UNIVERSITY OF BELGRADE
SCHOOL OF ELECTRICAL ENGINEERING
ETF Robotics

Action Recognition using 2D/3D Skeleton Sequences

Level: Bachelor/Master (1-2 students possible)
Duration: 3 months
Start: By agreement
Mentor: Milos Petrovic
Institution: ETF Robotics

Overview and Technology: This project addresses **action recognition** using **2D and 3D pose sequences** extracted from video. Instead of recognizing actions directly from pixels, the system uses a pose estimator to obtain a skeleton time series, reconstructs 3D joints, and then applies spatio-temporal deep models (GCNs, 3D CNNs, or transformers) to classify actions. Students will build an end-to-end pipeline (pose extraction -> skeleton normalization -> training/inference), evaluate on public datasets, and compare skeleton-based methods against RGB+D baselines in terms of accuracy, speed, and robustness to appearance changes.

Platforms / hardware

- PC Workstation (GPU recommended)
- Public video datasets
- RGB+D camera

Software & tools

- Python (NumPy, OpenCV)
- PyTorch + OpenMMLab (MMPose, MMAction2 / PySKL)
- PyTorchVideo (optional RGB baseline)

Project options (projects can be modified based on student interests)

- Extract 2D skeletons with MMPose and train MMAction2/PySKL skeleton models (ST-GCN, PoseC3D, etc.)
- Lift to 3D skeletons (VideoPose3D/MeTRAbs) and benchmark 2D vs 3D recognition
- Speed-up and deploy: real-time webcam inference, model compression/quantization, lightweight backbones

Expected outcomes

- Literature review
- Project code and documentation/video
- Final report in IEEE research paper form

Recommended background

- Digital image processing basics and camera geometry (2D/3D coordinates, normalization)
- Python programming (NumPy/OpenCV) and basic software engineering
- Basics of machine learning / deep learning (PyTorch)

Literature

- MMAction2 model zoo: mmaction2.readthedocs.io/en/latest/model_zoo/recognition.html and mmaction2.readthedocs.io/en/latest/model_zoo/skeleton.html
- MMPose for keypoint extraction; alternatives: MediaPipe Pose Landmarker and OpenPose
- ST-GCN paper; PySKL toolbox; PyTorchVideo (optional RGB baselines)
- OpenPose / MediaPipe for pose extraction