

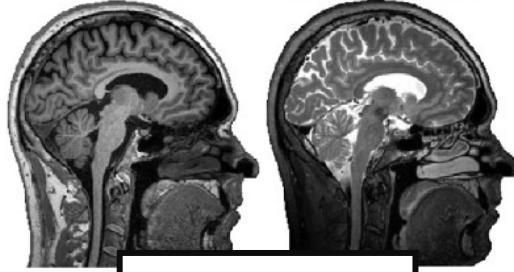
Medical Image Analysis

NBE-E4010

AI in medical imaging



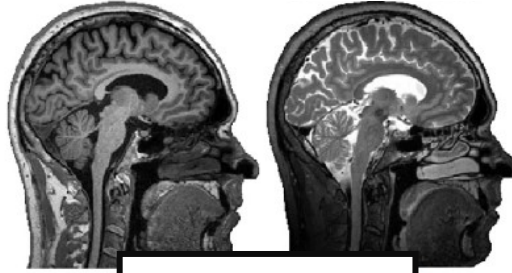
scanner



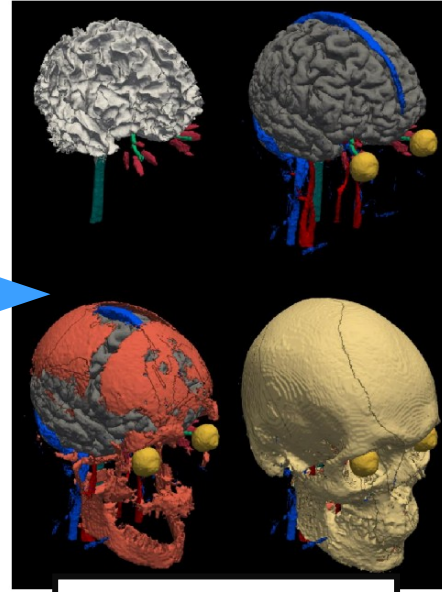
images

- acquire images faster
- visualize more details

AI in medical imaging



images



information

- expose the “unseeable”
- measure more consistently
- analyze images faster

Exposing the “unseeable”

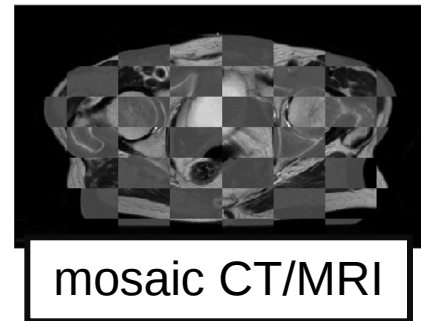
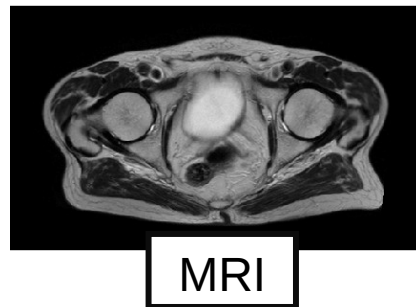
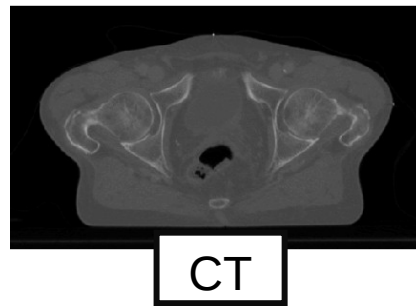
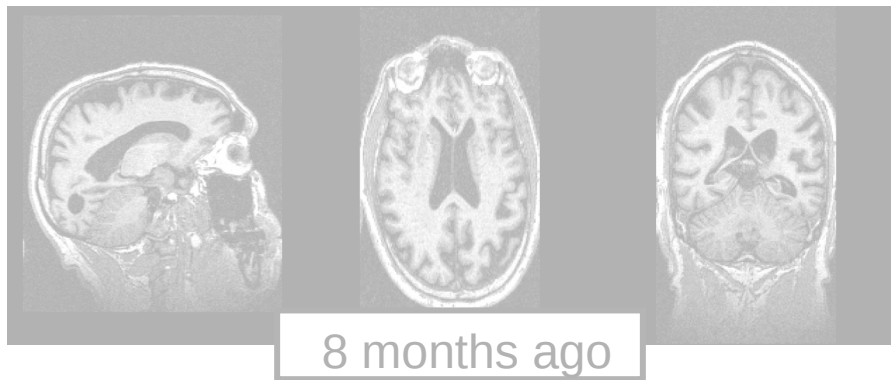
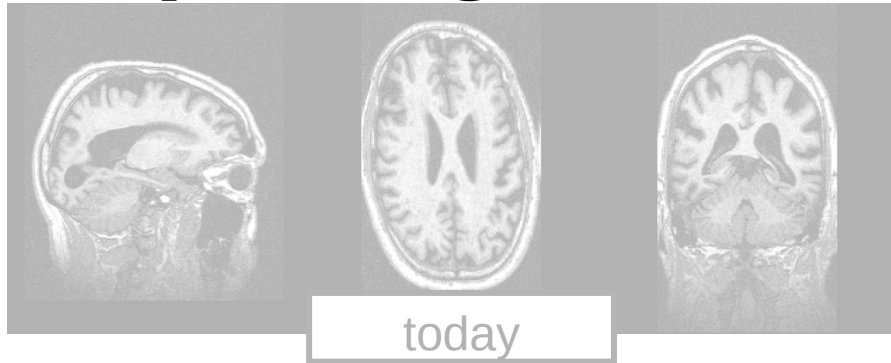


today

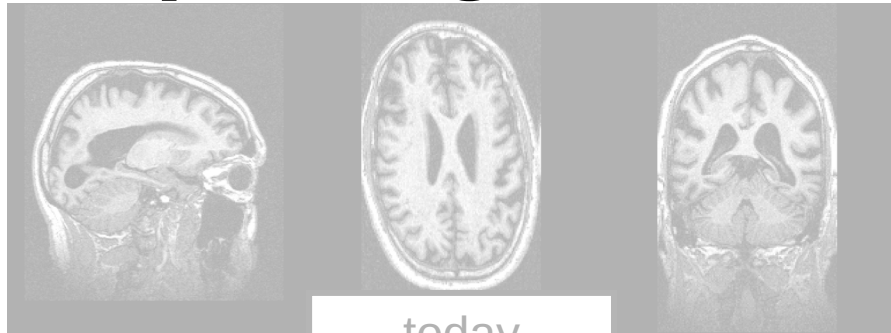


8 months ago

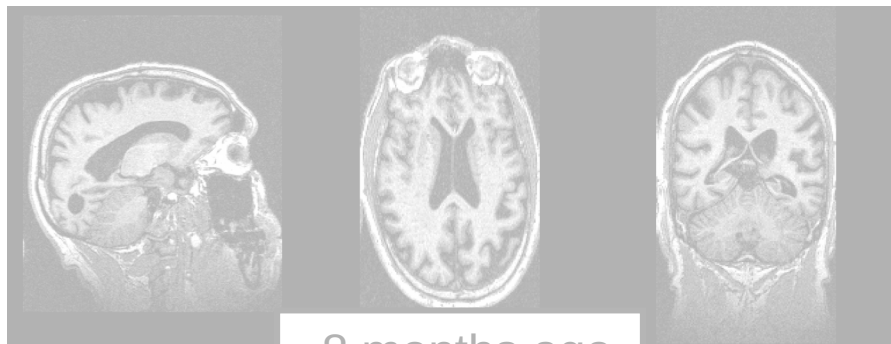
Exposing the “unseeable”



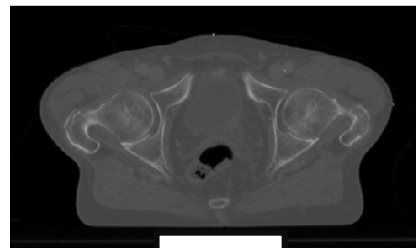
Exposing the “unseeable”



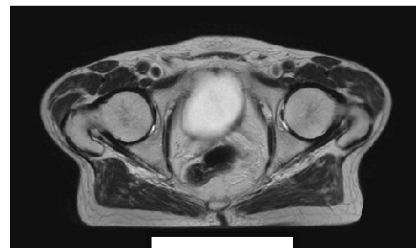
today



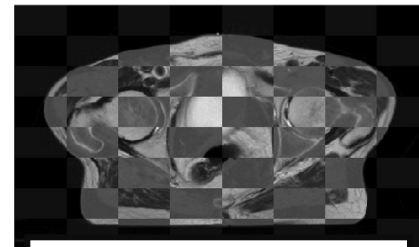
8 months ago



CT



MRI

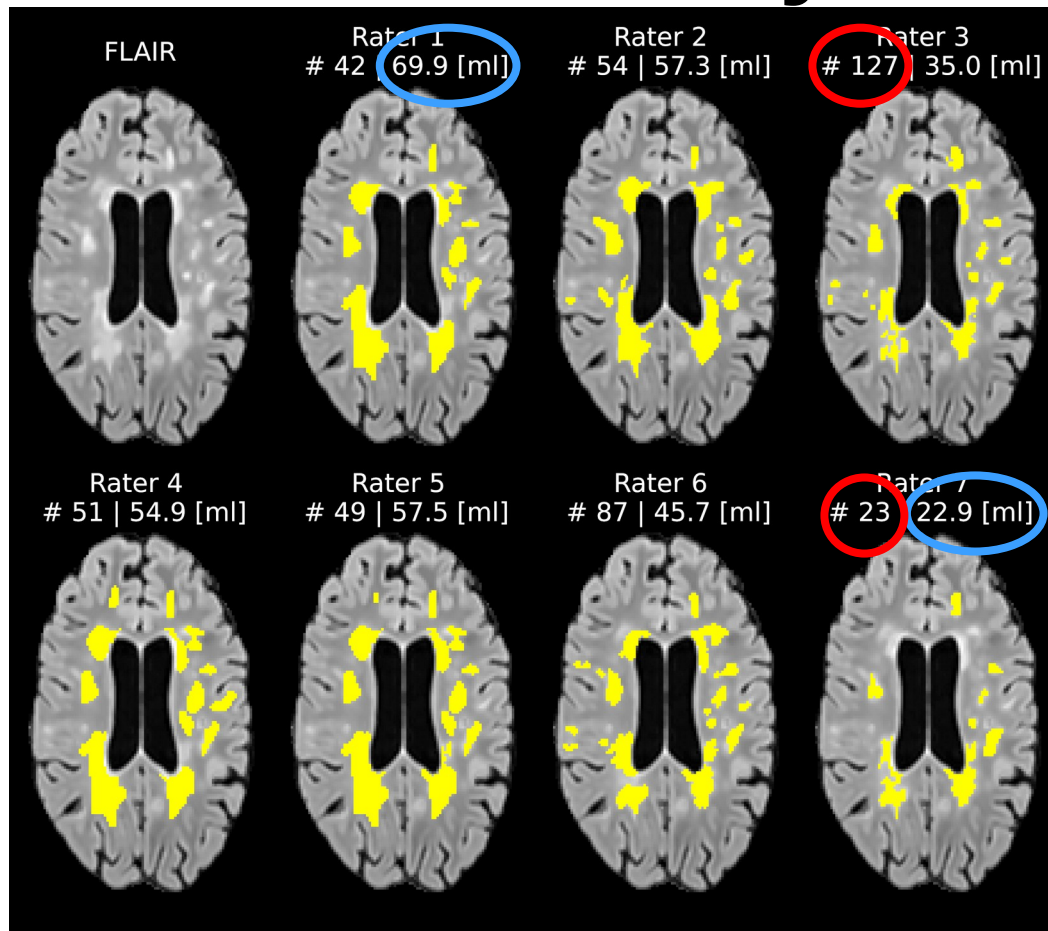


mosaic CT/MRI

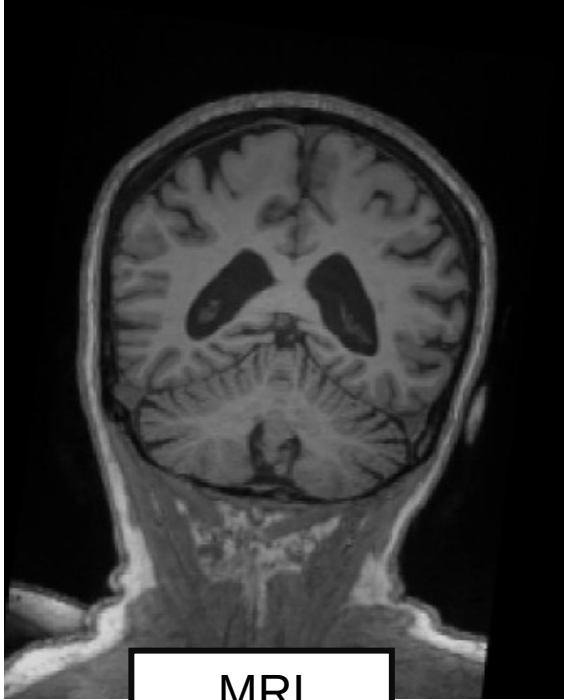
Measuring more consistently

Quantifying lesions in multiple sclerosis (MS):

- number (#)
- volume (ml)



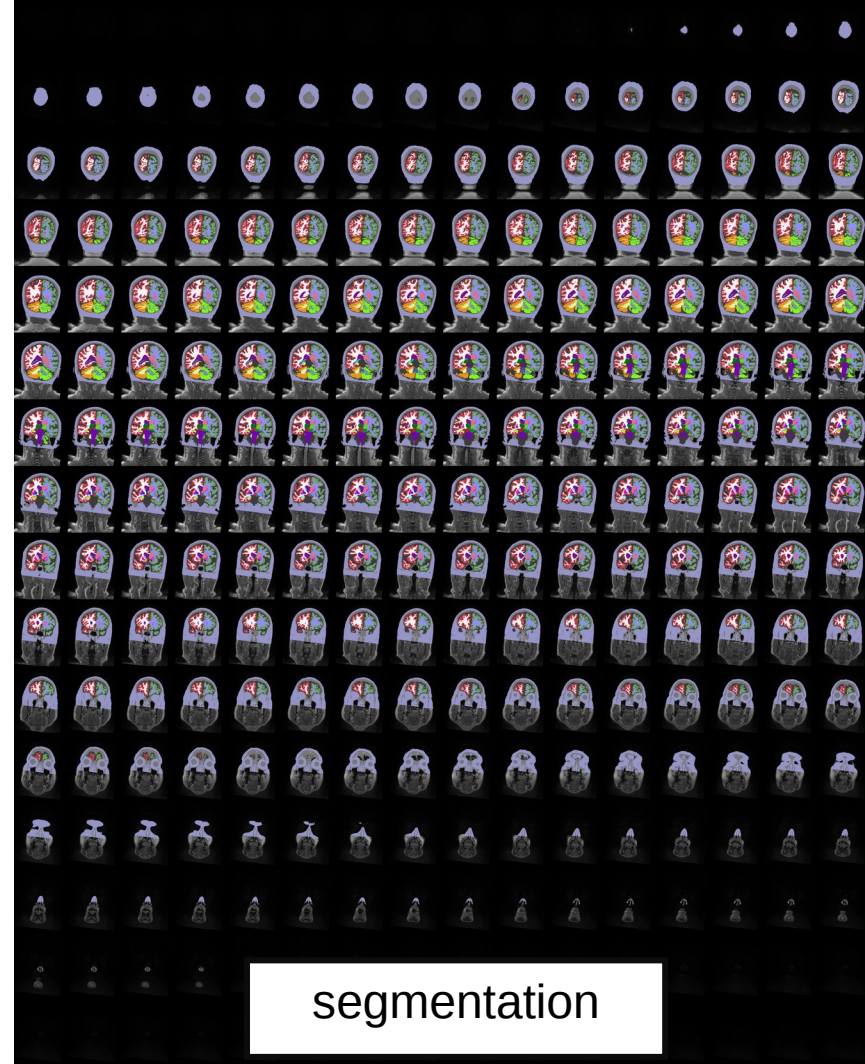
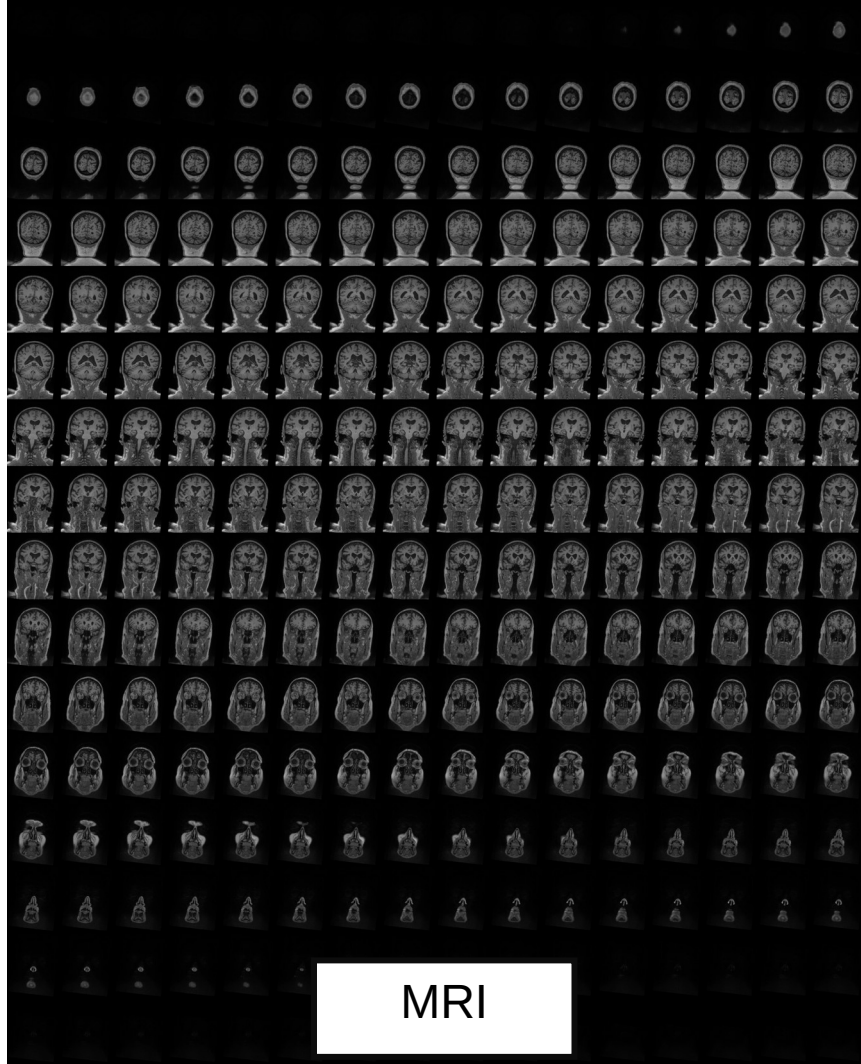
Analyzing images faster



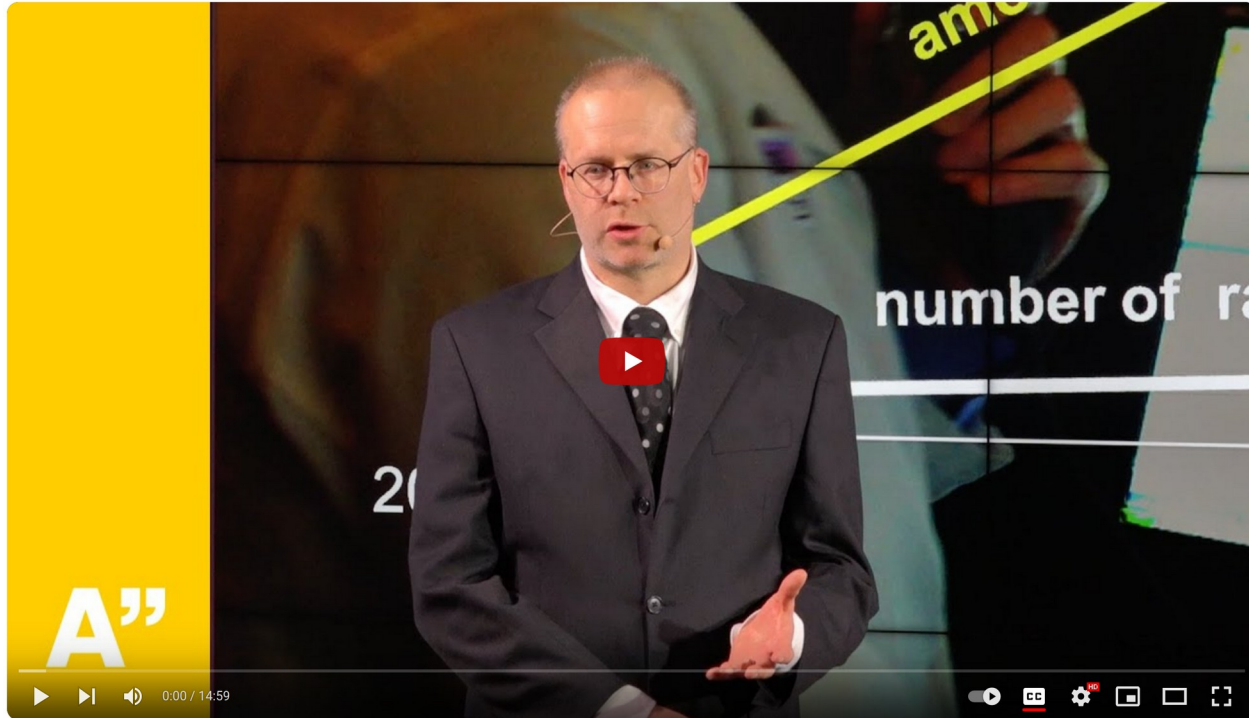
MRI



segmentation



Want to know more?



Artificial Intelligence in medical imaging: From research to clinical practice – Koen Van Leemput

Learning objectives

After this course you should be able to:

- ✓ **Implement** image smoothing and interpolation techniques
- ✓ **Use** spatial coordinate systems in medical images
- ✓ **Perform** landmark-based and intensity-based image registration
- ✓ **Select** the most appropriate similarity measure for specific image registration problems
- ✓ **Implement** rigid, affine and nonlinear spatial transformation models
- ✓ **Solve** segmentation problems using generative models
- ✓ **Perform** image segmentation using discriminative methods (neural nets)
- ✓ **Weigh** the advantages and limitations of generative vs. discriminative techniques in medical image analysis

A “do” course

Who are we?

- ✓ Koen Van Leemput, teacher (Professor, Neuroscience and Biomedical Engineering)
- ✓ Matias Vuokko, TA (Doctoral Researcher, Neuroscience and Biomedical Engineering)
- ✓ Amir Hassankhani, TA (Doctoral Researcher, Neuroscience and Biomedical Engineering)
- ✓ Alessandro De Florio, TA (Doctoral Researcher, Neuroscience and Biomedical Engineering)



Who are **you**?

- ✓ Master's Programme in Life Science Technologies 29
- ✓ Exchange studies (SCI + ELEC): 11
- ✓ Master's Programme in Computer, Communication and Information Sciences 6

- ✓ Bachelor's Programme in Electrical Engineering 5
- ✓ Bachelor's Programme in Science and Technology 3
- ✓ Aalto Bachelor's Programme in Science and Technology 3
- ✓ Bachelor's Programme in Chemical Engineering 2

- ✓ JOO Studies (Engineering, manufacturing and construction), SCI 2
- ✓ Master's Programme in ICT Innovation 2
- ✓ Master's Programme in Engineering Physics 2
- ✓ Master's Programme in Health Technology Engineering 1
- ✓ Aalto Doctoral Programme in Science 1

Who are **you**?

- ✓ Master's Programme in Life Science Technologies 29
- ✓ Exchange studies (SCI + ELEC): 11
- ✓ Master's Programme in Computer, Communication and Information Sciences 6

- ✓ Bachelor's Programme in Electrical Engineering 5
- ✓ Bachelor's Programme in Science and Technology 3
- ✓ Aalto Bachelor's Programme in Science and Technology 3
- ✓ Bachelor's Programme in Chemical Engineering 2

- ✓ JOO Studies (Engineering, manufacturing and construction), SCI 2
- ✓ Master's Programme in ICT Innovation 2
- ✓ Master's Programme in Engineering Physics 2
- ✓ Master's Programme in Health Technology Engineering 1
- ✓ Aalto Doctoral Programme in Science 1

Physical attendance:

- is expected in the lectures and the exercise sessions
- your primary way to ask questions and get feedback
- (it's possible to follow online, but not designed that way)

Teaching form

Lectures:

- ✓ Thursdays 12.15-14.00 (F239a)
- ✓ Lecture recordings are available
- ✓ One guest lecture (typically from industry)

Exercises:

- ✓ Thursdays 14.15-16.00 (Nanotalo 228)
- ✓ Python/NumPy + Jupyter notebooks
- ✓ Group-work (**2 students per group**)
- ✓ Five exercises in total, each split into an initial “easy” and a subsequent “difficult” part

go to MyCourses and select a group:

- otherwise you can't submit!
- exceptionally OK to have a 1-person “group” this week

simple grading by TAs

peer review by fellow students
+
feedback and final grading by TAs

Student participation:

- ✓ Students peer grade + present their fellow students' “difficult” exercise reports

Teaching form

Lectures:

- ✓ Thursdays 12.15-14.00 (F239a)
- ✓ Lecture recordings are available
- ✓ One guest lecture (typically from industry)

Exercises:

- ✓ Thursdays 14.15-16.00 (Nanotalo 228)
- ✓ Python/NumPy + Jupyter notebooks
- ✓ Group-work (**2 students per group**)
- ✓ Five exercises in total, each split into an initial “easy” and a subsequent “difficult” part

go to MyCourses and select a group:

- otherwise you can't submit!
- exceptionally OK to have a 1-person “group” this week

simple grading by TAs

peer review by fellow students
+
feedback and final grading by TAs

Student participation:

- ✓ Students peer grade + present their fellow students' “difficult” exercise reports

Logistics

<https://leempko.github.io/mia/>

- ✓ Course material (book, slides)
- ✓ Schedule

<https://mycourses.aalto.fi/>

- ✓ Group selections
- ✓ Exercises (input data, report submissions, peer grading)

Zulip course chat

- ✓ Link in MyCourses
- ✓ Important announcements
- ✓ Secondary place to ask questions

Week	Date	Activity	Location	Topic	
1	Thu 4 Sep	Lecture	F239a	Introduction; Image smoothing and interpolation	- chapter 1 in the book (html pdf) - slides: html pdf - lecture recording
	Thu 4 Sep	Exercise	Nanotalo 228	Smoothing and interpolation (first part)	submission deadline: Wed 10 Sep at 23:59
2	Thu 11 Sep	Lecture	F239a	Coordinate systems, linear spatial transformations, landmark-based registration	- sections 2.1-2.3 in the book (html pdf) - slides: html pdf - lecture recording
	Thu 11 Sep	Exercise	Nanotalo 228	Smoothing and interpolation (second part)	submission deadline: Wed 17 Sep at 23:59
3	Thu 18 Sep	Lecture	F239a	Intensity-based registration	- section 2.4 in the book, excluding Gauss-Newton optimization (html pdf) - slides: html pdf - lecture recording
	Thu 18 Sep	Exercise	Nanotalo 228	Linear registration (first part)	submission deadline: Wed 24 Sep at 23:59
4	Thu 25 Sep	Lecture	F239a	Student presentation: smoothing and interpolation exercise	
	Thu 25 Sep	Exercise	Nanotalo 228	Linear registration (second part)	submission deadline: Wed 1 Oct at 23:59
5	Thu 2 Oct	Lecture	F239a	Nonlinear registration	- sections 2.2.2 and 2.4 in the book, especially Gauss-Newton optimization (html pdf) - slides: html pdf - lecture recording
	Thu 2 Oct	Exercise	Nanotalo 228	Nonlinear registration (first part)	submission deadline: Wed 8 Oct at 23:59
6	Thu 9 Oct	Lecture	F239a	Student presentation: Linear registration exercise; Model-based segmentation I	- sections 3.1-3.3 in the book (html pdf) - probability refresher: html pdf - slides: html pdf - lecture recording

Python/NumPy, Jupyter Notebooks

- ✓ Python: <https://lectures.scientific-python.org/>
- ✓ Jupyter Notebooks:
<https://www.dataquest.io/blog/jupyter-notebook-tutorial/>

<https://jupyter.cs.aalto.fi/>

Jupyter exampleNotebook (autosaved)

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3 (ipykernel)

Image Smoothing and Interpolation

Linear regression

Let $\mathbf{x} = (x_1, \dots, x_D)^T$ denote the spatial position in a D -dimensional space. In medical imaging, D is typically 2 or 3. Given N measurements $\{t_n\}_{n=1}^N$ at locations $\{\mathbf{x}_n\}_{n=1}^N$, a frequent task is to predict the value t at a new location \mathbf{x} . A simple model, known as `lemph` (linear regression), uses the function value

$$y(\mathbf{x}; \mathbf{w}) = w_0 + w_1 x_1 + \dots + w_D x_D$$

as its prediction, where w_0, \dots, w_D are tunable weights that need to be estimated from the available measurements. A more general form uses nonlinear functions of the input locations instead:

$$y(\mathbf{x}; \mathbf{w}) = w_0 + \sum_{m=1}^{M-1} w_m \phi_m(\mathbf{x}),$$

which greatly increases the flexibility of the model. Here the functions $\phi_m(\mathbf{x})$ are known as `lemph` (basis functions), and it is often convenient to define an additional “dummy” basis function $\phi_0(\mathbf{x}) = 1$, so that the model can be written as

$$y(\mathbf{x}; \mathbf{w}) = \sum_{m=0}^{M-1} w_m \phi_m(\mathbf{x}),$$

where $\mathbf{w} = (w_0, \dots, w_{M-1})^T$ are M tunable parameters.

In order to find suitable values of the parameters of the model, the following energy can be minimized with respect to \mathbf{w} :

$$E(\mathbf{w}) = \sum_{n=1}^N \left(t_n - \sum_{m=0}^{M-1} w_m \phi_m(\mathbf{x}_n) \right)^2,$$

which simply sums of the squared distances between the measurements t_n and the model's predictions $y(\mathbf{x}_n; \mathbf{w})$.

```
In [4]: #
import numpy as np
from matplotlib import pyplot as plt
plt.ion()

#
N = 30;
ns = np.arange( N ).reshape( -1, 1 )
A = np.cos( np.pi * ( ns + 0.5 ) * np.arange( 3 ) / N )
A[ :, 0 ] *= 1/np.sqrt(2) # DC component is scaled differently
plt.figure()
plt.plot( ns, A )

Out[4]: [<matplotlib.lines.Line2D at 0x7fdf2fc61a60>,
<matplotlib.lines.Line2D at 0x7fdf2fc61a90>,
<matplotlib.lines.Line2D at 0x7fdf2fc6fc70>]
```

Copyrights

"Accurate and robust whole-head segmentation from magnetic resonance images for individualized head modeling" by Puonti et al. NeuroImage (2020), licensed under CC-BY-NC-ND