Course Introduction

Geometric Computer Vision

GCV v2021.1

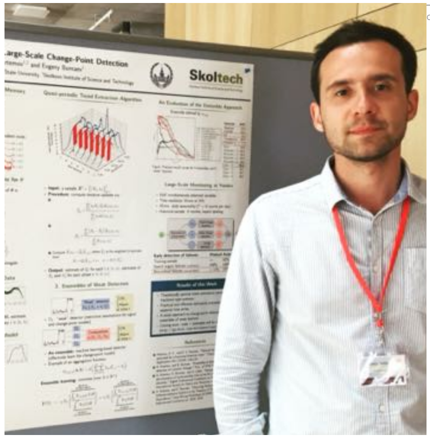
Alexey Artemov, Spring 2021

Your staff

Who am I?

Alexey Artemov, Ph.D.

- 2006–2012 Lomonosov MSU, *Physics*
- 2010–2012 Yandex Data School, *Data Science*
- 2011–2017 Yandex, Yandex Data Factory, Yandex Self-Driving Team, *Computer vision*
- 2012–2017 IITP RAS, Ph.D., *Statistics/ Data Science/Software*
- 2017–now Skoltech, *Computer vision*
- **Core:** software, statistics and data science, computer vision



Your TAs

Alexander Safin,
Ph.D. student, CDISE Skoltech

Sofia Potapova,
Computer vision developer
(ex-Yandex)

You

Outline: Course Introduction

§1. Getting to know each other & the GCV course [5 min]

1.1. Your staff [done]

1.2. Course goals

§2. Geometry processing applications [10 min]

2.1. What is geometric computer vision and why learn it?

2.2. What is a useful result of geometry acquisition?

2.3. Example applications

Outline: Course Introduction

§3. Course topics [10 min]

3.1. Topics covered by this course

3.2. Course flow and topic dependencies

§4. Organization, Q&A [10 min]

4.1. Learning scheme & assessment

4.2. Prerequisites, reaching out, contacts

§1. Getting to know each other & the GCV course

Skoltech
Skolkovo Institute of Science and Technology

Course goals

Skoltech
Skolkovo Institute of Science and Technology

§1. Getting to know each other & the GCV course

1.2. Course goals

- Provide an introduction into the **data representations, methods and models** in 3D/geometric computer vision
- Unlike traditional geometry modelling courses: **study where deep learning is useful and may bring advantages** in the geometry acquisition pipelines
- Hands-on experience with common **tools** for processing 3D data
- Obtain the **skills** needed to continue to engage in 3D/geometric vision

What you **will not** learn:

- Deep learning per se (only a subset of operations will be used, most are custom...)
- Techniques to obtain 3D data (will do a review but mostly assume these are given...)

Skoltech
Skolkovo Institute of Science and Technology

§2. Geometry processing applications

Skoltech
Skolkovo Institute of Science and Technology

What is geometric computer vision and why learn it?

§2. Geometry processing applications

2.1. What is geometric computer vision and why learn it?

Two main aspects:

- **Construct** 3D geometry representations suitable for various tasks from raw data (range images, volumetric CT and MRI, LIDAR)
  - Usually involves multiple steps going from low to high level
  - As an intermediate tool, requires analysis, e.g. segmentation
  - E.g.: Points -> meshes -> parametrized patch layout
- **Manipulate and analyze** geometry:
  - Deformations, boolean operations, comparisons, physically-based deformations (related to CAE)

What is a useful result of geometry acquisition?

§2. Geometry processing applications

2.2. What is a useful result of geometry acquisition?

- We briefly review several common applications
  - The goal is to understand better what is a useful result
  - For example: for geometry reconstruction how do we define acceptable quality?
  - E.g. accuracy of reconstruction:
    - 5-10 cm may be ok for cities (but barely)
    - 5 mm may be for furniture
    - << 1mm is needed for detailed objects, medical
  - Noise handling:
    - Outliers almost never ok
    - For medical, visual noise may not matter, need accuracy
    - For games/special effects/VR, better to remove some signal, vs. keep noise

Applications

2.2. What is a useful result of geometry acquisition?

Special effects

Slide Credit: Denis Zorin

Applications

2.2. What is a useful result of geometry acquisition?

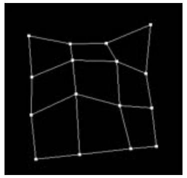
https://www.youtube.com/watch?v=zwU7io8Q7xo (see next slide, 3 min video)

- Highest-quality manually created geometry + physical simulation
- Different emphasis from CAD/CAE used in engineering: robustness + handling complexity, vs physical accuracy
- Driving a lot of research over years:
  - Animation+games industry (not serious): $250 bln
  - CAD/CAE industry (serious): ~$20 bln

Computer-Aided Geometric Design

2.2. What is a useful result of geometry acquisition?

- Traditional pipeline for modeling shapes from scratch

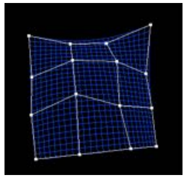


User defines a layout of surface patches and control points

Computer-Aided Geometric Design

2.2. What is a useful result of geometry acquisition?

- Traditional pipeline for modeling shapes from scratch

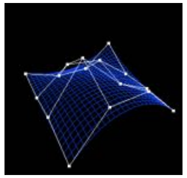


User defines a layout of surface patches and control points

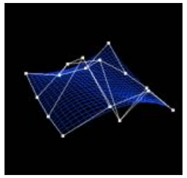
Computer-Aided Geometric Design

2.2. What is a useful result of geometry acquisition?

- Traditional pipeline for modeling shapes from scratch



User defines a layout of surface patches and control points

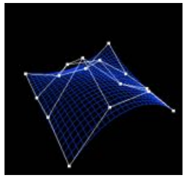


Editing is performed by moving control points and/or prescribing tangents

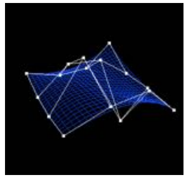
Computer-Aided Geometric Design

2.2. What is a useful result of geometry acquisition?

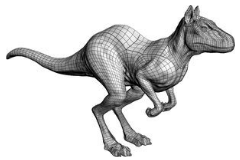
- Traditional pipeline for modeling shapes from scratch



User defines a layout of surface patches and control points



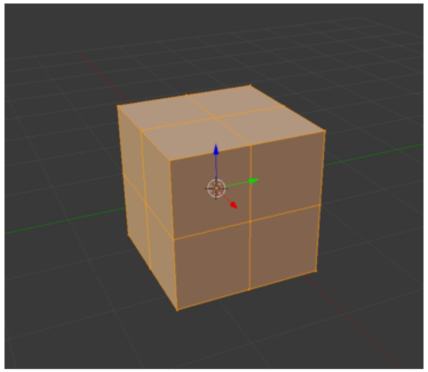
Editing is performed by moving control points and/or prescribing tangents



Patch-based construction of a surface

Blender Demo

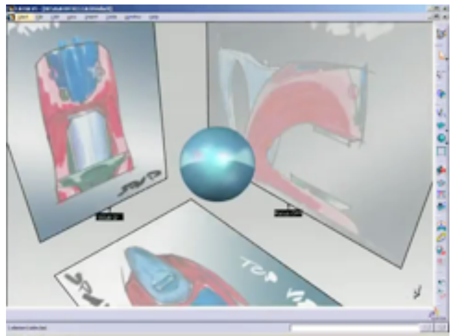
2.2. What is a useful result of geometry acquisition?



Computer-Aided Geometric Design

2.2. What is a useful result of geometry acquisition?

- High-quality surfaces
- Constrained modeling
- Requires a specific idea of the object first
  - Not easy to experiment and explore alternatives
- Requires training, skill and tedious work

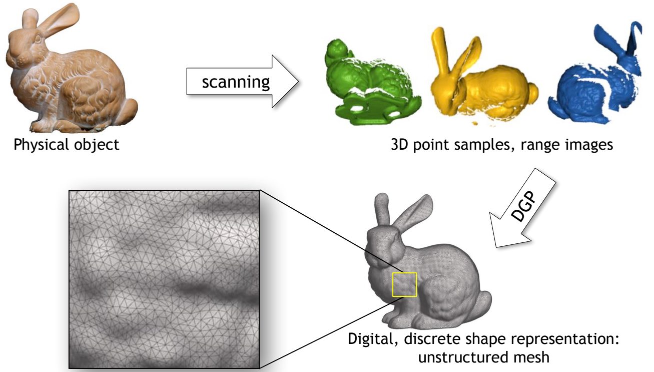


CATIA, Dassault Systemes

http://youtu.be/gTC5zMktMr0

Modern Geometry Acquisition Pipeline

2.2. What is a useful result of geometry acquisition?

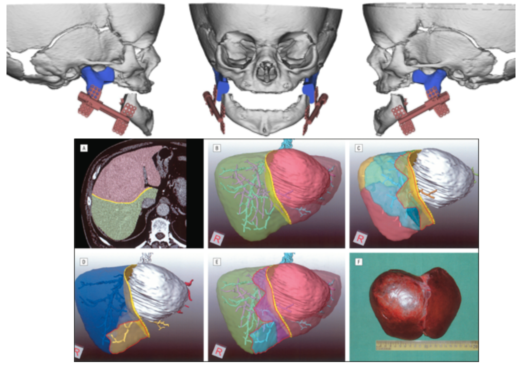


The Stanford Bunny, Stanford 3D Scanning Repository

Applications

2.2. What is a useful result of geometry acquisition?

Medical: e.g. virtual surgery planning



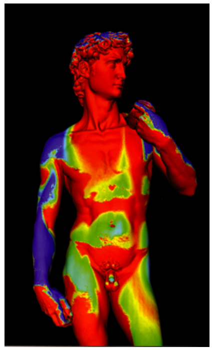
Digital Michelangelo Project

Cultural heritage preservation

1G sample points → 8M triangles   4G sample points → 8M triangles

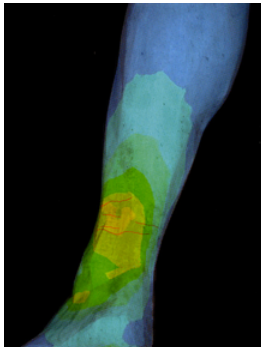
Digital Michelangelo Project

Cultural heritage preservation



Deposition of rainwater, dust, and other contaminants

Hypothetical placement of 16th century gilding



Tensile stresses in the left leg with the statue tilted 3 degrees forward, as it was in 1871.

§3. Course topics

Skoltech
Skolkovo Institute of Science and Technology

Topics covered by this course

Skoltech
Skolkovo Institute of Science and Technology

§3. Course topics

3.1. Topics covered by this course

Skoltech
Skolkovo Institute of Science and Technology

- Course structure for v2021.1: structured around data modalities/representations occurring in 3D
  - **Module 1: Geometry acquisition pipeline (today!)**
  - Module 2: Hardware systems for 3D data acquisition
  - Module 3: Dense range-images (+cameras and transformations)
  - Module 4: Point sets (+invariances in learning)
  - Module 5: Volumetric depth
  - Module 6: Implicit functions
  - Module 7: Surface-based representations
  - Module 8: CAD/Vectorized modalities (+shape programs)

Learning with...

§3. Course topics

3.1. Topics covered by this course

Skoltech
Skolkovo Institute of Science and Technology

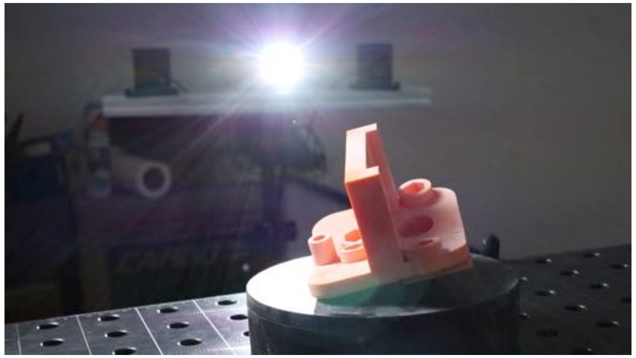
- Course structure for v2021.1: structured around data modalities/representations occurring in 3D
  - This year: few math-intensive methods, few
  - This year: focus on getting familiar with basic methods, models, and tools
  - Course updates: adding algorithmic / mathematical depth and detail

M1. Geometry Acquisition Pipeline

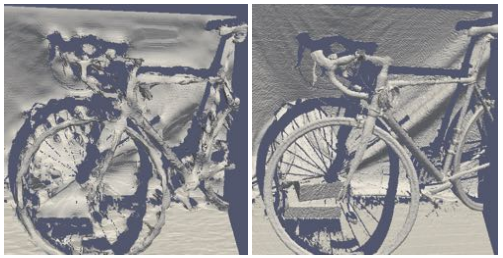
Scanning: results in range images ⇨ **Registration:** bring all range images to one coordinate system ⇨ **Stitching/reconstruction:** Integration of scans into a single mesh ⇨ **Postprocess:**
- Topological and geometric filtering
- Remeshing
- Compression

Slide Credit: Denis Zorin, Daniele Panozzo

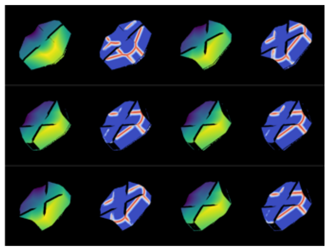
M2. Hardware systems for 3D data acquisition



M3. Dense range-images



Range-image super-resolution



Geometric feature line extraction

M4. Point sets

Point cloud generation

Recognition

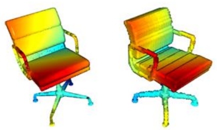
Generation

Invariances

Equivariances

M5. Volumetric depth

Skoltech
Skolkovo Institute of Science and Technology

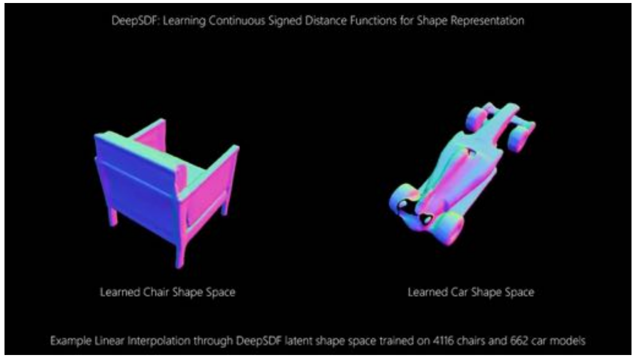


Sparse 3D CNNs

Depth fusion

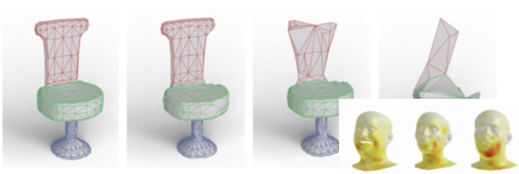
M6. Implicit functions

Skoltech
Skolkovo Institute of Science and Technology

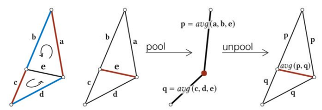


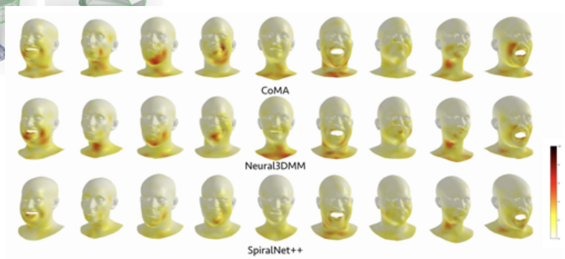
M7. Surface-based representations

Skoltech
Skolkovo Institute of Science and Technology



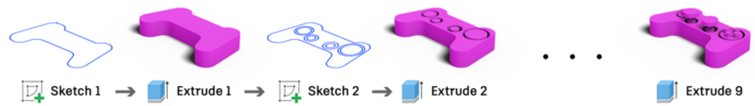
Convolutions on 3D meshes





M8. CAD/Vectorized modalities

Skoltech
Skolkovo Institute of Science and Technology



Constructive solid geometry trees

Boundary representations

Slide Credit: Denis Zorin, Daniele Panozzo

Course flow and topic dependencies

§3. Course topics

3.2. Course flow and topic dependencies

Structure of typical module:

- Monday: Pre-recorded lecture [45 – 60 min]
- Wednesday: Read supplementary [30 – 60 min]
- Wednesday: Offline quiz [should take 15 min, 48 hours] → submit for assessment
- Friday: Live practical [45 – 60 min]
- Friday: Live/offline Lab exercise [45 – 60 min] → submit for assessment
- Home assignment: offline [120 min] → submit for assessment

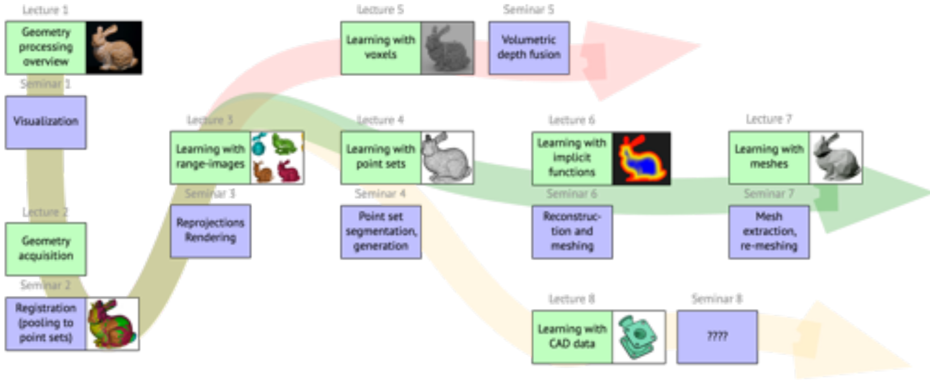
§3. Course topics

3.2. Course flow and topic dependencies

	Term 3 Week 1	Term 3 Week 2	Term 3 Week 3	Term 3 Week 4	Term 3 Week 5	Term 3 Week 6	Term 3 Week 7	Term 3 Week 8
Monday Lecture	Geometry processing overview	Geometry acquisition	Range-images	Point sets	Voxels	Implicit functions	Meshes	CAD data
Wednesday t Quiz								
Friday Practical	Visualizatio n	Registration (pooling to point sets)	Reprojectio ns Rendering	Point set segmentati on,	Volumetric depth fusion	Reconstruc-tion and meshing	Mesh extraction, re-meshing	
Friday Labwork								

§3. Course topics

3.2. Course flow and topic dependencies



§4. Organization

Prerequisites, reaching out, contacts

§4. Organization

4.1. Prerequisites

- Machine learning (including deep learning):
  - We will not cover the basic concepts that you need. If you are feeling insecure with **basic optimization, derivatives, gradients, or convolutions**, this course will be difficult to follow.
- Programming in Python (including basic operations with linear algebra packages such as NumPy and NN packages such as PyTorch), Linux, Dockers and Git:
  - We will not cover **basics of Linux, Dockers or Git, or give any reference on Python language**. If your are not familiar with Python programming or any of these tools, we suggest you first take the basic course

§4. Organization

4.1. Timetable

- Monday 12:30—15:30: pre-recorded lectures [Alexey]
- Friday 9:00—12:00: practicals, labwork [Alexander, Sofia]

§4. Organization

4.1. Reaching out & contacts

- Chat URL: https://t.me/joinchat/Iu0XzIjEIU_zPzri
- E-mails: a.artemov@skoltech.ru, a.safin@skoltech.ru

Learning scheme & assessment

§4. Organisation

4.2. Learning scheme & assessment

- The goal of this course is to introduce you to the baseline techniques and improve knowledge, not evaluate you
- But Education asks us to still somehow do this…

Final grade =	25% ×	Computer labs	8 Labs
	54% ×	Homeworks	2 HWs
	21% ×	Test/quiz	8 Quizzes

Textbooks

1. Botsch, M., Kobbelt, L., Pauly, M., Alliez, P., & Lévy, B. (2010). *Polygon mesh processing*. CRC press.
2. Zhang, S. ed., 2013. Handbook of 3D machine vision: *Optical metrology and imaging*. CRC press.
3. Ma, Y., Soatto, S., Kosecka, J. and Sastry, S.S., 2012. *An invitation to 3-d vision: from images to geometric models* (Vol. 26). Springer Science & Business Media.
4. Hartley, R. and Zisserman, A., 2003. *Multiple view geometry in computer vision.*
5. Hoffman, C. M. *Geometric & Solid Modeling: An Introduction.*