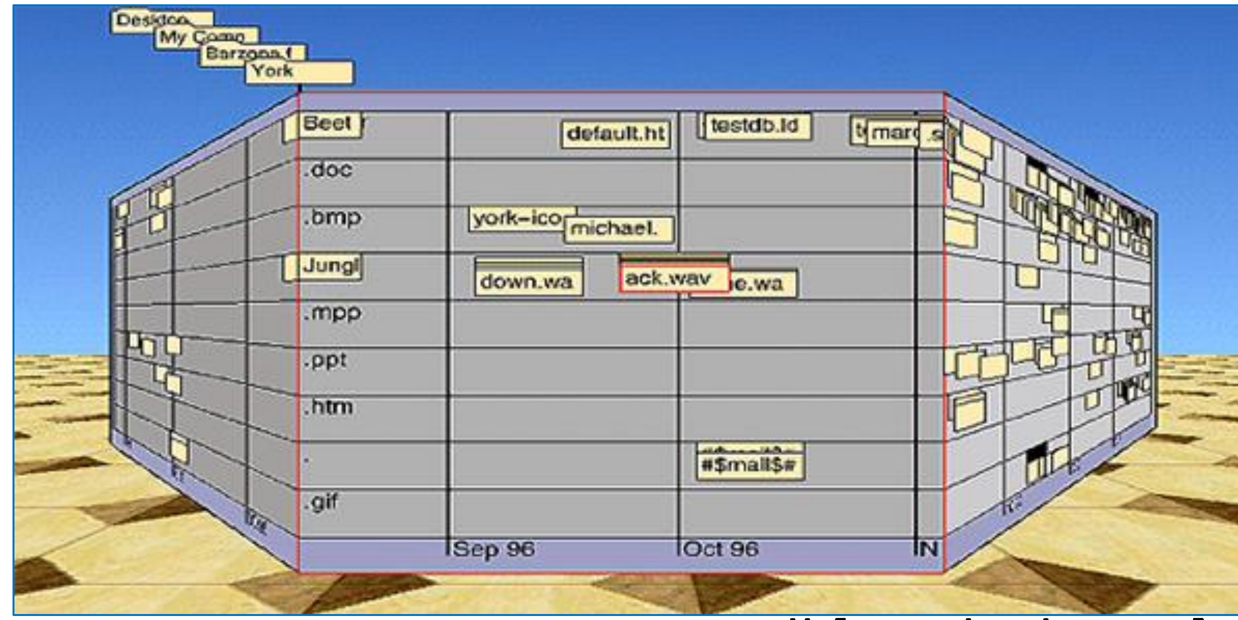
- **Problem**
  - Impossible to show all details of large data set in single image
- **Overview+detail visualization**
  - Spatially separate overview / detail (e.g., juxtaposed views)
  - User has to switch attention between representations
- **Focus+context visualization**
  - Seamlessly integrates focus / context in single visualization



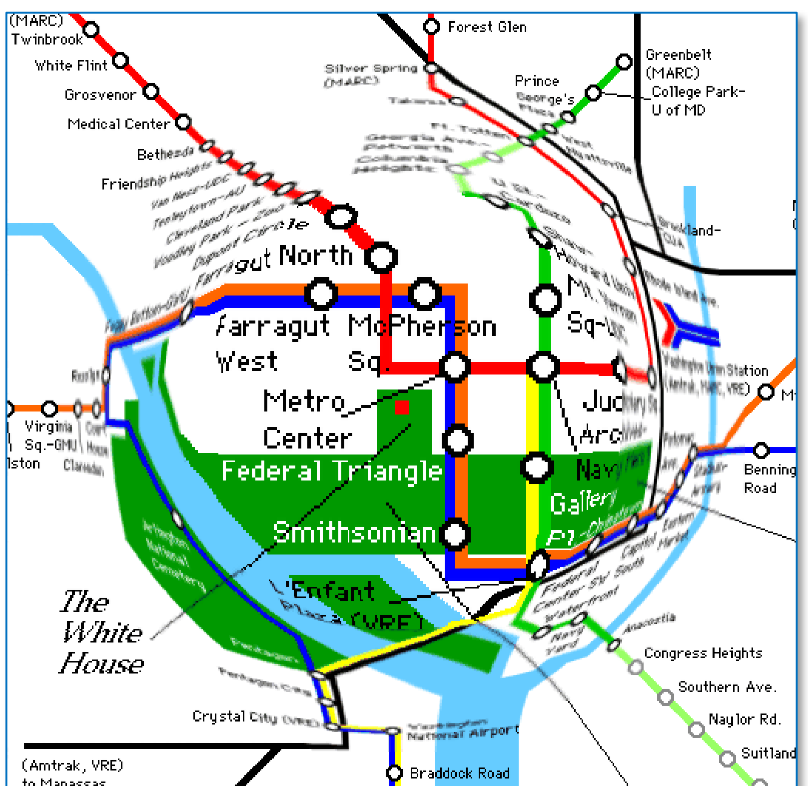
Fisheye views [Furnas 86]

Focus+context (F+C) visualization

- Originally spaced distortion used
- More space for focus
- Keep context, without cropping away data outside of zoom area



Perspective wall [Mackinlay 91]



Fisheye views [Furnas 86]

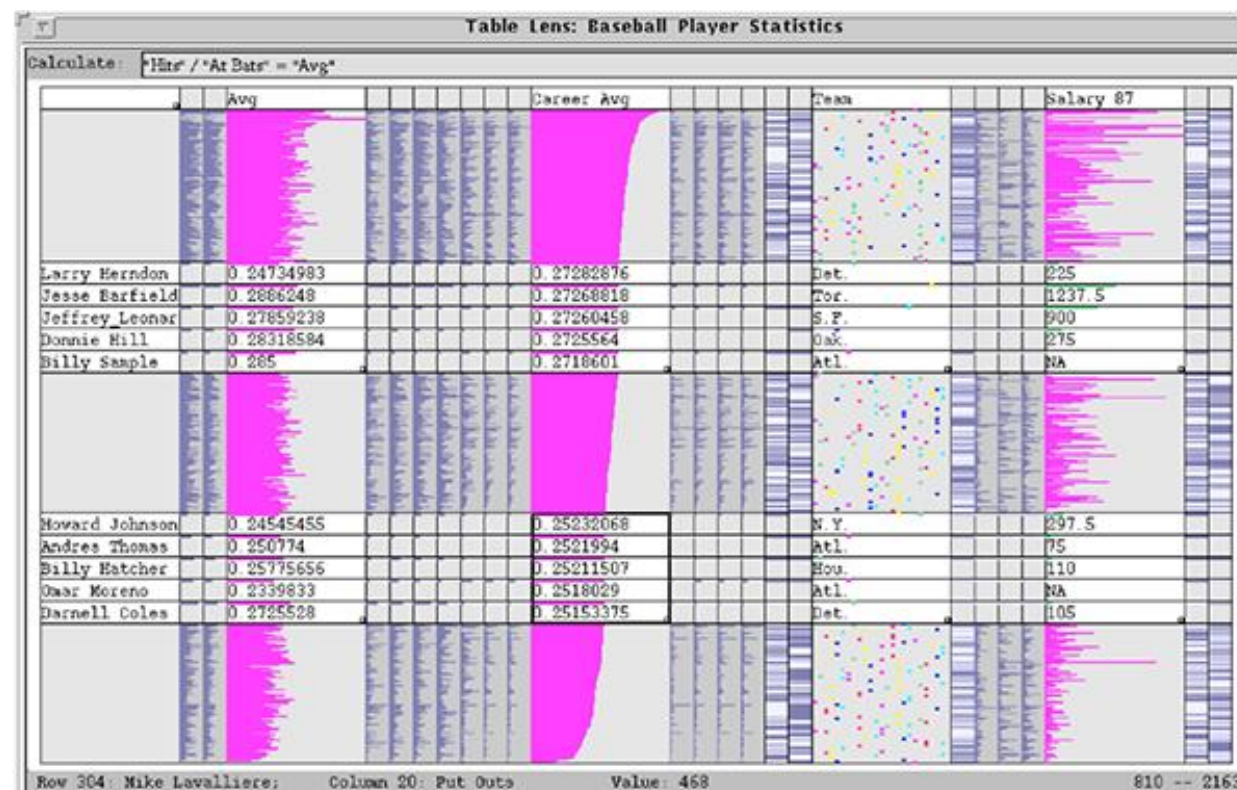


Table lens [Rao/Card 94]

visual mapping — interactive visual analysis — comput. analysis

visual data fusion · relation & comparison · navigation · **focus+context & overview+detail** · interactive feature spec. · data abstraction & aggregation

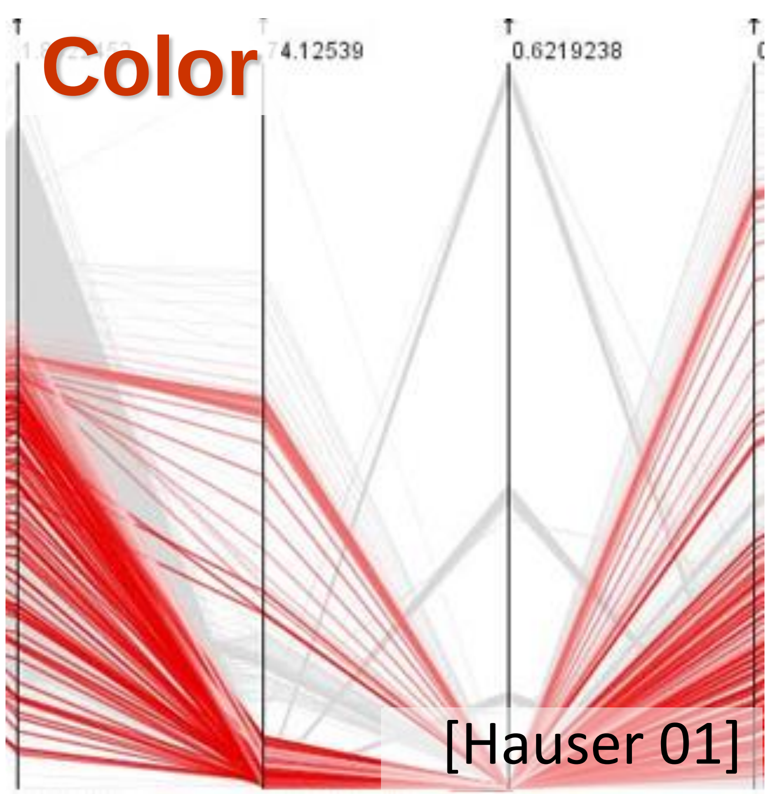
Generalized F+C visualization [Hauser 05]

- Emphasize data in focus
- Keep context for orientation/navigation
- Focus specification, e.g., by pointing, brushing or querying

Opacity/transparency

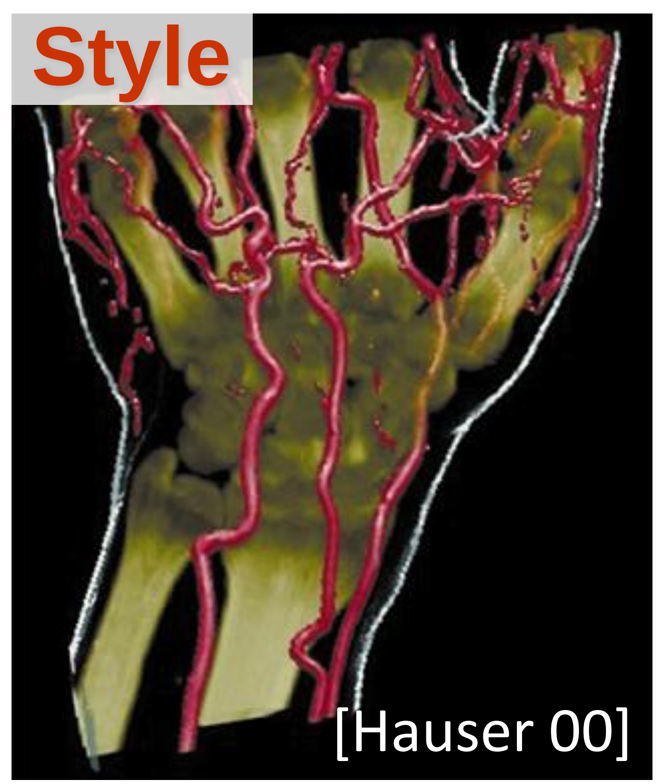
Importance-driven rendering [Viola 04]

Color



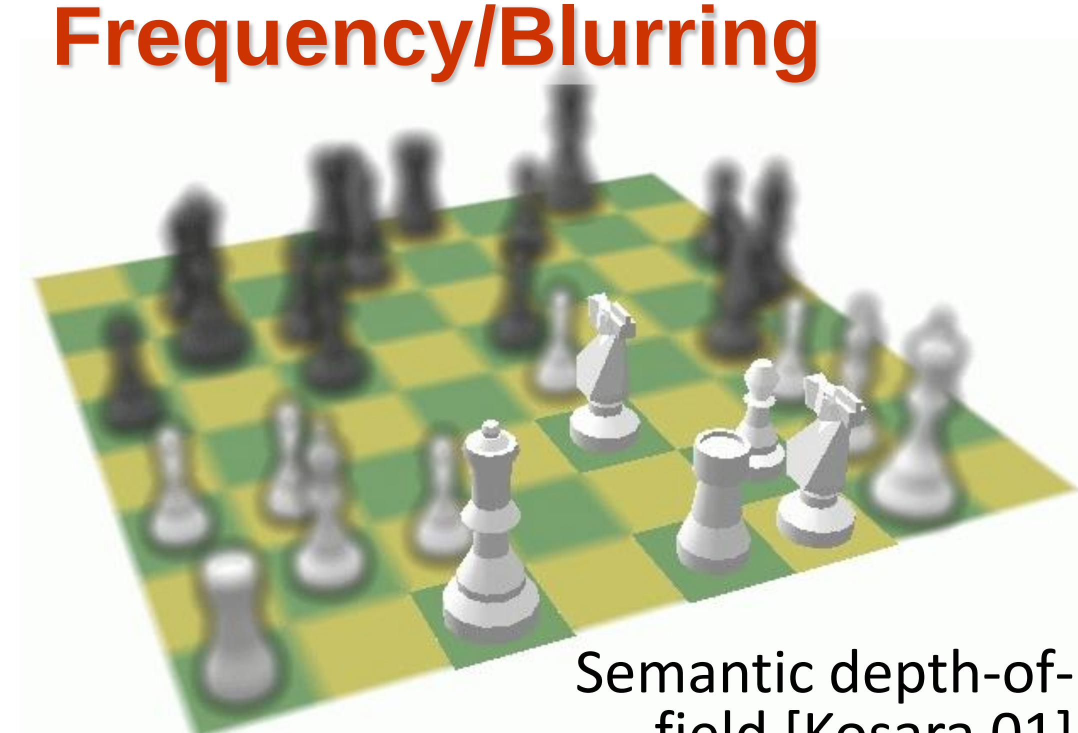
[Hauser 01]

Style



[Hauser 00]

Frequency/Blurring



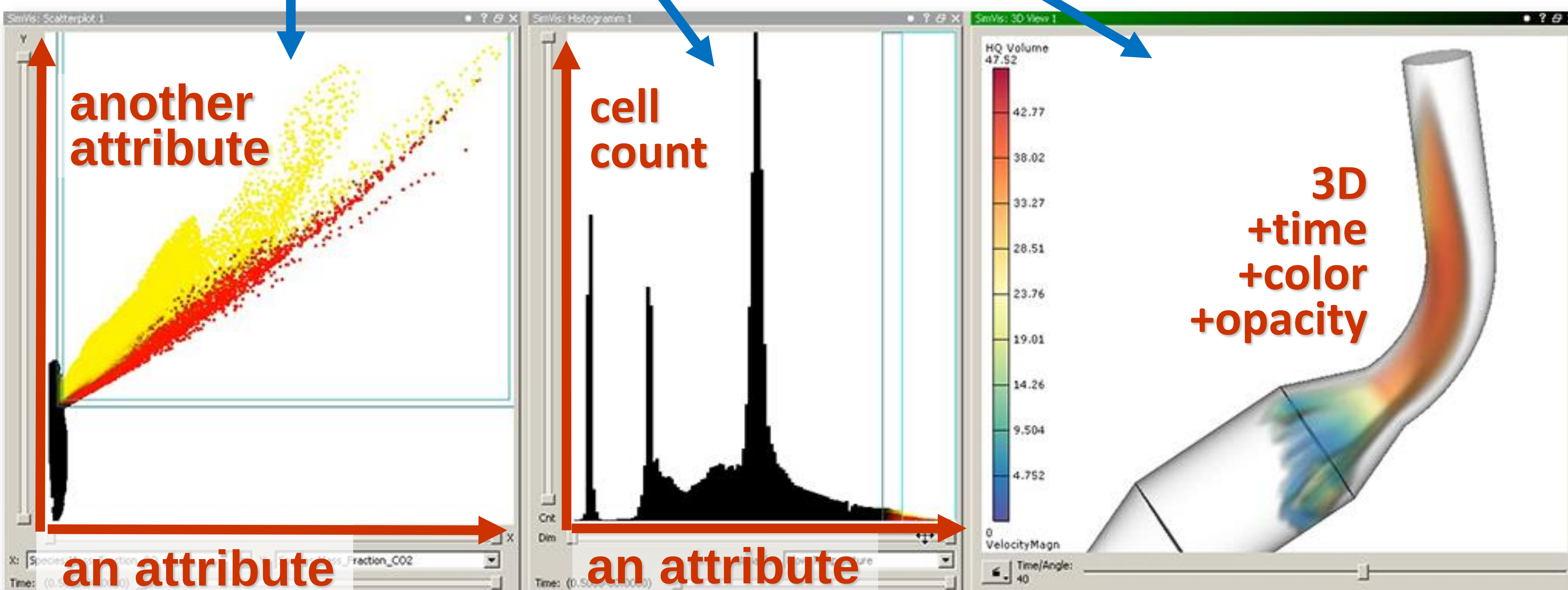
Semantic depth-of-field [Kosara 01]

visual mapping — interactive visual analysis — comput. analysis

visual data fusion · relation & comparison · navigation · focus+context & overview+detail · **interactive feature spec.** · data abstraction & aggregation

Brushing in multiple linked views

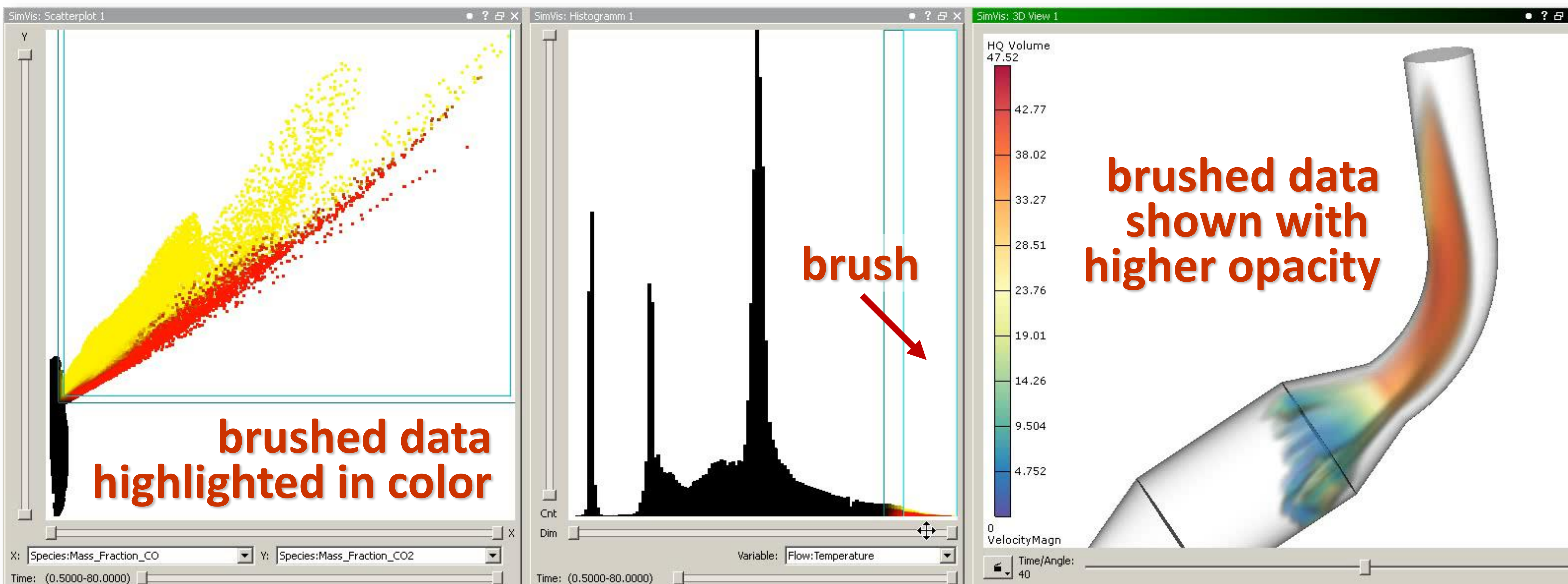
- One dataset, but multiple views
- Scatterplots, histograms, 3D(4D) views, etc.



visual mapping — interactive visual analysis — comput. analysis

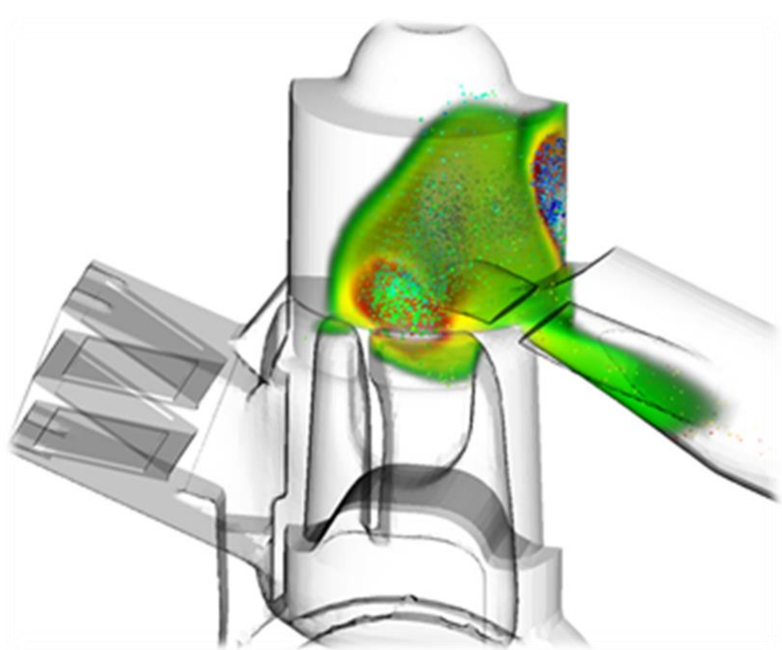
visual data fusion · relation & comparison · navigation · focus+context & overview+detail · **interactive feature spec.** · data abstraction & aggregation

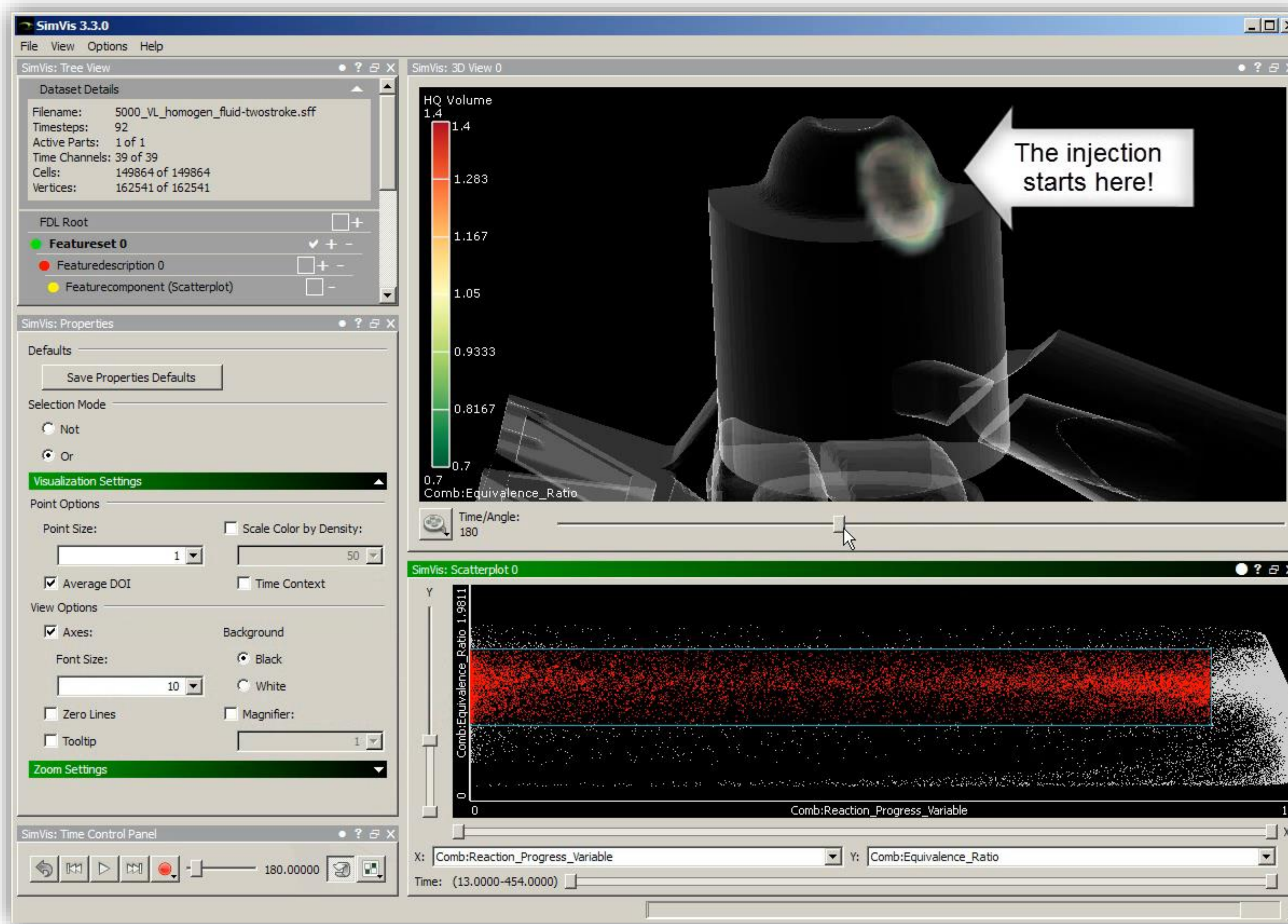
- **Brushing:** mark interesting data subset
- **Linking:** enhance/highlight brushed data in linked views
- Move/alter/extend brush



Brushing in multiple linked views

– Example: Combustion process in a two-stroke engine

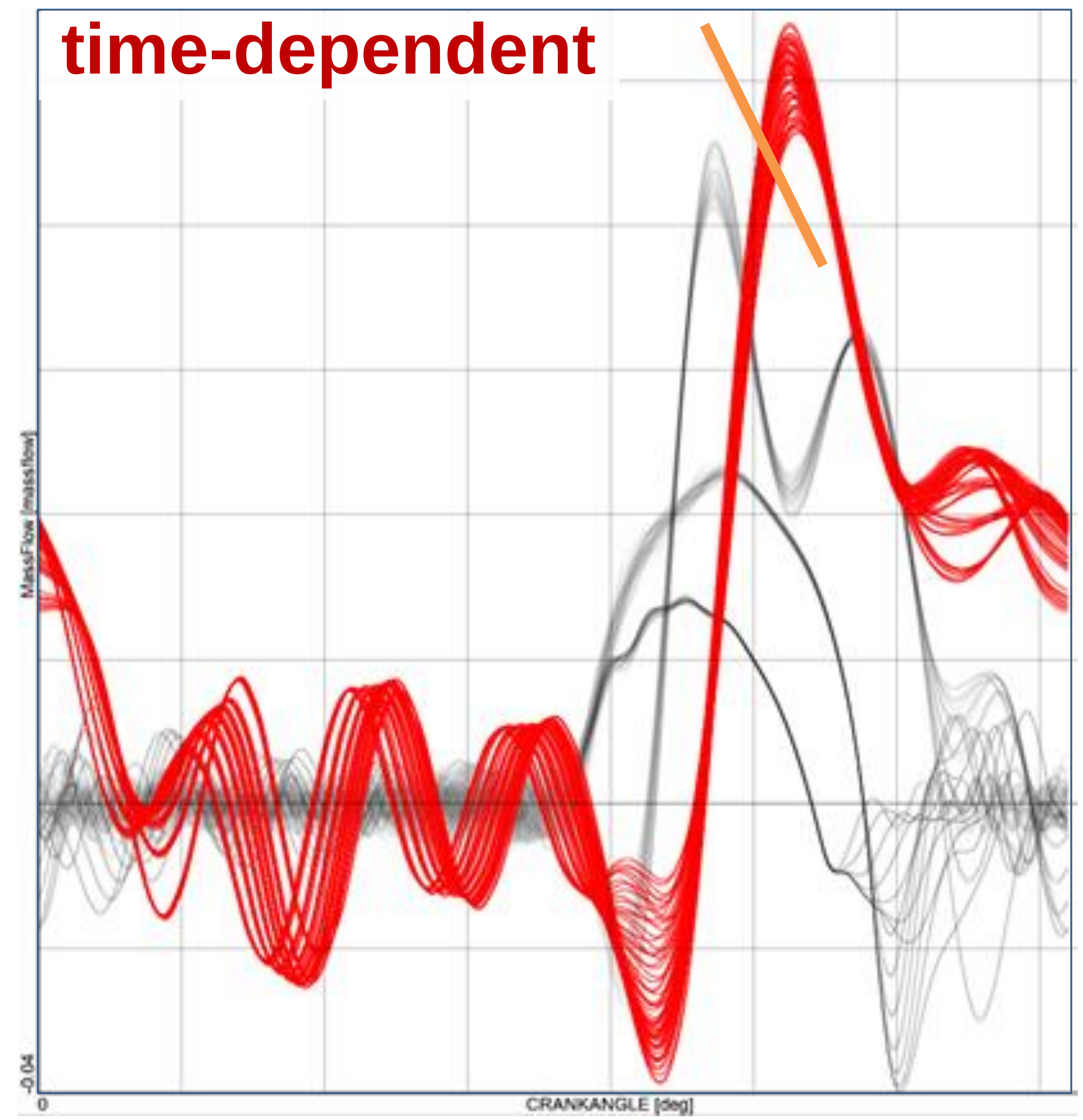


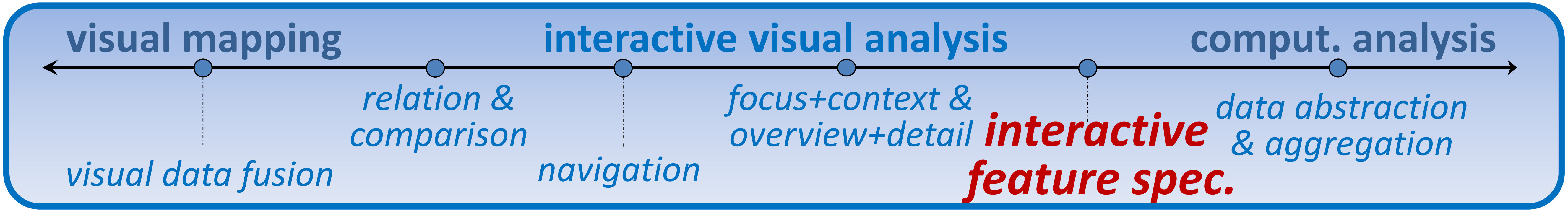


[SimVis GmbH]

Line brush [Konyha et al. 06]

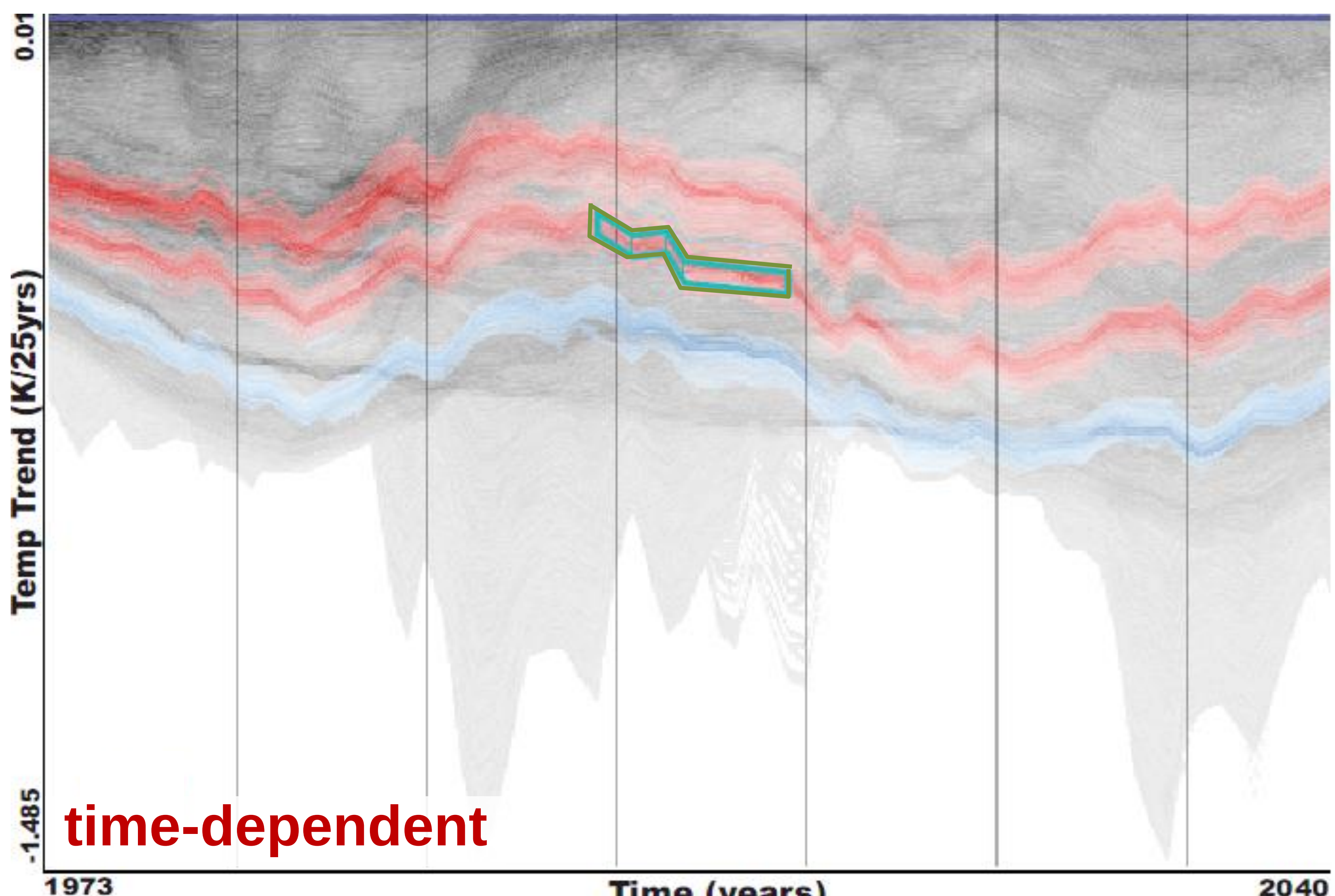
– Select function graphs that intersect with user-specified line





Similarity-based brushing [Muigg et al. 2008]

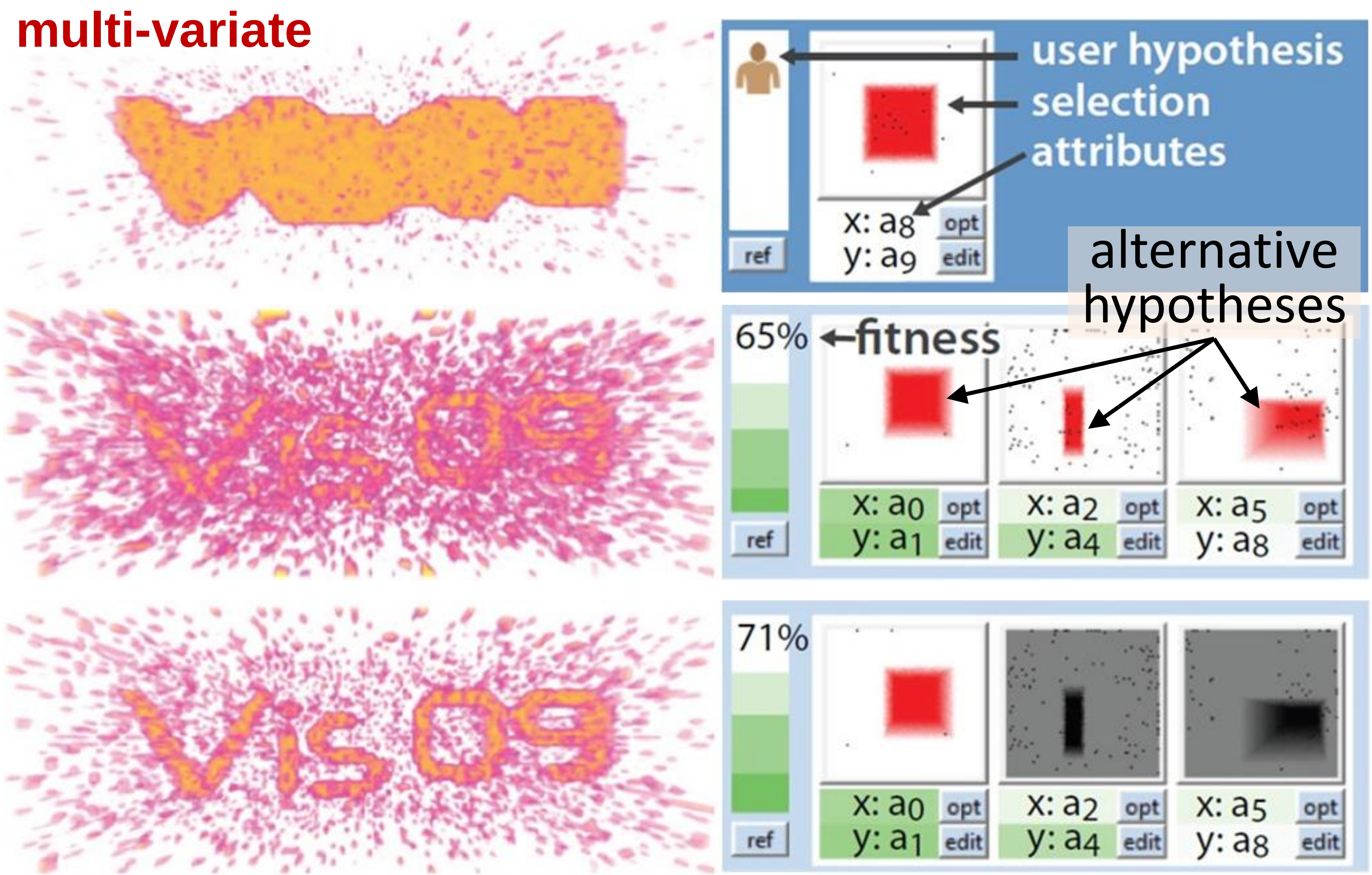
– Select function graphs by similarity to user-sketched pattern
– Similarity evaluated based on gradients (1st derivative)



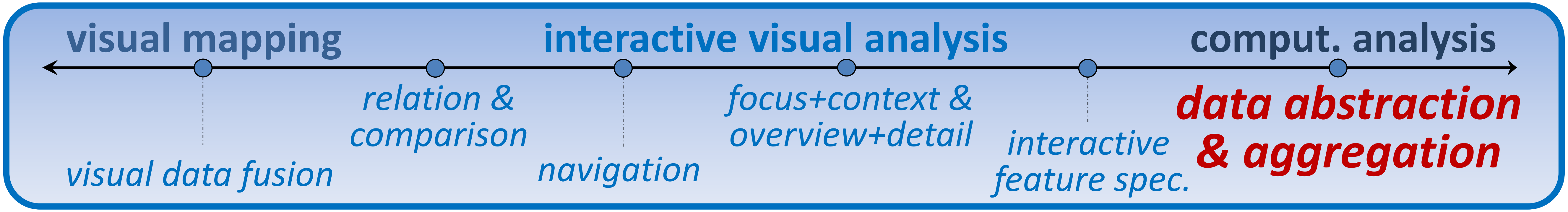
visual mapping — interactive visual analysis — comput. analysis

visual data fusion · relation & comparison · navigation · focus+context & overview+detail · **interactive feature spec.** · data abstraction & aggregation

- Tight integration with supervised machine learning



Visual human+machine learning [Fuchs et al. 09]



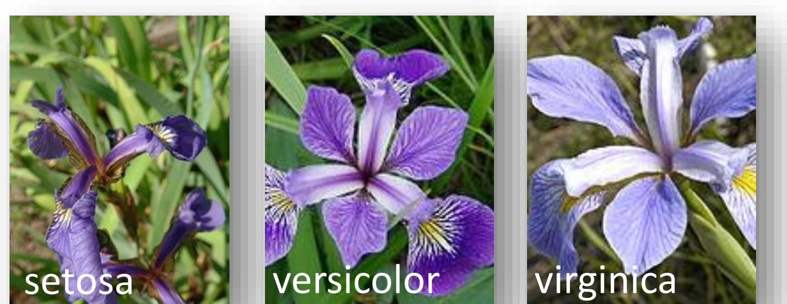
Machine Learning Approaches

Supervised Learning

Learning with a **labeled training set**

Unsupervised Learning

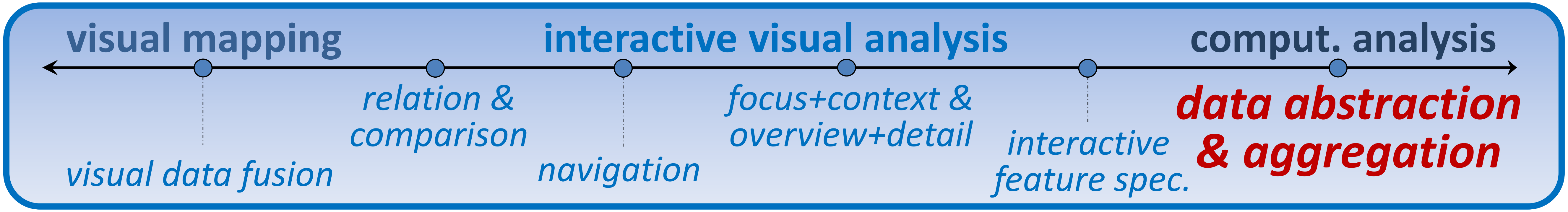
Discovering patterns in unlabeled data



	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	species
0	5.1	3.5	1.4	0.2	0
1	4.9	3.0	1.4	0.2	0
2	4.7	3.2	1.3	0.2	0
3	4.6	3.1	1.5	0.2	0
4	5.0	3.6	1.4	0.2	0

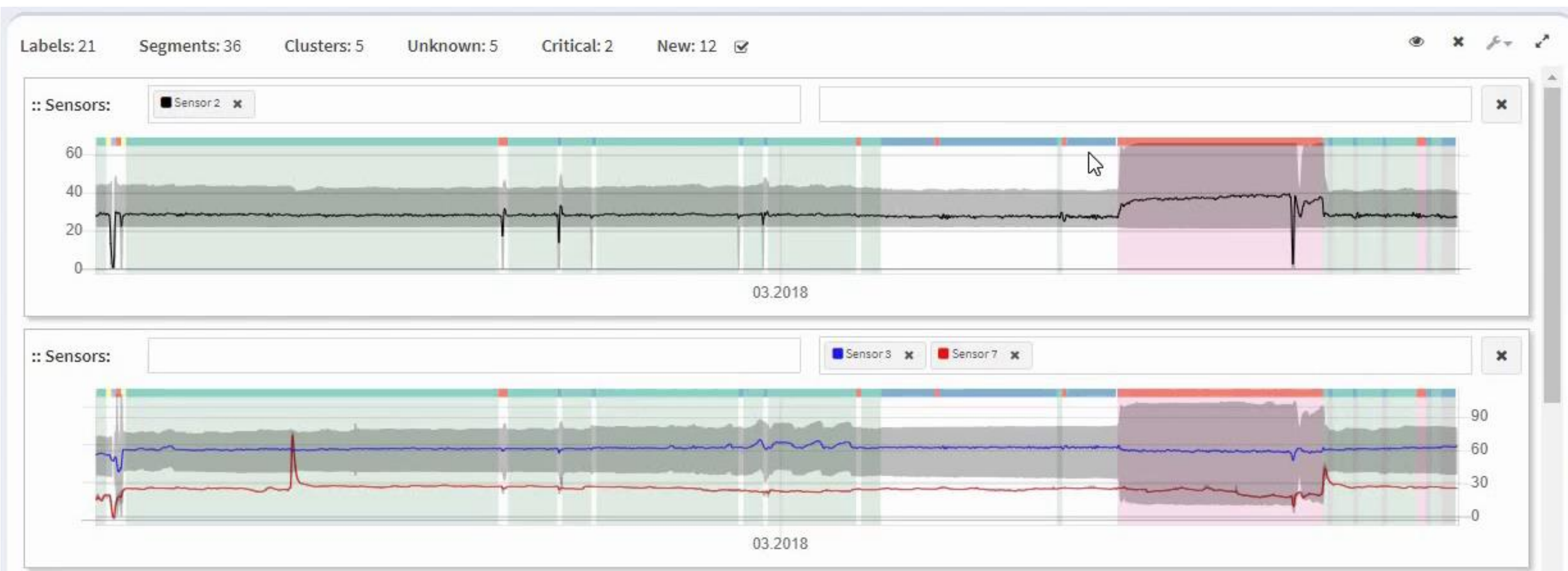
Reinforcement Learning

Learning based on **feedback** or **reward**

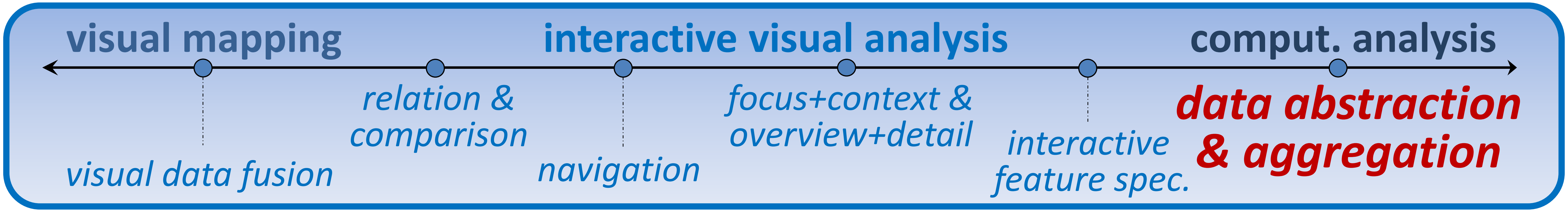


Example: Semi-Automatic Labeling Tool (SALT)

- Labeling of large time series data by domain experts
- Integrates supervised & unsupervised segmentation methods
- User can iteratively improve labeling

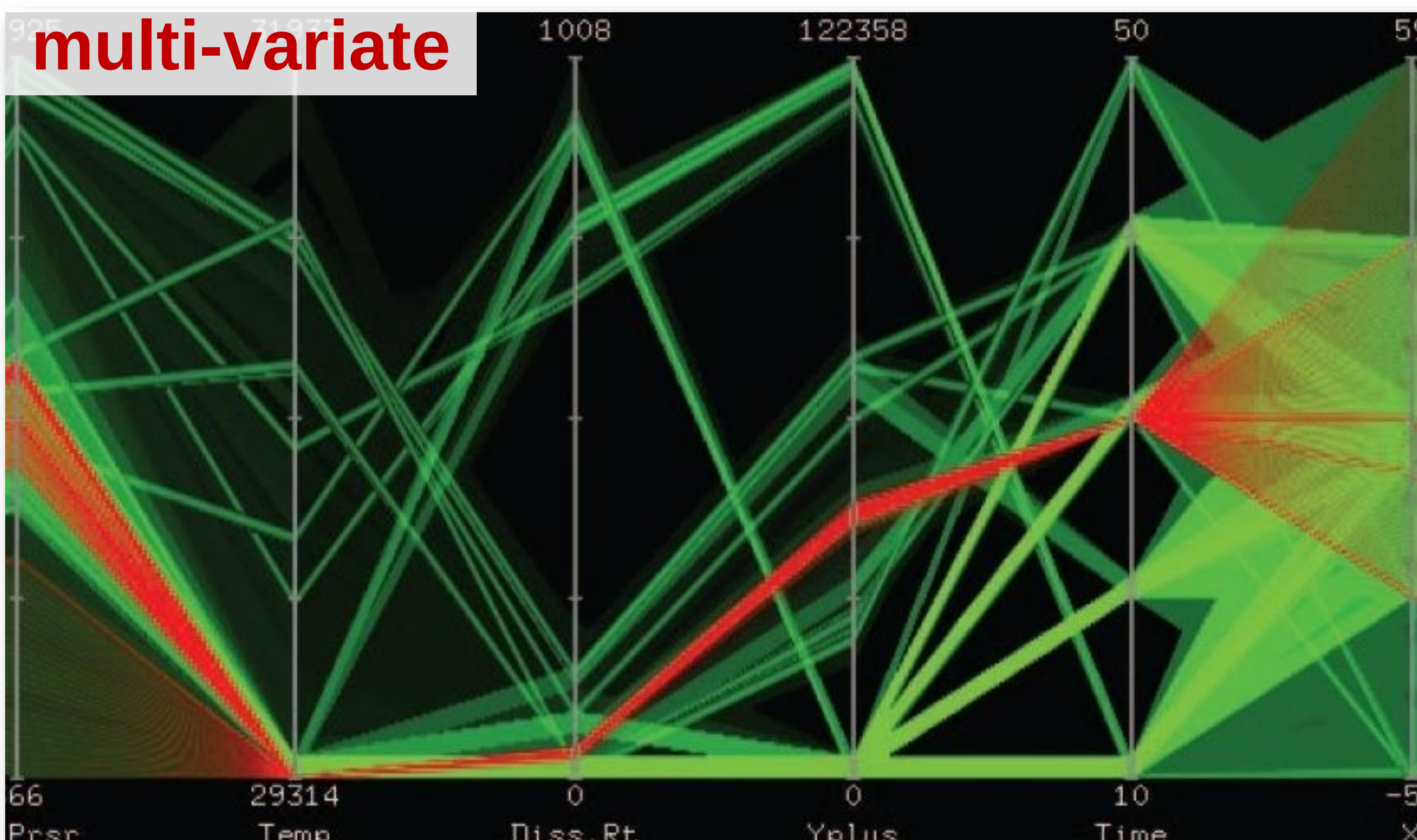


SIEMENS

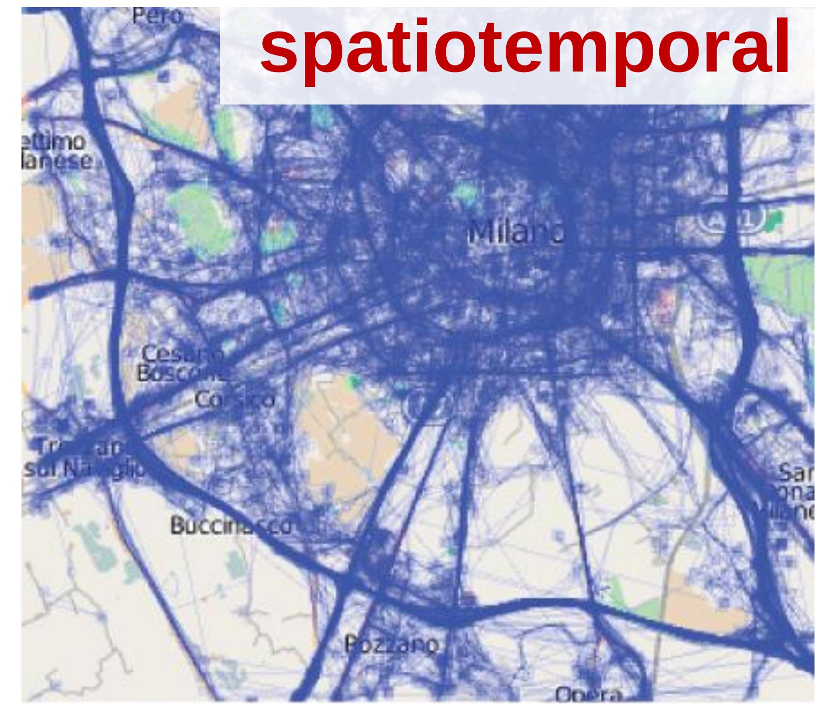


Algorithmic extraction of values & patterns

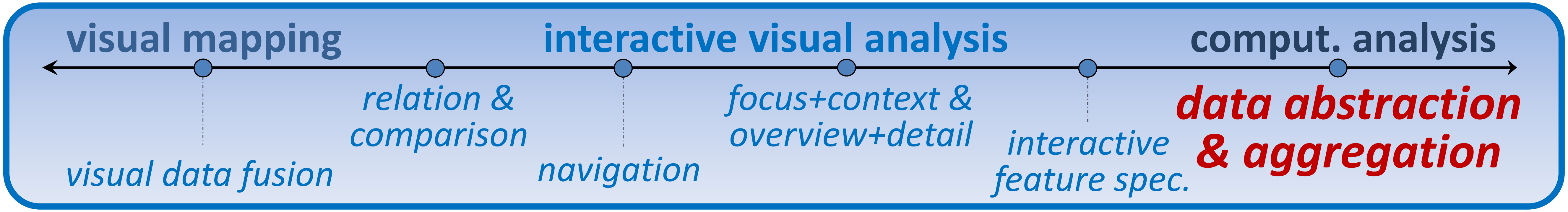
- Dimensionality reduction
- Aggregation, summary statistics
- Clustering, classification, outliers, etc.



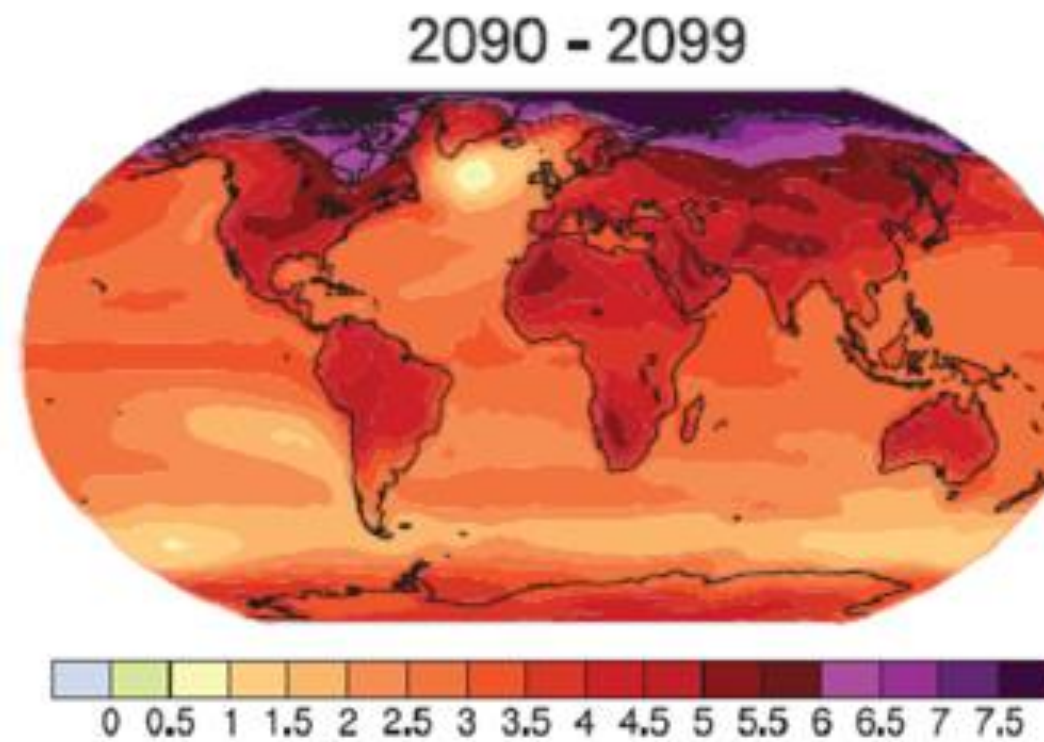
Clustering+outlier preservation [Novontný & Hauser 06]



[Andrienko & Andrienko 11]

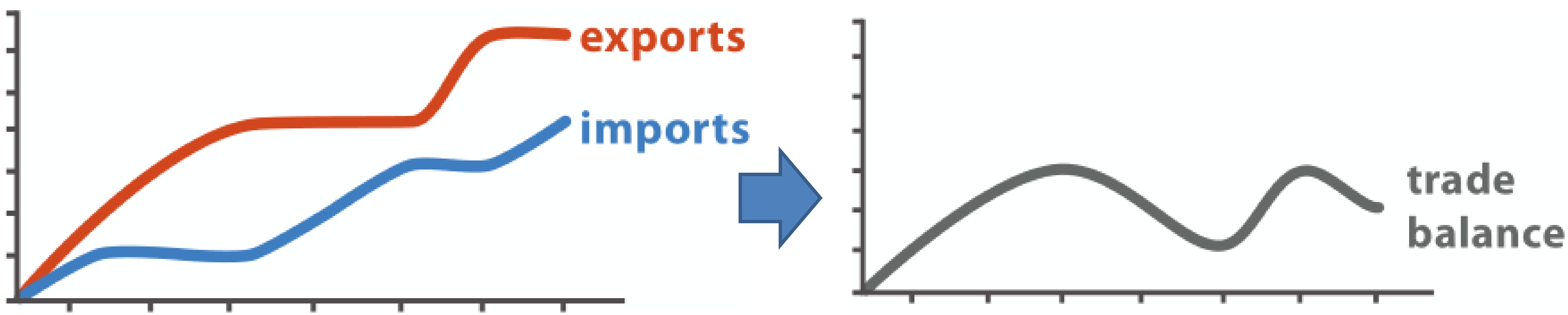


- Summary statistics
  - Mean, variance, etc.
- Derive new data attributes
  - What do you want to see?



Average temp. in ten years (° C)

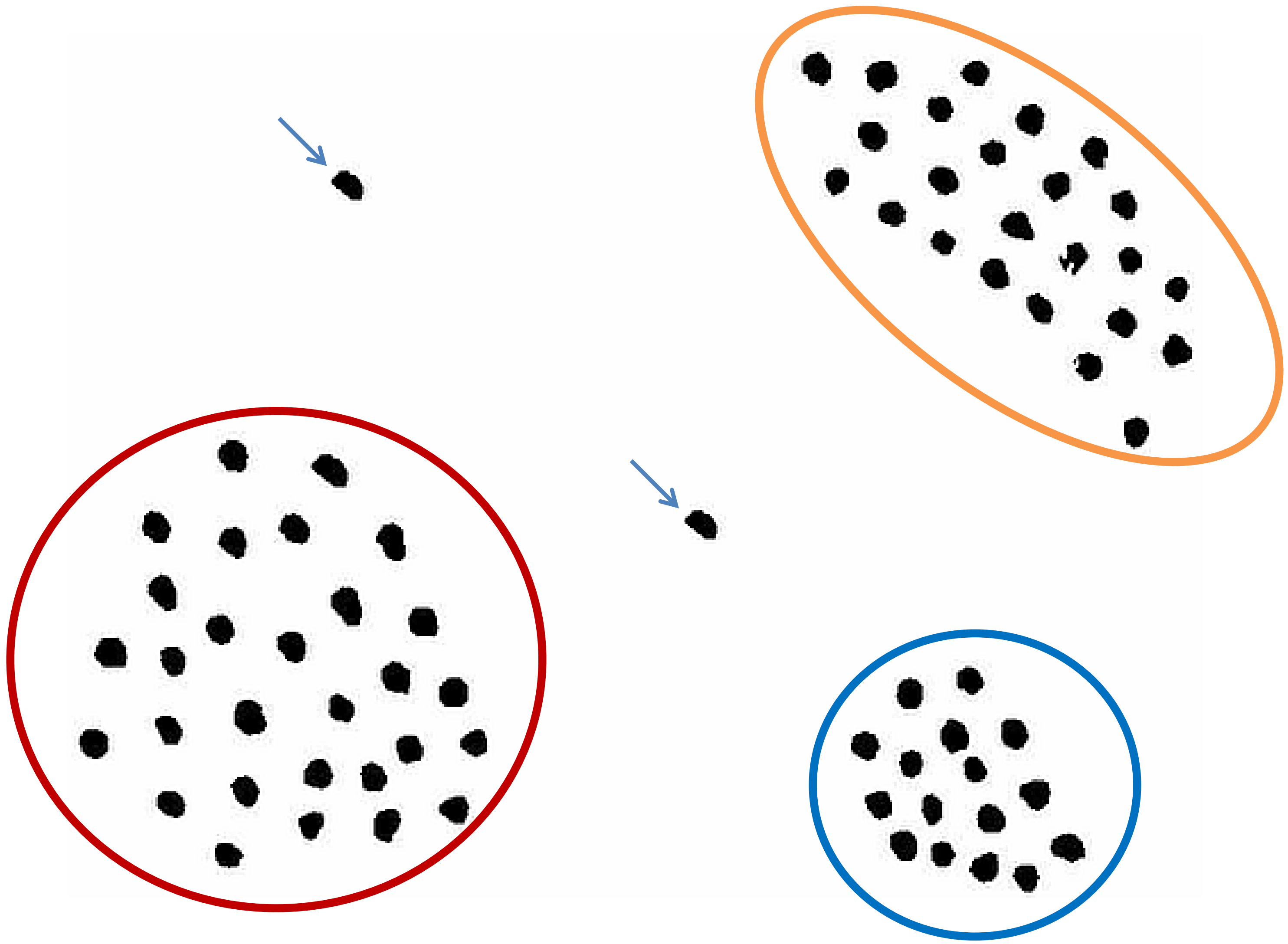
[from IPCC AR #4, 2007]

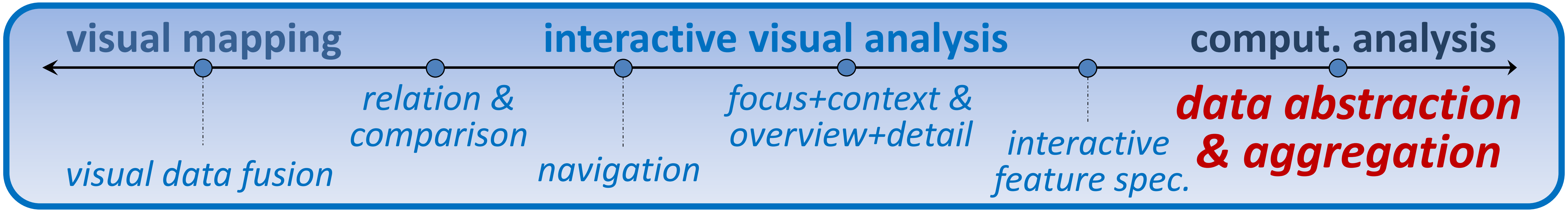


trade balance = exports − imports

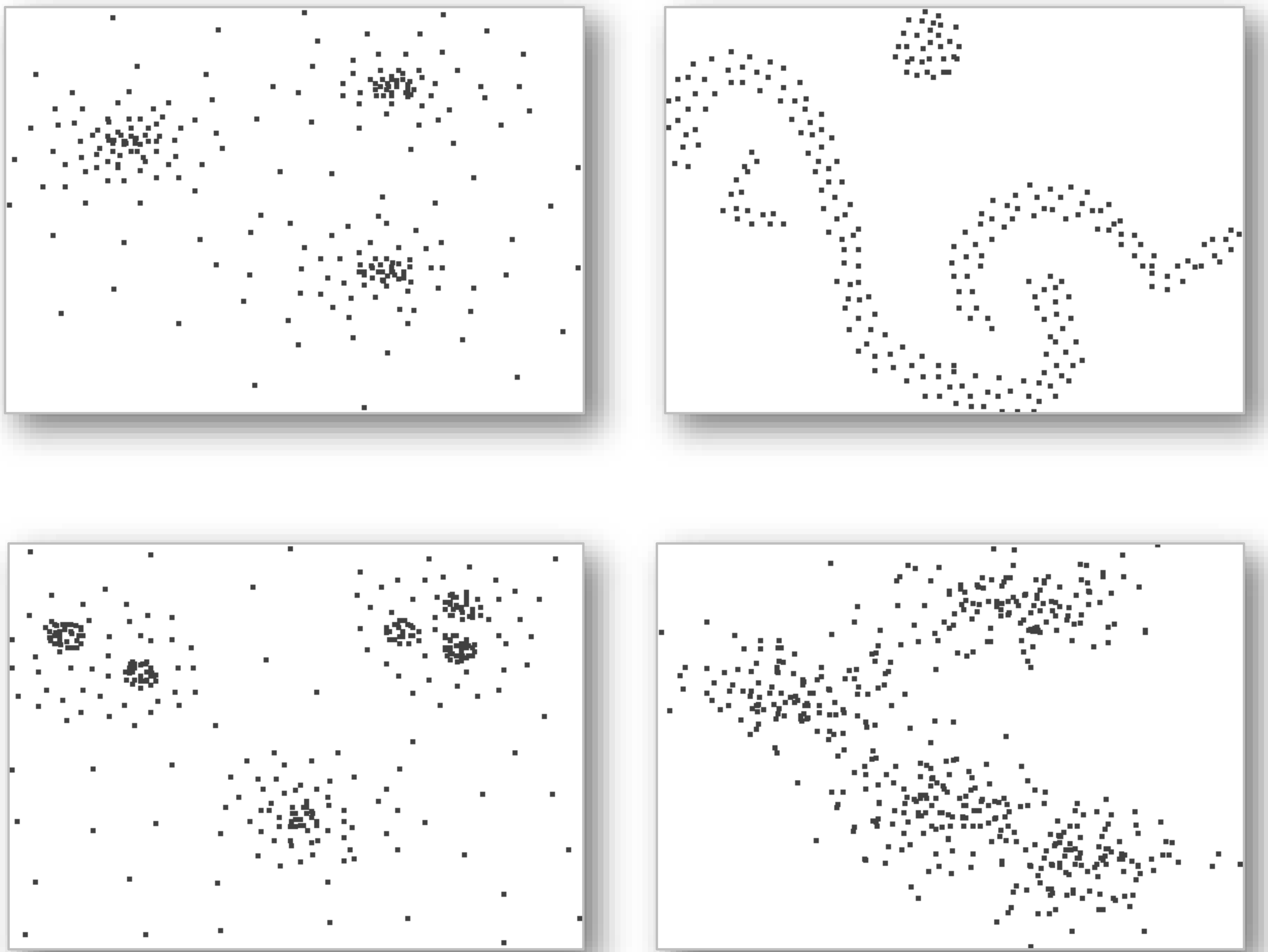
Clustering

– Given some data points, we'd like to understand their structure





How many clusters do you see?

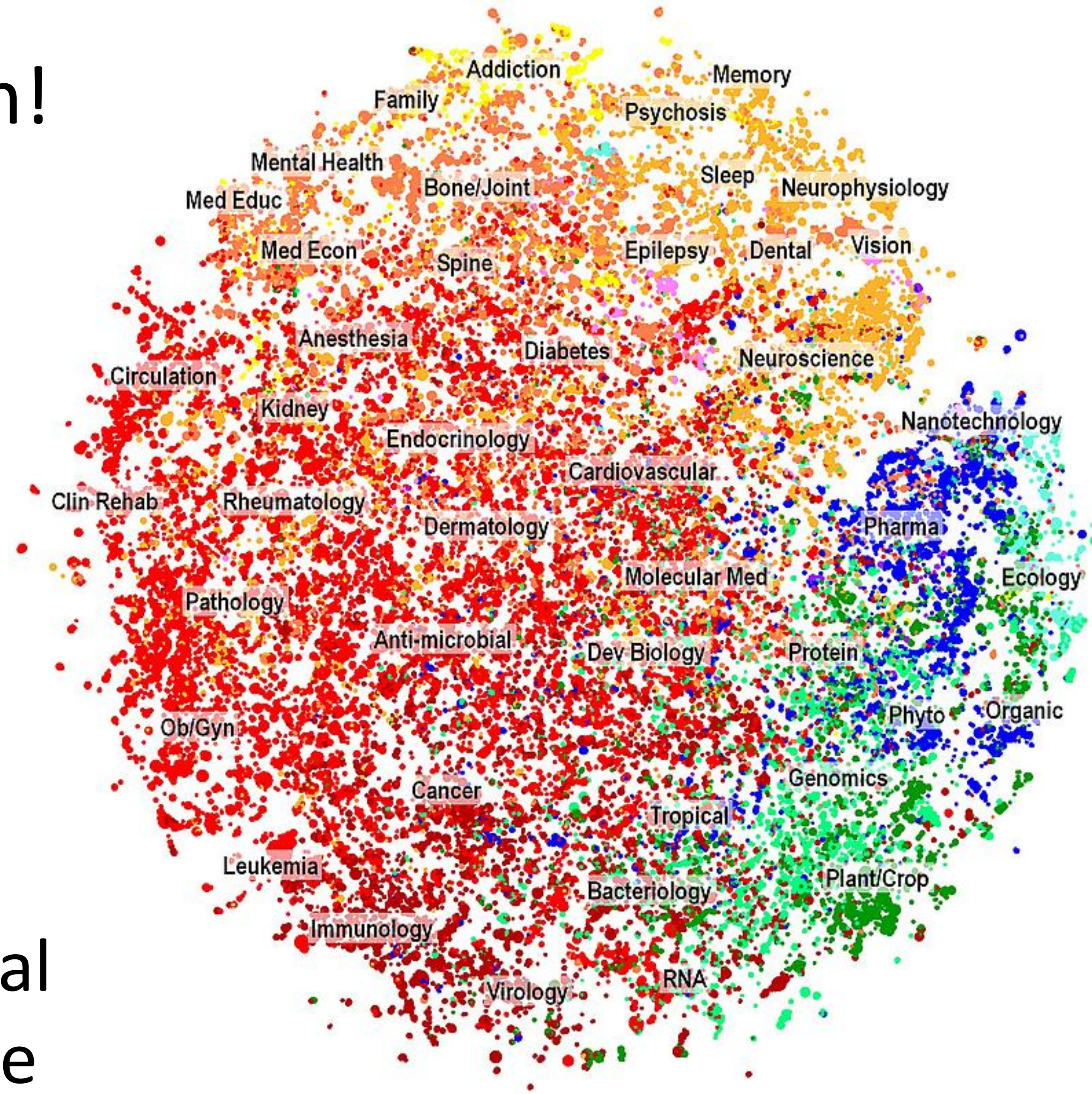


Clustering is a hard problem!

- Clustering in 2D looks easy
- Clustering small amounts of data looks easy

However,

- many applications involve 10 or 10,000 dimensions
- distances in high-dimensional spaces are at similar distance

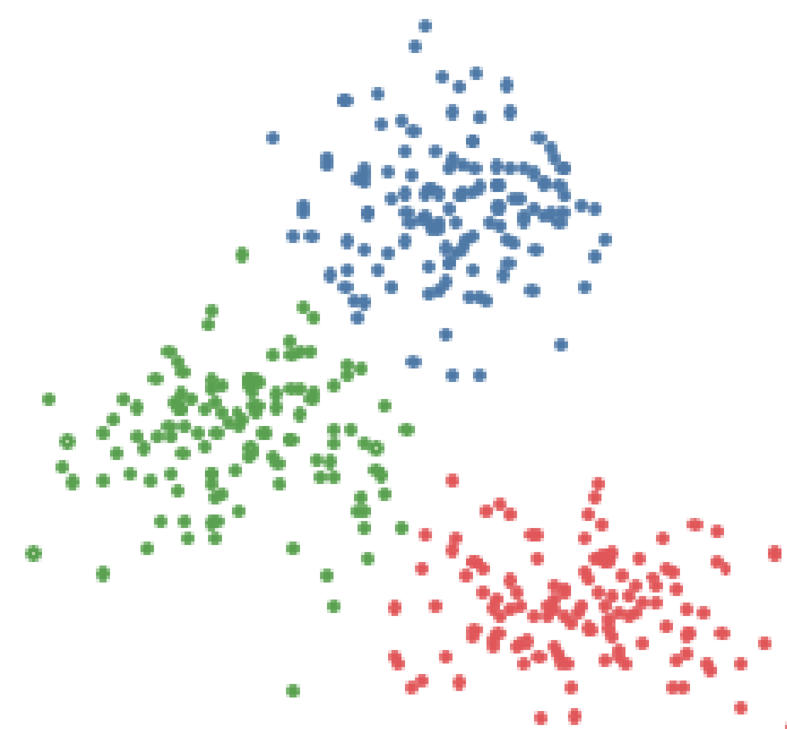


Clustering more than 2 mill. biomedical publications
Boyack et al., PLoS ONE, 2011.

Clustering

- Given a **set of points** with some **notion of distance** between points, **group** them into **clusters** such that
  - Members of a cluster are close/similar to each other
  - Members of different clusters are dissimilar

- Usually
  - Points are in high-dimensional space
  - Similarity defined by distance measure (e.g. Euclidean)



http://tech.nitoyon.com/en/blog/2013/11/07/k-means/

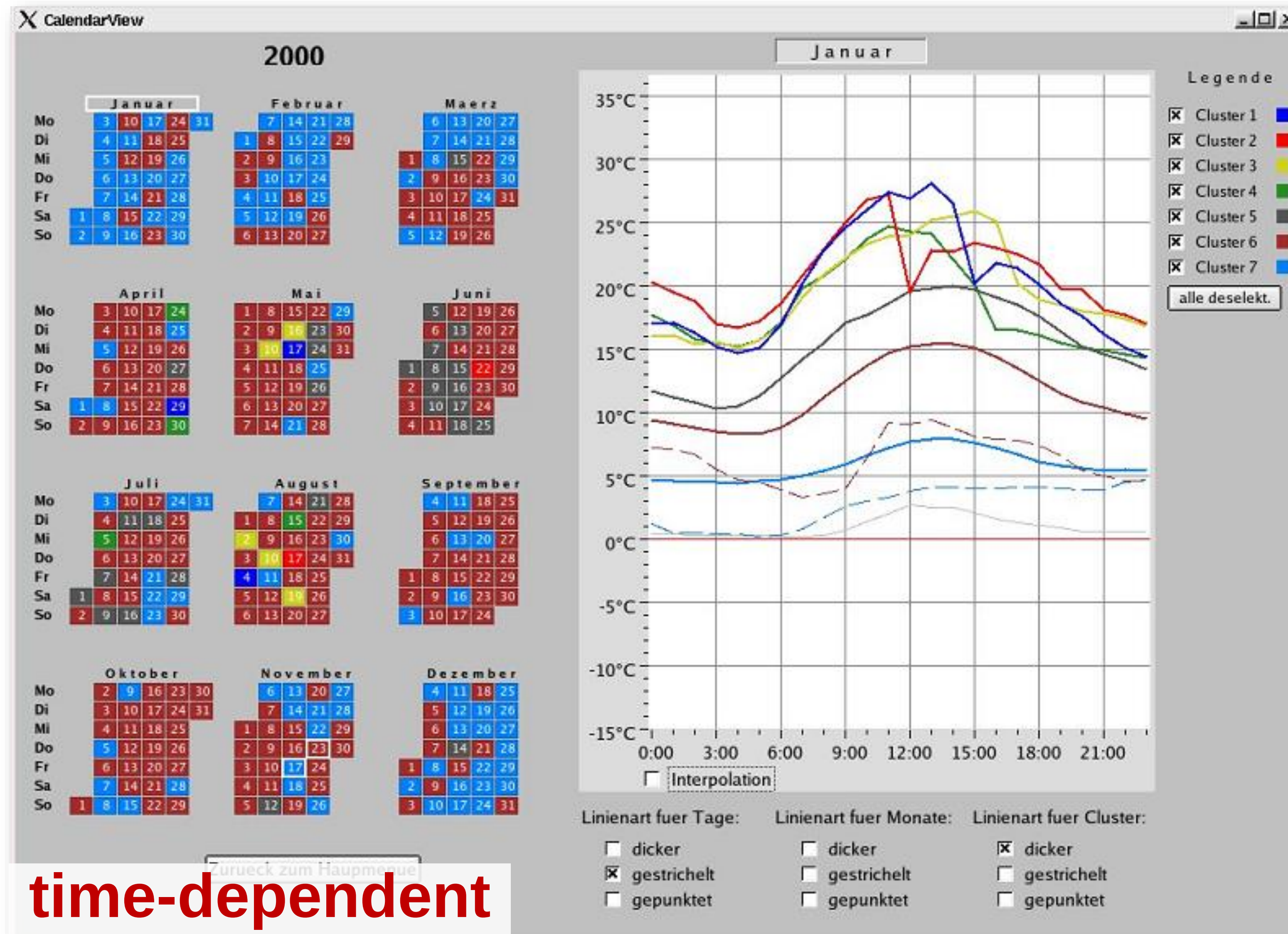
visual mapping | interactive visual analysis | comput. analysis

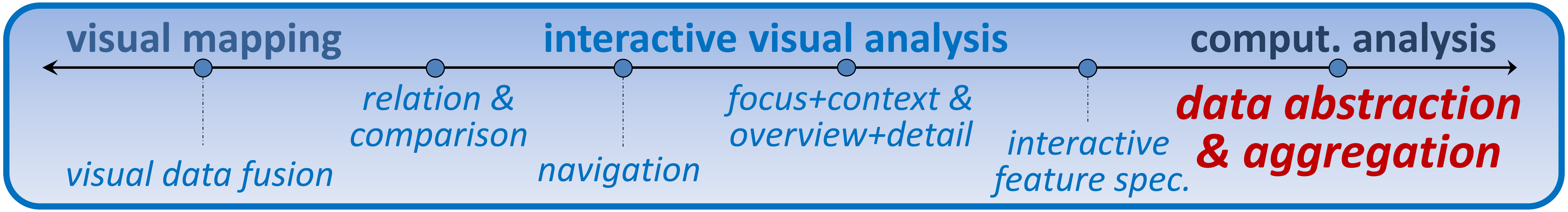
visual data fusion · relation & comparison · navigation · focus+context & overview+detail · interactive feature spec. · **data abstraction & aggregation**

Cluster Calendar View

[vanWijk & van Selow ’99]

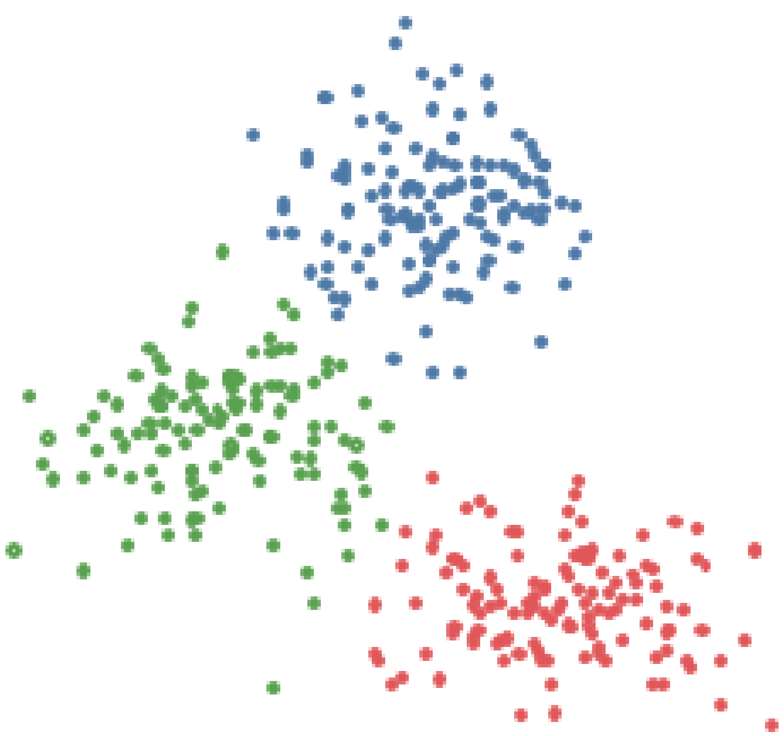
- Time series clustered by similarity (K-means)
- Cluster affiliation of daily pattern shown in calendar



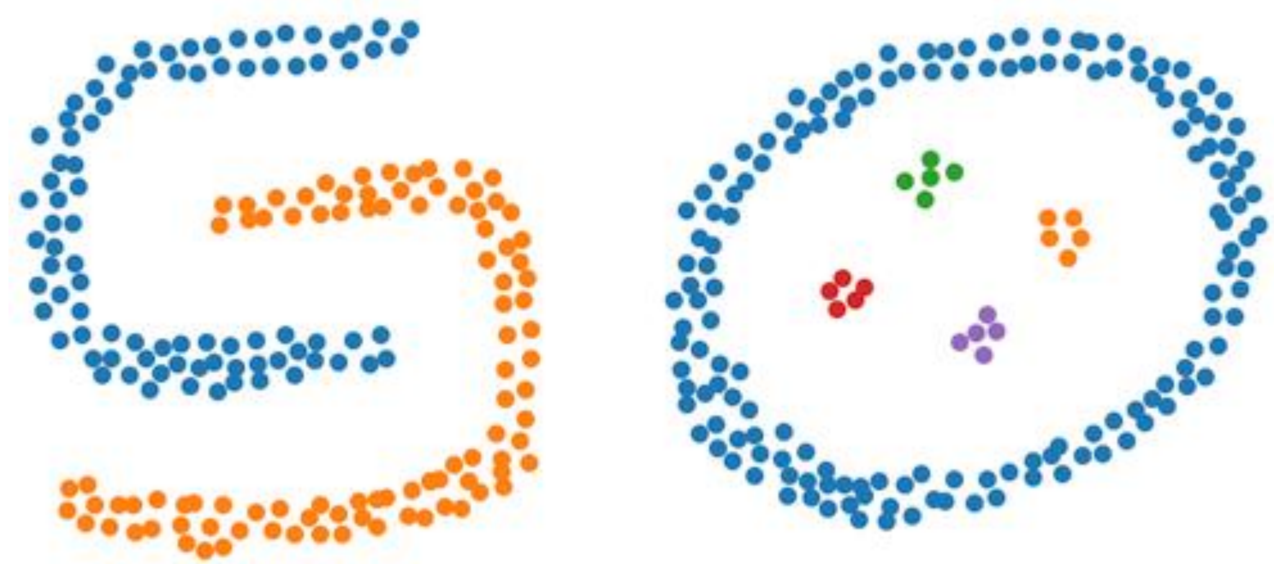


Density-based Clustering (DBSCAN)

- Identify dense regions in data
- Clusters can be arbitrarily shaped
- Difficult to find good parameter settings



k-means



DBSCAN

[Ester et al.]

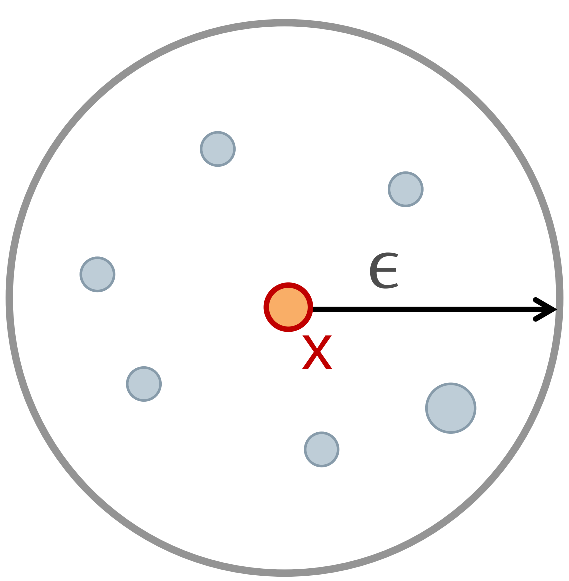
DBSCAN: 2 parameters

- Radius ϵ
- Number of MinPts

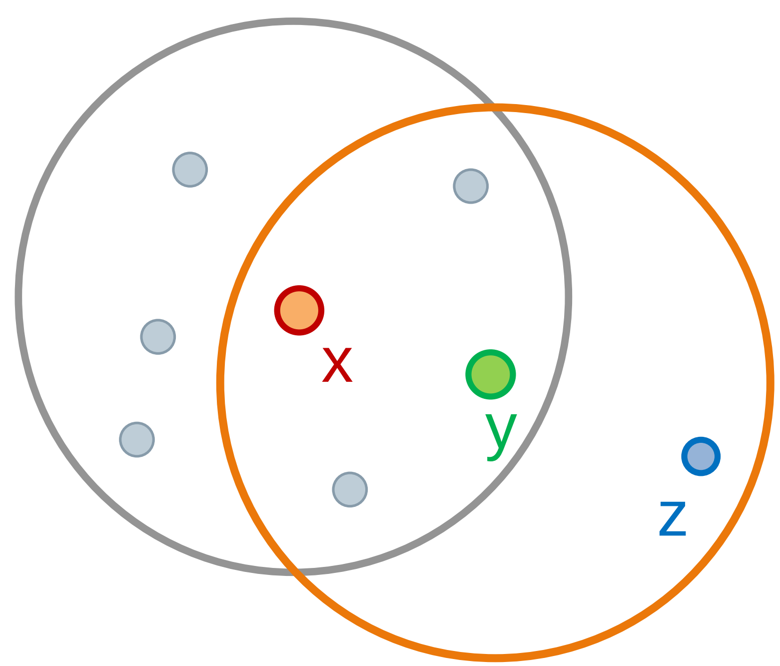
core point (ϵ-neighborhood contains at least minimum number (*MinPts*) of points)

border point (in the ϵ-neighborhood of core point)

noise (neither a core object nor a border object)

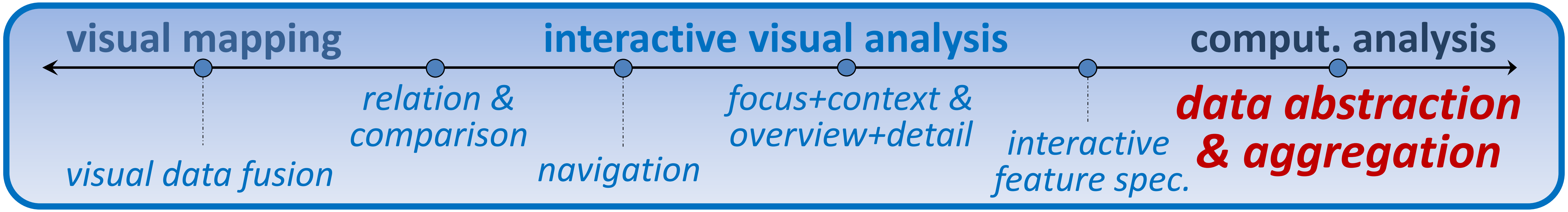


MinPts = 6



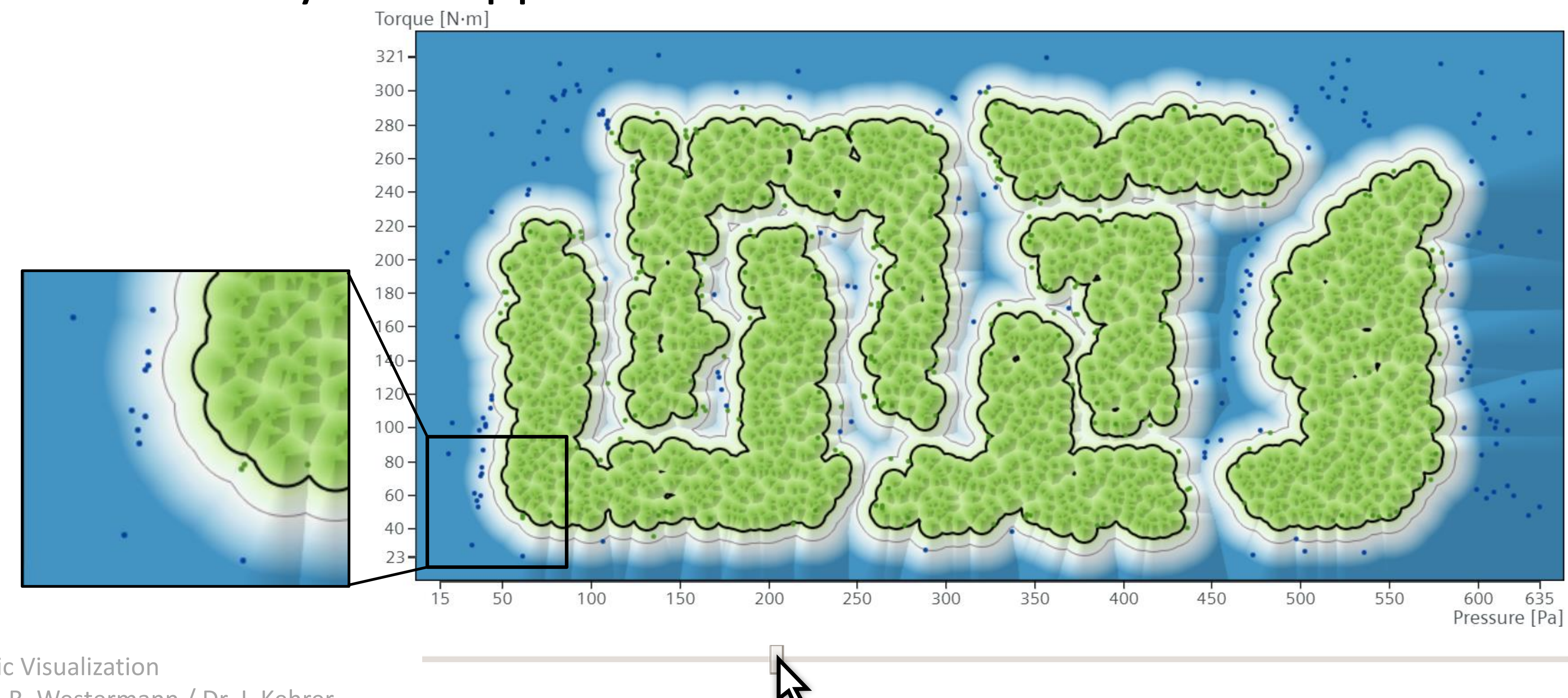
x … core point
y … border point
z … noise

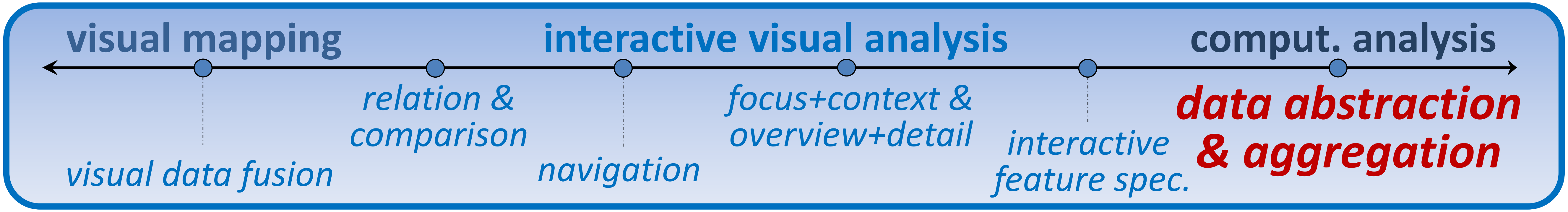
[Ester et al.]



Interactive parameter tuning

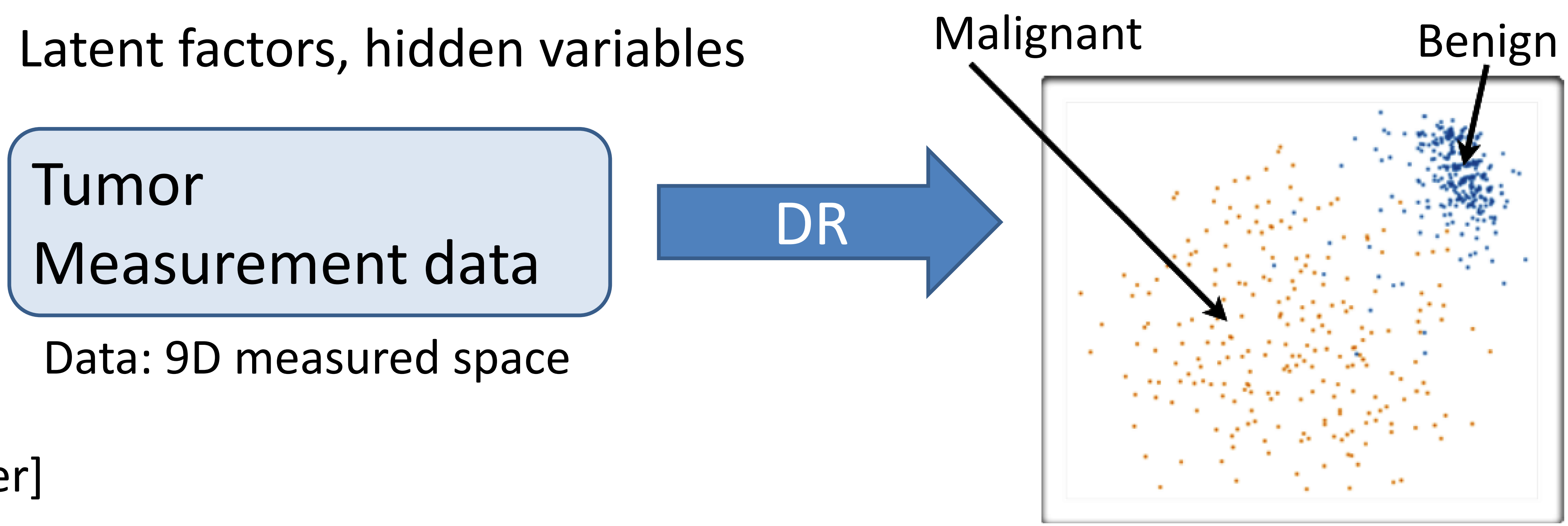
– Real-time approximation of DBSCAN result
– User changes parameter values
– Immediately sees approximation of result





Dimensionality reduction

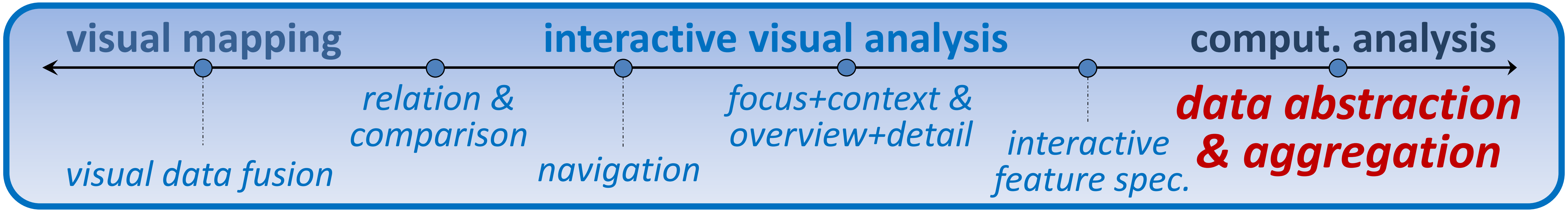
- Derive low-dimensional target space from high-dimensional measured space
- Use when you can't directly measure what you care about
  - True dimensionality of dataset assumed to be smaller than dimensionality of measurements
  - Latent factors, hidden variables



Data: 9D measured space

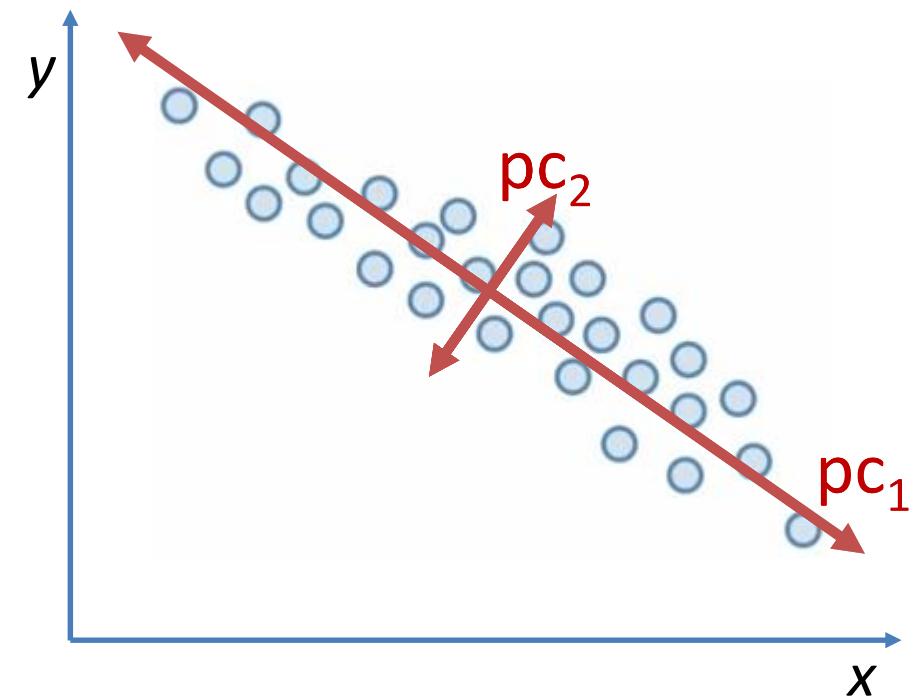
Derived data: 2D target space

[Munzner]



Principal Component Analysis (PCA)

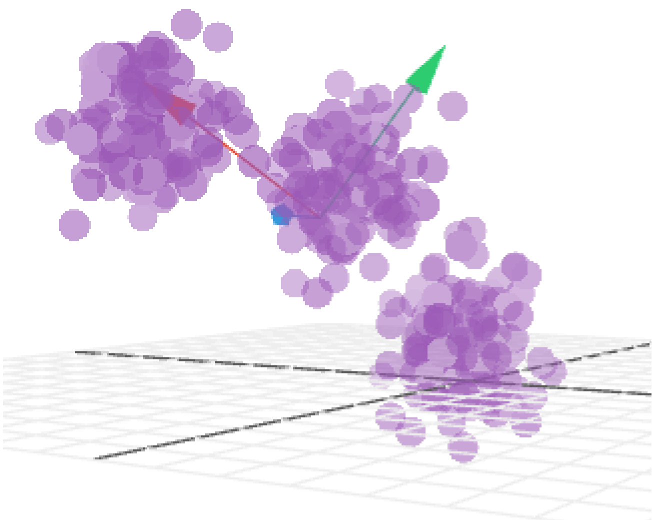
- Find directions of largest variance
- Neglect directions of small variance (not descriptive)
- Coordinate system transformation (rigid rotation)

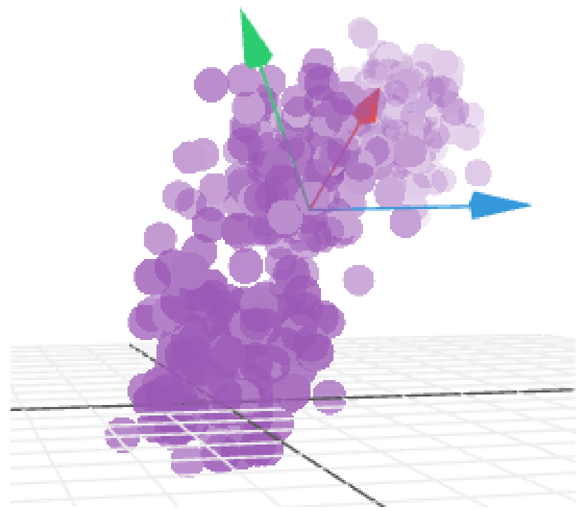


- Result
  - New axes (eigenvectors) & explained variances (eigenvalues)
  - New axes usually don’t mean anything physical

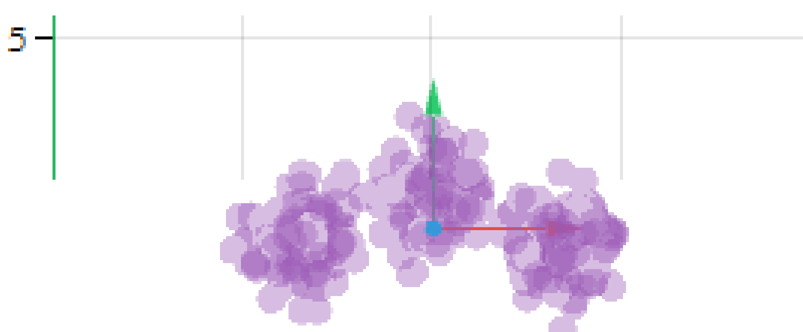
Principal Component Analysis (PCA)

– 3D example

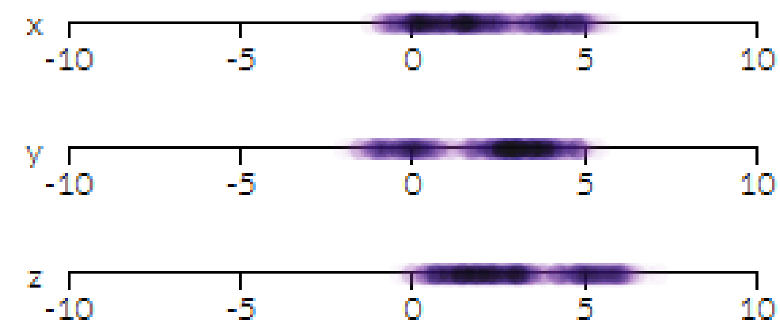




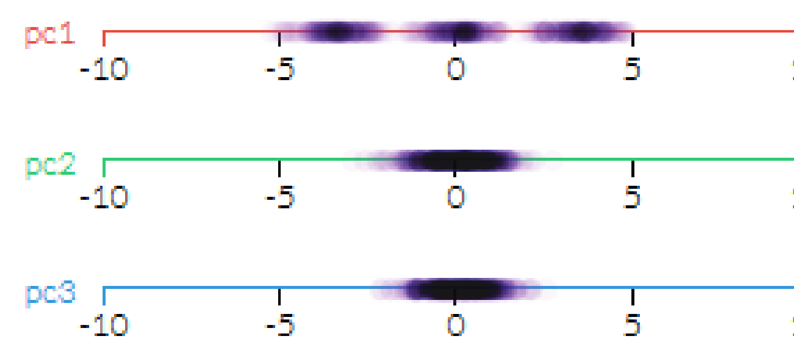
– Axes aligned with highest variation



Variation along x, y, z

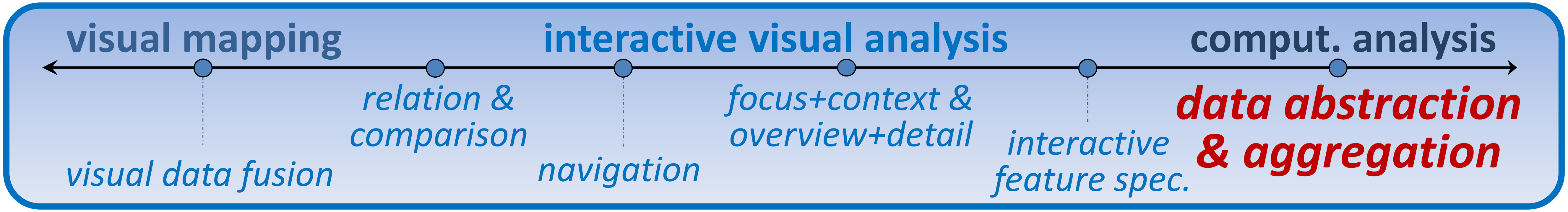


Variation along principal components

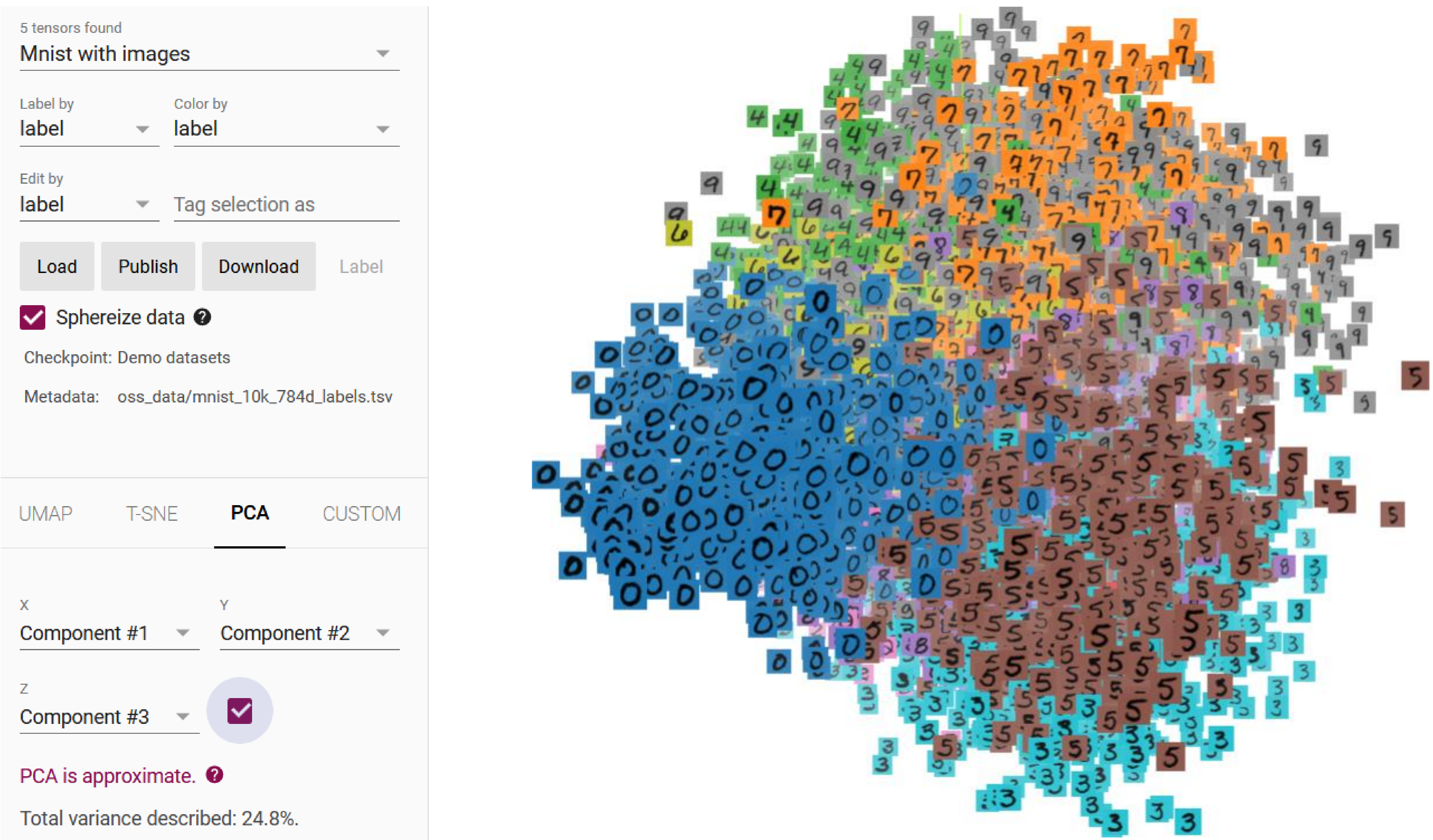


Using pc1, we can see 3 clusters

http://setosa.io/ev/principal-component-analysis/



Embedding Projector: Inspect results of dimensionality reduction

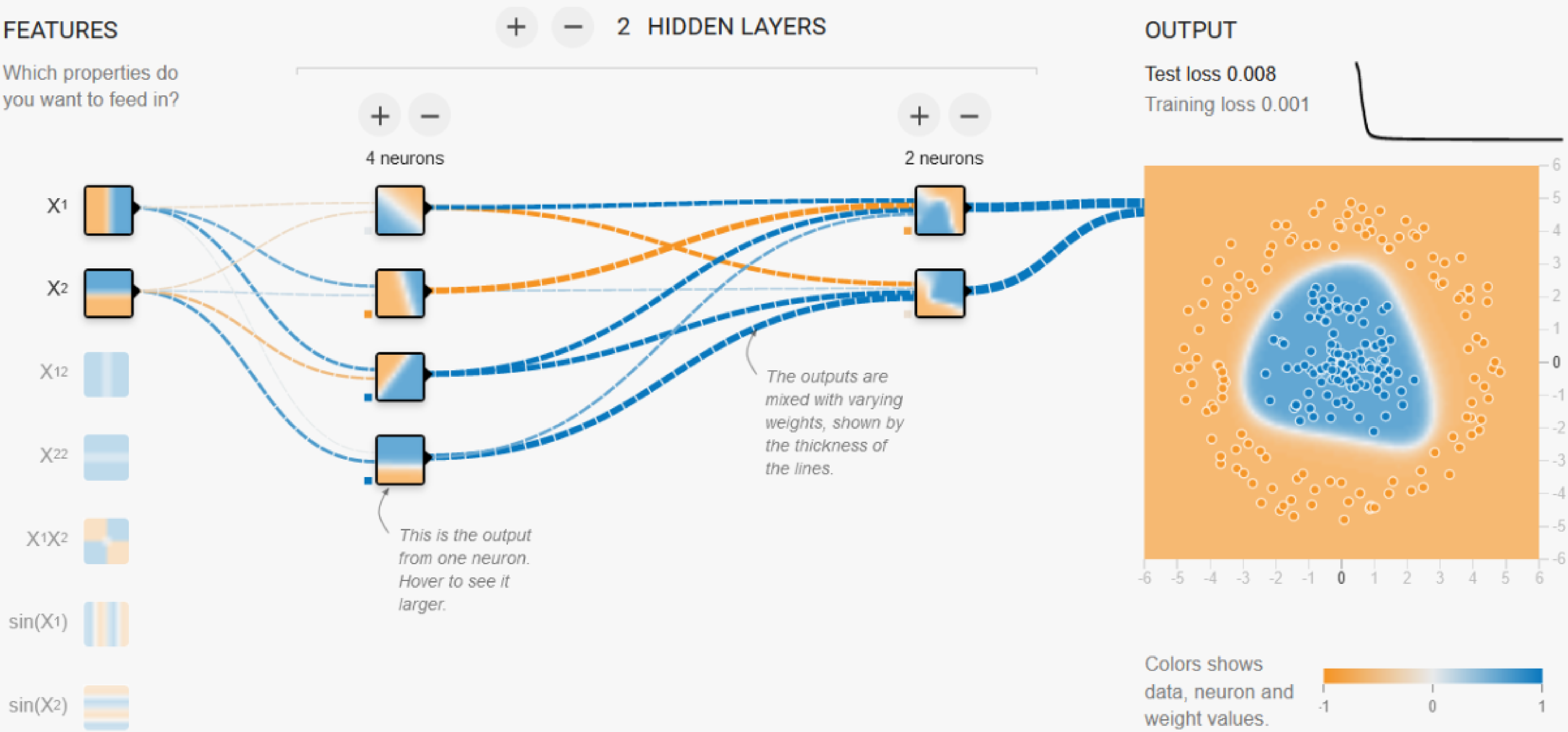


projector.tensorflow.org

visual mapping — interactive visual analysis — comput. analysis

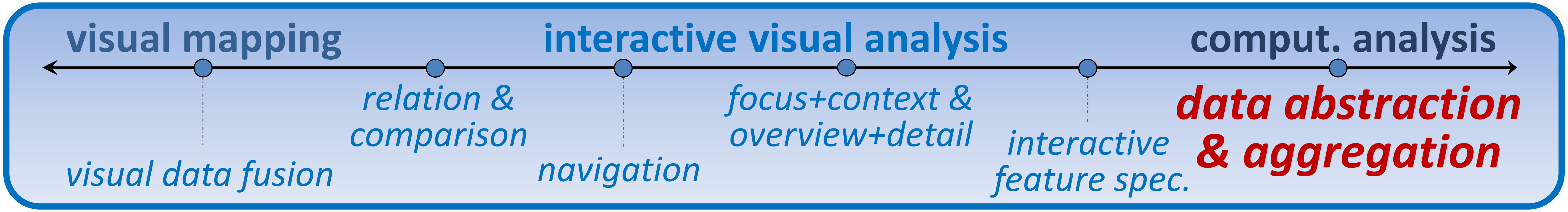
visual data fusion · relation & comparison · navigation · focus+context & overview+detail · interactive feature spec. · **data abstraction & aggregation**

Explainable AI (XAI) allows to open the black box



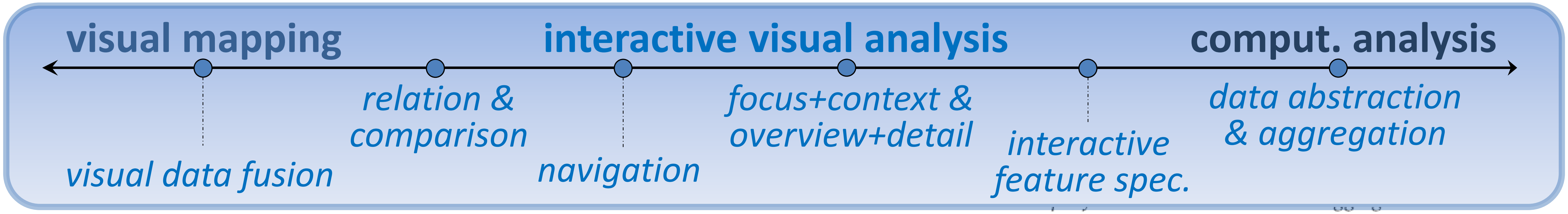
Tensor Flow Playground

playground.tensorflow.org



- Summary: Data abstraction & aggregation
  - Extract meaningful values/patterns (e.g., clustering)
  - Preserve main data characteristics (e.g., mean, variance, outliers) while suppressing irrelevant details

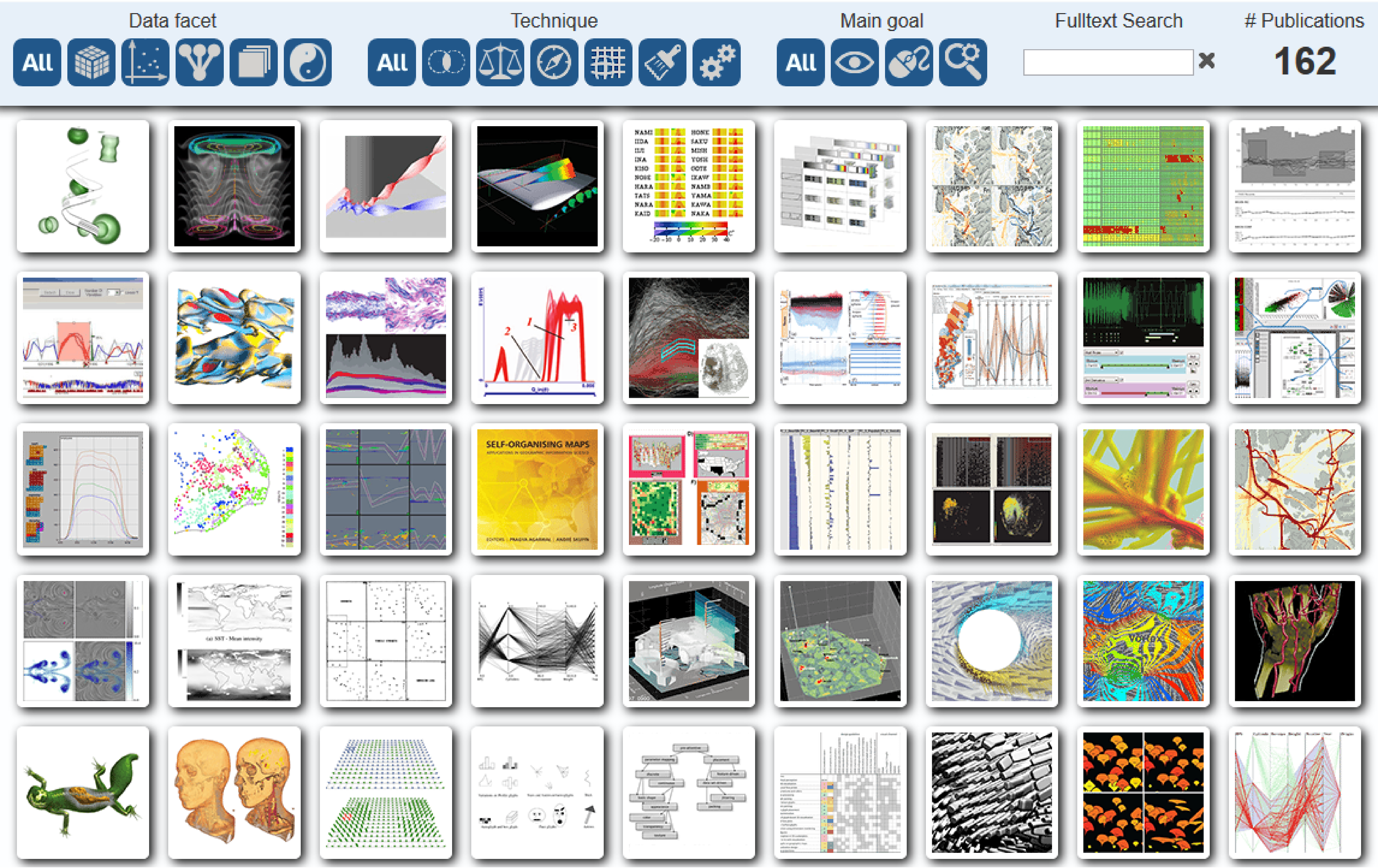
- Combine best of two worlds [Keim et al.]
  - data exploration/analysis by the user, based on interactive visualization
  - and data analysis by the computer, based on statistics, machine learning, etc.



spatio temporal	maps [13], [14], [92]; Helix glyphs [93]; flow maps [105]; function graphs [70], [71], [72]; Time Histograms [94], [110], [111]; chrono volumes [98]; illustrative techniques [99]; texture-based flow vis. [100]	2-tone coloring [20]; Helix glyphs [93]; juxtaposed views [19], [110]; difference views [107]	search, zooming and panning [40], [54]	2-tone coloring [20]; multi-level focus+context [71]; pixel-based multi-resolution techn. [104]	brushing [21], [70], [71], [95], [113]; transfer functions [110], [111]	aggregation [15], [103], [105]; trends [21]; flow features [82]; clustering [83], [84], [110]; PCA [17], [78], [85]; SOM [89], [90]; KDE [106], [107]; information theory [108]; wavelet analysis [109], [110]
multi-variate	attribute views [22], [50], [67]; color & texture [119]; layering [115], [124], [126]; 2-level volume rendering [127], [128]; glyphs [120], [121], [122], [123], [124], [125]	correlation fields [133]; operators [134]; multiple linked views [9], [26], [29], [73], [74], [76]	grand tour [47]; ScatterDice [46]; ranking & quality metrics [48], [130], [131], [132]	illustrative vis. [115], [116]; oulier-preserving methods [69]; smooth brushing [80]	brushing [9], [50], [74], [75], [112]; multi-dim. transfer func. [114], [115]; machine learning [91], [135], [136]	clustering [68], [130]; data binning [69]; PCA [78]; MDS [86], [87]; SOM [88], [89]; projections [47], [48], [130], [132]; point clouds [129]
multi-modal	resampling [138]; data model [142]; illumination model [143]; multi-volume rendering [128], [139], [143], [144], [145], [146]	difference views [107]; multi-image view [153]; nested surfaces [31], [154], [156]; features [44], [155]	viewpoint selection [49]	cutaway views [147], [139], [49]	transfer functions [143], [144]	registration [27]; mutual information [28]; comparison metrics [148], [151], [152], [133]
multi-run	glyphs & box plots [37], [43], [162], [163], [164]; shape descriptors [164]; families of surfaces [41]; spaghetti plots [35], [42], [165]	aggregated & multi-run data [36], [37], [41], [174]; HyperMoVal [51], [52]	aggregated & multi-run data [36], [37], [41]; parameter space nav. [51], [52]	aggregated & multi-run data [36], [37], [41]; simulation process vis. [173], [174]	trends & outliers [36], [37], [41]; visual steering [172]	overview statistics [31], [35], [36]; projections [41], [51], [170], [171]; operators [33]; PCA [169]; clustering [167], [168], [169]
multi-model	feature fusion across multiple data parts [37]	feature relation across data parts [37]	x	x	feature spec. across data parts [37]	x

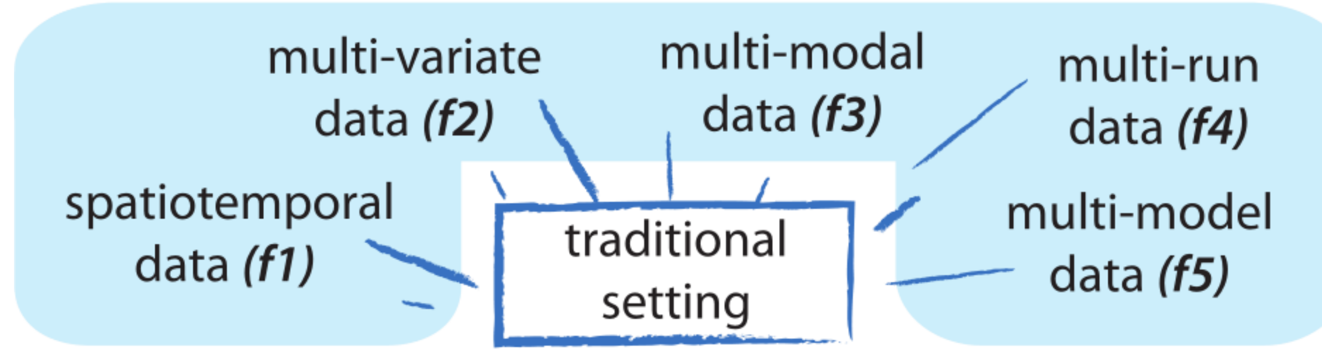
http://multivis.net [Kehrer & Hauser 13]

MultiVis.net



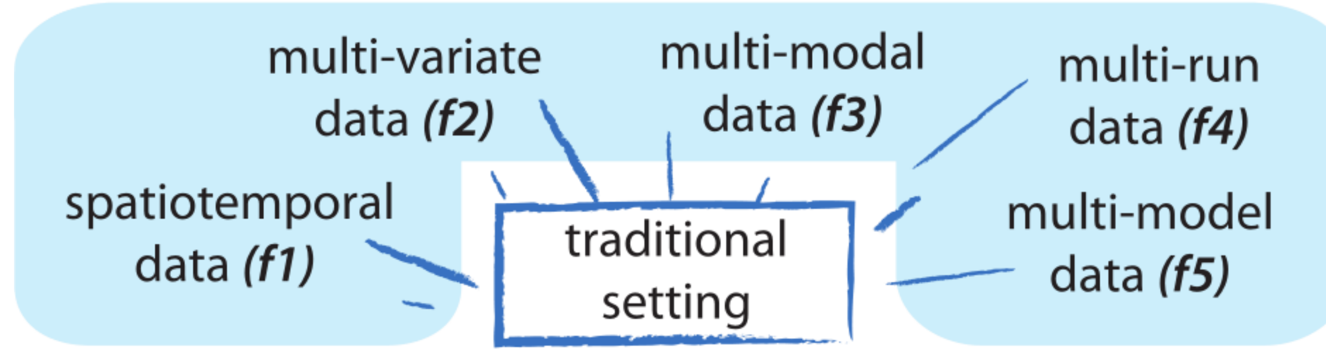
http://multivis.net [Kehrer & Hauser 13]

Open Issues

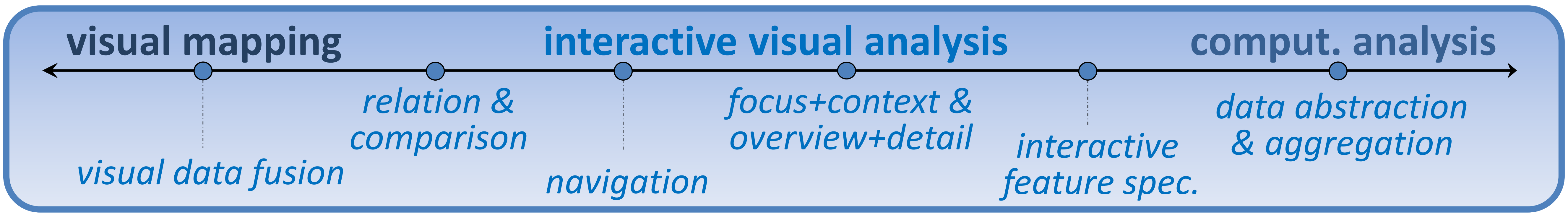


- How to deal with data heterogeneity?
  - Most approaches only address one or two data facets
  - Coordinated multiple views with linking & brushing
  - Investigation of features across views, data facets, levels of abstraction, and data sets
  - Fusion of heterogeneous data at feature/semantic level
- Combination of vis., interaction, and comput. analysis
  - Analytical methods can control steps in visualization pipeline (e.g., visualization mapping or quality metrics)
  - Interactive feature specification + machine learning

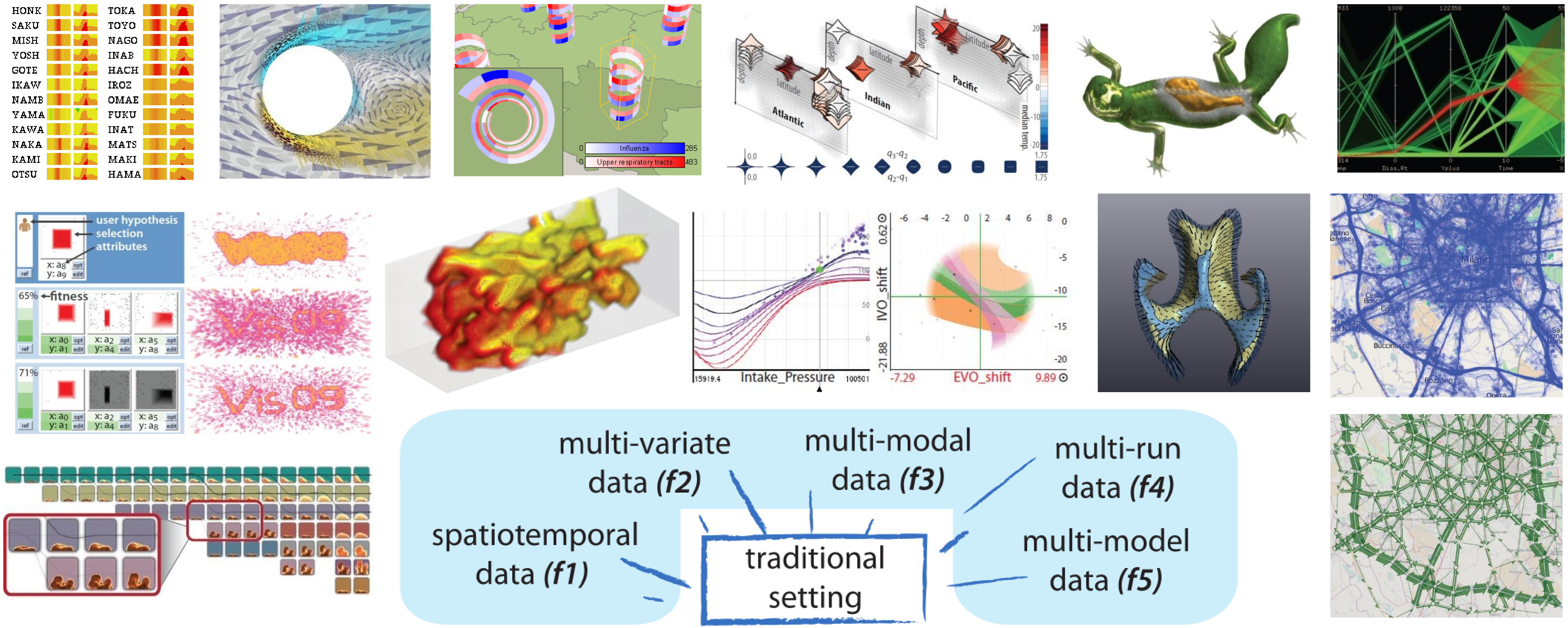
Conclusions



- Scientific data are becomming multi-faceted
- Categorization based on common visualization, interaction, and comput. analysis methods



- Promising data facets, e.g., multi-run & multi-model data



Acknowledgements

H. Hauser, H. Schumann, M. Chen, T. Nocke,

H. Piringer, M. Streit, M.E. Gröller,

VisGroup in Vienna/Bergen

Contact information

Dr. Johannes Kehrer

Siemens Technology
T DAI HCA-DE
Otto-Hahn-Ring 6
81739 München, Deutschland

E-mail:
kehrer.johannes@siemens.com

Internet
siemens.com/innovation

Intranet
intranet.ct.siemens.com