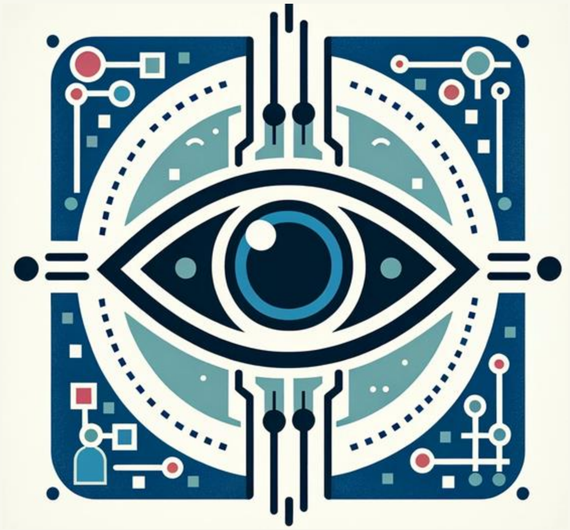


COMPUTER VISION

Project


C
D
E
F
G

TASK 1

- Input:
  - Image with chess board
- Output:
  - Total number of black/white pieces on the board
  - Position of the pieces on the image (bounding boxes)
  - Position of the pieces on the board (8x8 matrix with 0/1 values representing absence/presence of piece - any board orientation is acceptable)

TASK 1

Dataset: https://data.4tu.nl/datasets/99b5c721-280b-450b-b058-b2900b69a90f/2

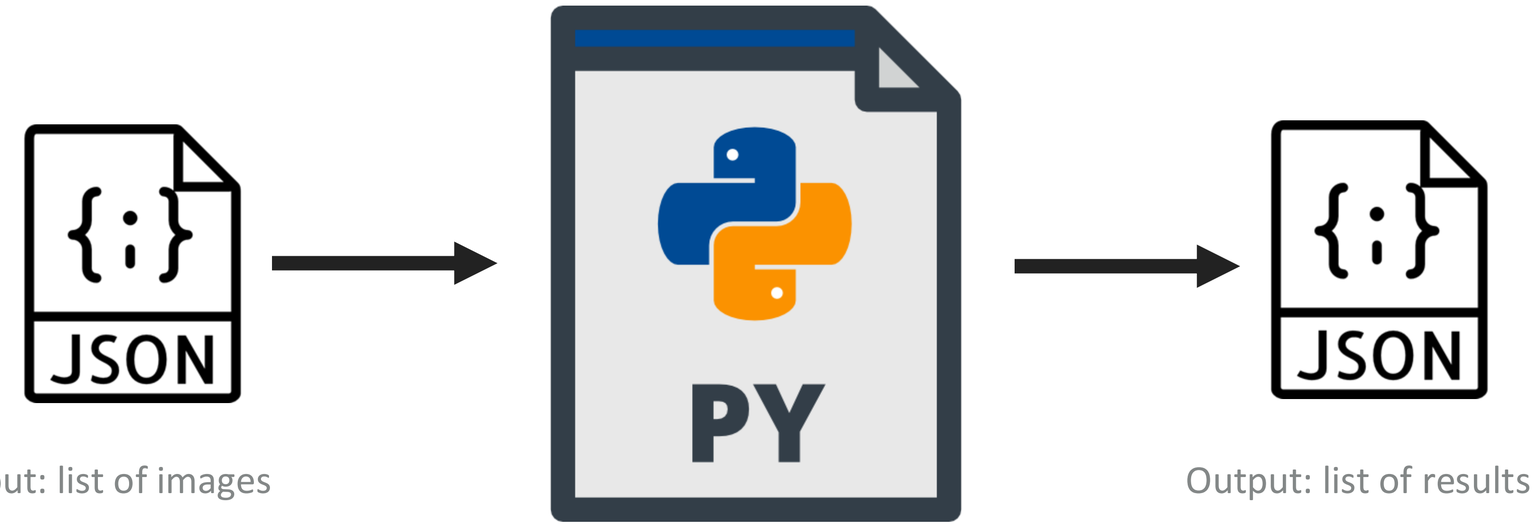
TASK 1

- Dataset:
  - 50 images randomly chosen from a public dataset
  - The results will be tested in 10 **undisclosed** images

- Deliverables:
  - Short report (2 pages max) presenting the **methodology** and some **results**
  - Python script (only one file)

- Deadline: **April, 14** (23:59 AoE)

TASK 1



Input: list of images

Image processing pipeline (*)

Output: list of results

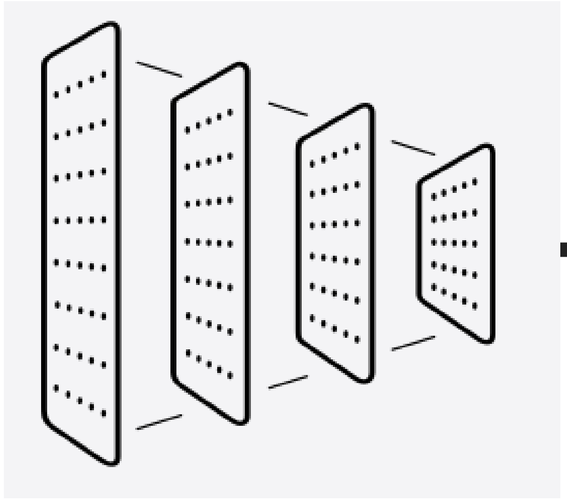
(*) using only OpenCV and other common libraries, like numpy and matplotlib

TASK 1

- Grading
  - Task 1 accounts for **30%** of the overall project grade
  - Elements being considered: methodology, report and quality of the results

- Important remarks
  - Follow **strictly** the JSON structure for the input and output files
  - It is **okay** to use AI tools while developing your work, but it is **not okay** to use them without acknowledging it
  - All members of the group **are expected to understand** the methodology and the submitted code

TASK 2



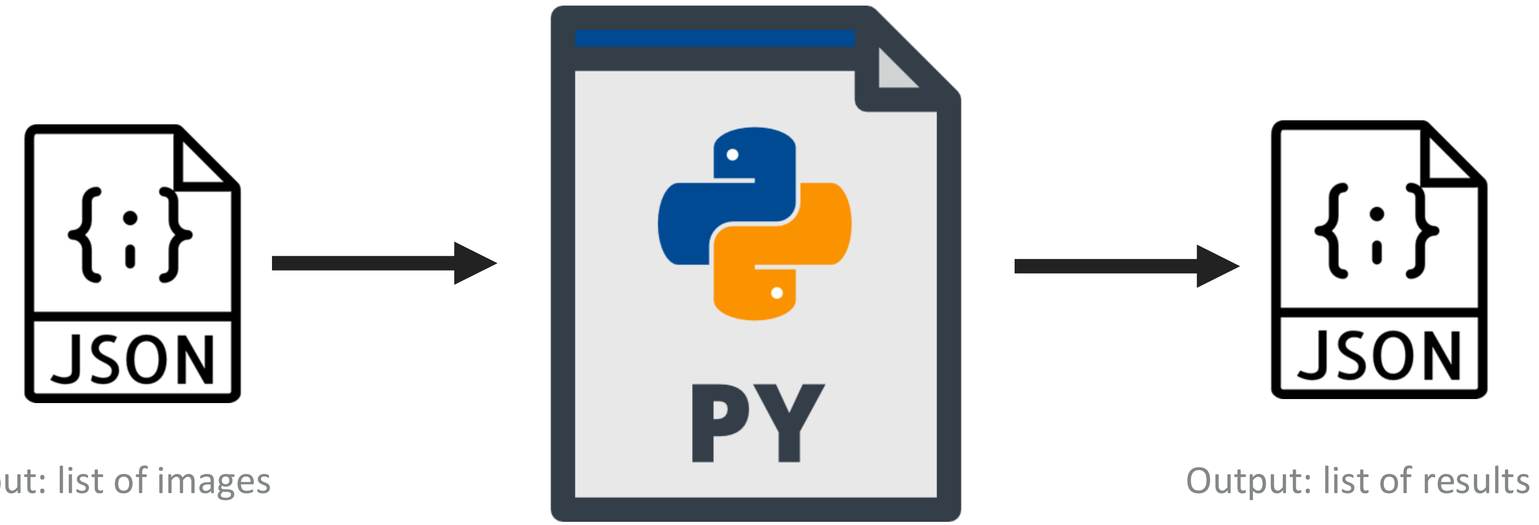
15

CNN image by Oleksandr Panasovskyi

TASK 2

- Input:
  - Image containing a game of chess
- Output:
  - Total number of pieces within the chess board
- Model(s):
  - CNN-based architecture
  - extra: quantitative comparison (with **adequate metrics**) different architectures

TASK 2



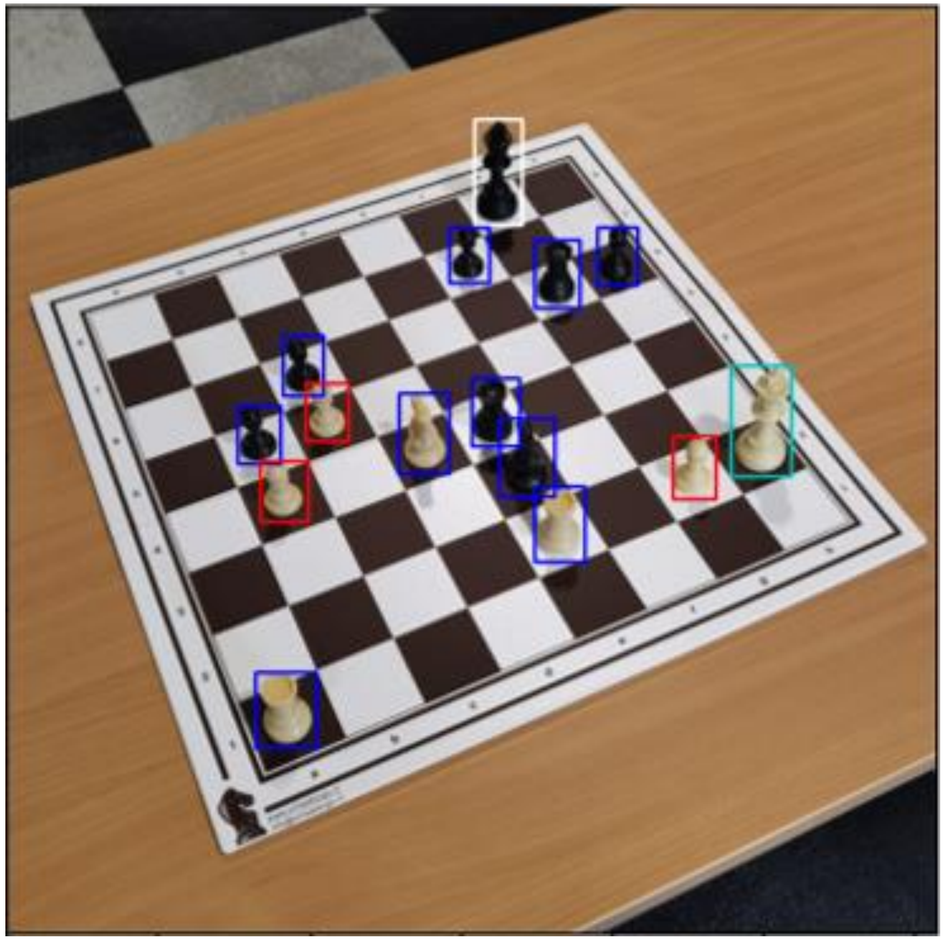
Input: list of images

Script of the deep learning-based model (*)

Output: list of results

(*) using only PyTorch, OpenCV and other common libraries, like numpy and matplotlib

TASK 3



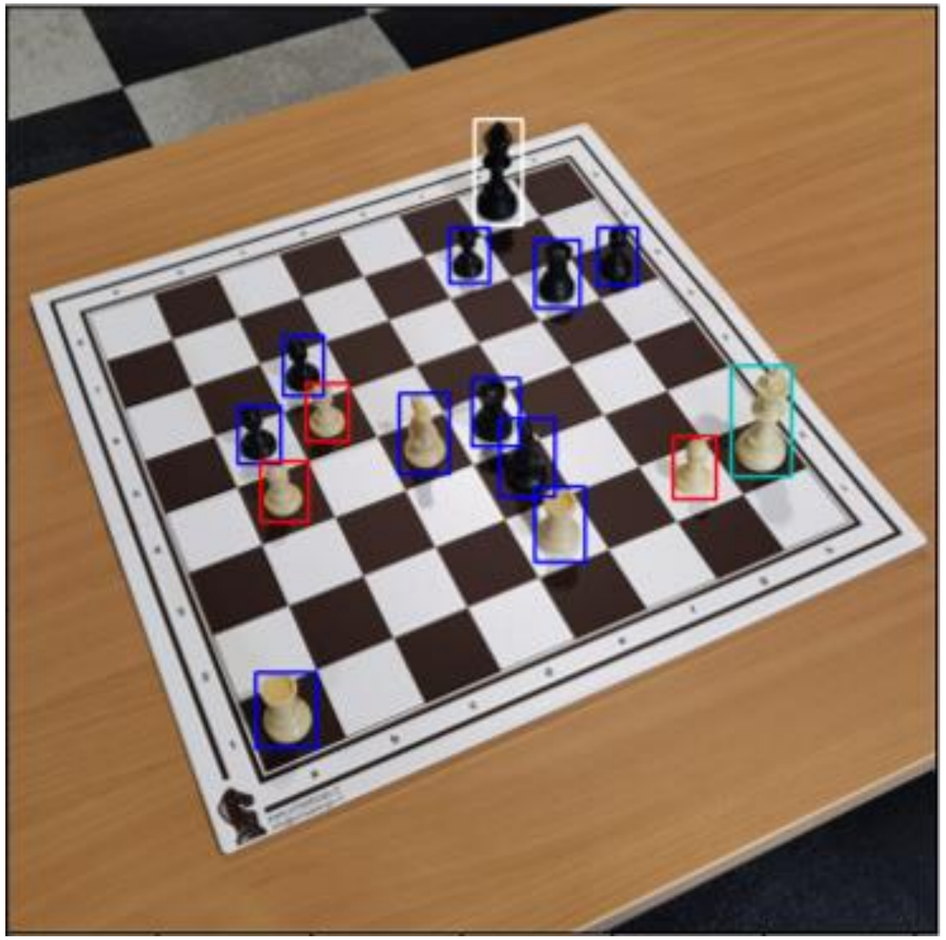
+

(example, the output does not have to be an image)

Chess Pieces Detection

Board *“Digital Twin”*

TASK 3



- Chess Pieces Detection:
  - At least one model, e.g. YOLO, Faster R-CNN
  - extra: quantitative comparison (with **adequate metrics**) of different architectures

- Board “Digital Twin”:
  - Identify the board status, i.e. where each piece (incl. colour and type) is in the board
  - One model, e.g. detection + traditional methods from task 1
  - Qualitative evaluation with some (good and bad) results are enough

TASKS 2 + 3

- Dataset:
  - https://doi.org/10.4121/99b5c721-280b-450b-b058-b2900b69a90f
  - Training/test splits available in annotations.json
  - Others (need to be documented)

- Deliverables:
  - Short report (3 pages max) presenting the **methodology** and some **results**
  - Python script (only one file) + model (.pth) for task 2
  - Notebook with some results for task 3

- Deadline: **June 13** (23:59 AoE)

- Intermediate Presentation: **May 29/30** (Thu/Fri classes)

TASKS 2 + 3

- Grading
  - Tasks 2+3 account for **60%** of the overall project grade
  - Elements being considered: methodology, report and quality of the results
  - The remaining 10% results from the intermediate presentation

- Important remarks
  - Follow **strictly** the JSON structure for the input and output files
  - It is **okay** to use third-party code and AI tools while developing your work, but it is **not okay** to use them without acknowledging it
  - All members of the group **are expected to understand** the methodology and the submitted code