# Expert-Level Deep Learning for Computer Vision and Biomedicine

Practical Course

Winter Semester 2019/2020

Vladimir Golkov
Prof. Dr. Daniel Cremers

These slides will be available on the course website

# Focus, Learning Goals

- For excellent students
- Ongoing path to becoming an expert and making key contributions to deep learning
- Gain advanced practical experience in deep learning craftsmanship
  - Understand real open problems
  - Create methods, solutions, insights, systematizations, publications
    - Creating things is crucial for profound understanding of existing things
- All under guidance of experienced supervisors
- Gain profound insights
- Not limited to existing methods, rather developing novel methods

- Projects are geared towards producing scientific publications
- Topics include biomedicine, computer vision, etc.

- Practice presentation skills

# Prerequisites

- Excellent programming skills
  - Python
  - Array programming in NumPy (or Matlab or similar)
  - PyTorch (or TensorFlow or similar)
- Curiosity
- Solid skills in mathematics
- Good literature-reading/skimming skills
  - Theory and practice go hand in hand!
- Solid knowledge of deep learning theory
- Soft skills
- Time for regular hard work
- Proactivity
  - Project success depends on a two-way communication between the students and supervisors
  - If you expect to just passively receive detailed instructions and directions rather than also establishing communication and asking questions, then this practical course is not for you
  - If you just need some practical course for your curriculum and don’t plan to do excellent work, then this practical course is not for you
- Projects about biomedicine can be chosen, but prior knowledge in biomedicine is **not** required
  - You will learn from your supervisor

# Structure of Practical Course

- Three lectures in the beginning of the semester (Tuesday 2-4pm)
- Practical project
  - Each student gets assigned to one project (or a few very similar projects)
  - Each project consists of a “pool” of tasks
    - Requirements elicitation and agreeing upon solutions
  - Usually 1 or 2 students per task
  - Access to computers and GPUs in Garching and remotely
  - Deep learning requires early and regular efforts
  - Regular communication with supervisors (important for progress of learning and project success)
    - Depending on the project, there may be a weekly meeting/presentation discussing progress and challenges
    - Emailing skills are also important
- Final presentations
  - Groups can learn from each other and discuss
  - Presentation dates will be determined by voting (end of semester)

# Next Steps

- 19-24 July: Apply for a place at https://matching.in.tum.de/
- There are many applicants
- Sending info about yourself is crucial
- Email us info at the latest when you use the matching website, better several days earlier:
  - Your interests, learning goals
  - Description of your skills (see prerequisites)
  - Some code you wrote in any context
  - All grade transcripts
  - Ongoing courses
- If you require project info in advance, contact us
- If you want to propose own projects ideas, they should be discussed with us until 20 July
- Places in the course will be assigned on 30 July

# After 30 July

- Projects will be announced, discussed and assigned (based on your preferences) as soon as possible

# Most Imporantly

- Read project descriptions very carefully, ask as soon as possible whenever something is unclear, select projects wisely
- Follow all announced recommendations

# Other Options

- If you don’t get a place in the practical course:
  - Email us, enter the waiting list
  - Apply in subsequent semesters
- Whether you get a place or not, also consider applying for:
  - Bachelor Thesis
  - Master Thesis
  - Interdisciplinary Project
  - Guided Research
  - etc.

# Literature

- Christopher M. Bishop: “Pattern Recognition and Machine Learning”, Springer, 2006. (Skim the Chapters 1, 2, 5.)
- http://www.deeplearningbook.org/
- http://www.mlyearning.org/
- NumPy: Advanced Array Indexing https://docs.scipy.org/doc/numpy/reference/arrays.indexing.html