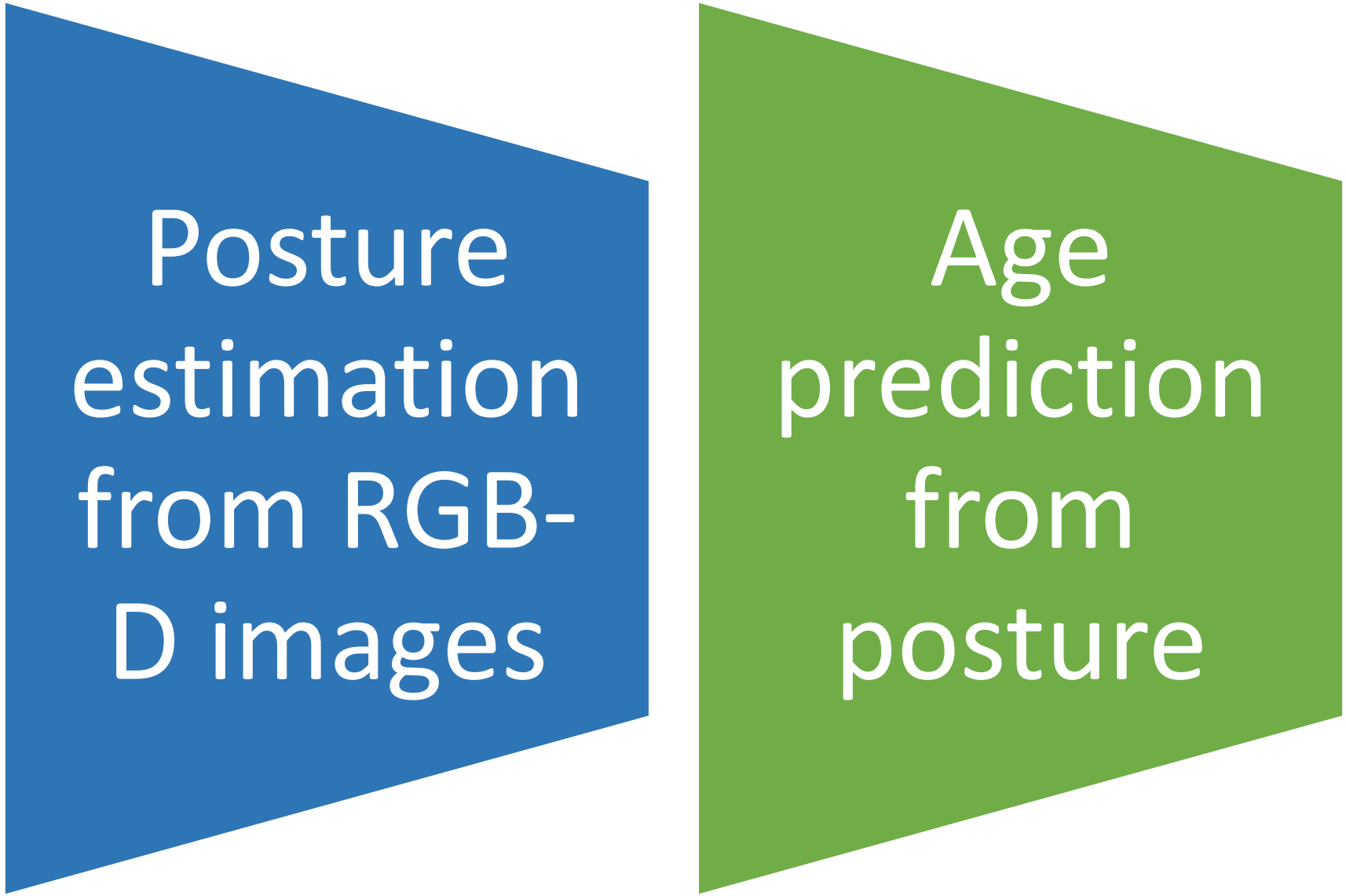
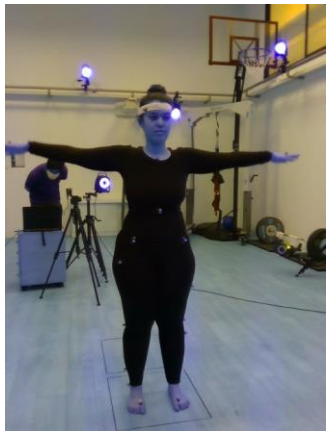
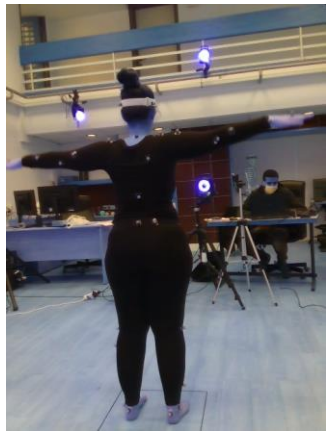
Age Prediction from Posture Estimation

Lotem Nadir, October 2021


Posture estimation from RGB-D images
Age prediction from posture

Dataset - Overview

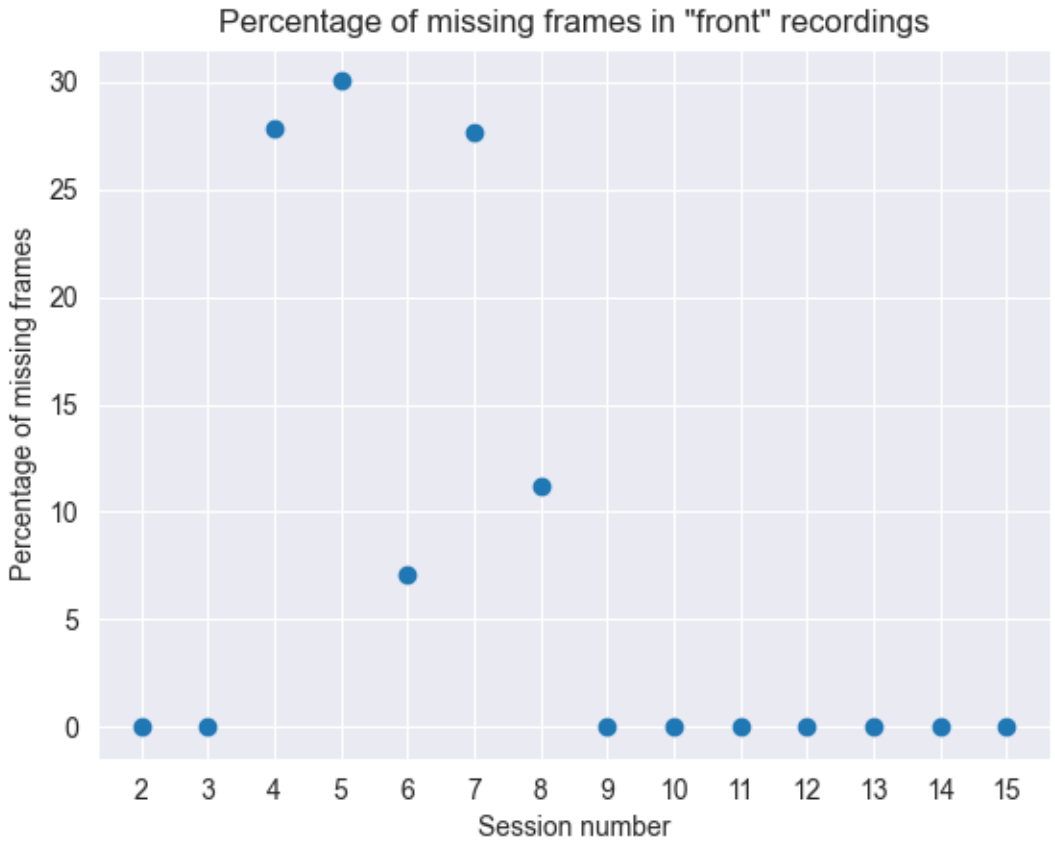
- 15 recording sessions.
- In each recording session, a person was recorded using 3 RealSense cameras (RGB-D) and Vicon sensors (3D coordinates).
- The RealSense cameras record 3 different angles: front, back and side.


Dataset - Challenges

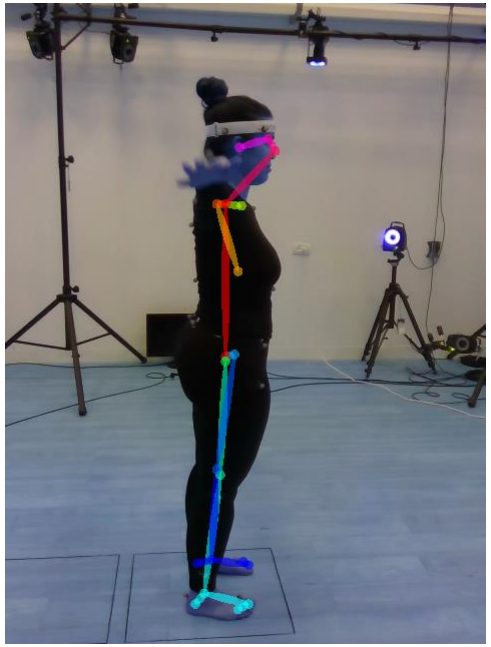
- The recordings are not synchronized in time by default, i.e. in the first frame in each recording the object is in a slightly different position.
- The FPS of the RealSense is 30, and the FPS of the Vicon is 120. In some recordings, the RealSense's FPS is 15.
- Since 2 of the RealSense cameras in each session are connected to the same laptop, there is a frame drop in the output of these cameras.
- There are no timestamps in the Vicon data.

Dataset – Challenges



Dataset – Main Progress

- Synchronizing the recordings:
  - Tried to use OpenPose for detecting the T-pose at the beginning of each recording. Bad results on the side angle, required manual fixes on the front and back angles.
  - Manually detected the T-pose in each recording.



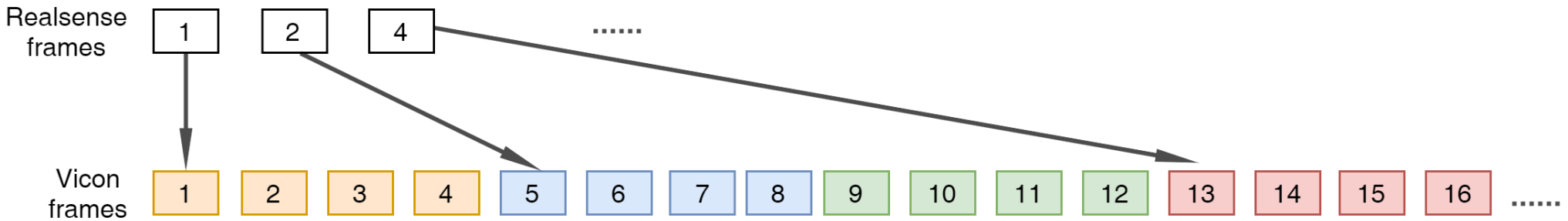
OpenPose output on “side” images

Dataset – Main Progress

- Different FPS:
  - I have decided to take every 4^{th} (or 8^{th}) frame from the Vicon data. I've also tried to average every 4 (or 8) frames. The measurement to check which method is better was to calculate an angle in the neck. There was an average difference of ~0.05 degrees per second between the 2 methods.

Dataset – Main Progress

- Frame drop in RealSense videos:
  - I have extracted for each RealSense recording the frames numbers from the bag file, and used the differences in the frames in order to "keep" only the corresponding frames from the Vicon data.



Dataset – Main Progress

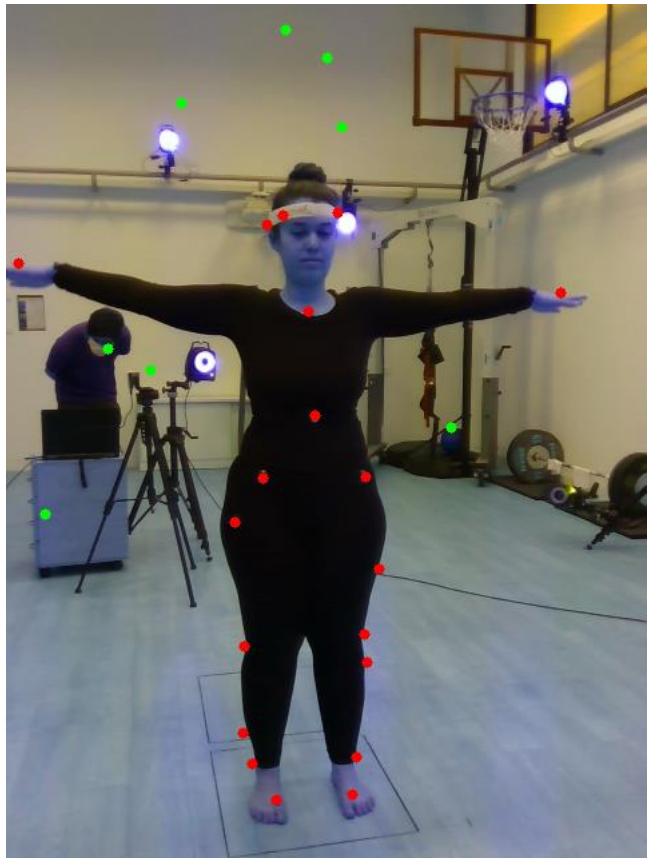
- Implemented trimming scripts.
- Validation of the synchronizing was done manually as well. Sometimes, due to the frame-drop, the T-pose was not detected correctly, and I had to pick another frame as first frame.

Projecting Vicon points into RealSense pixels

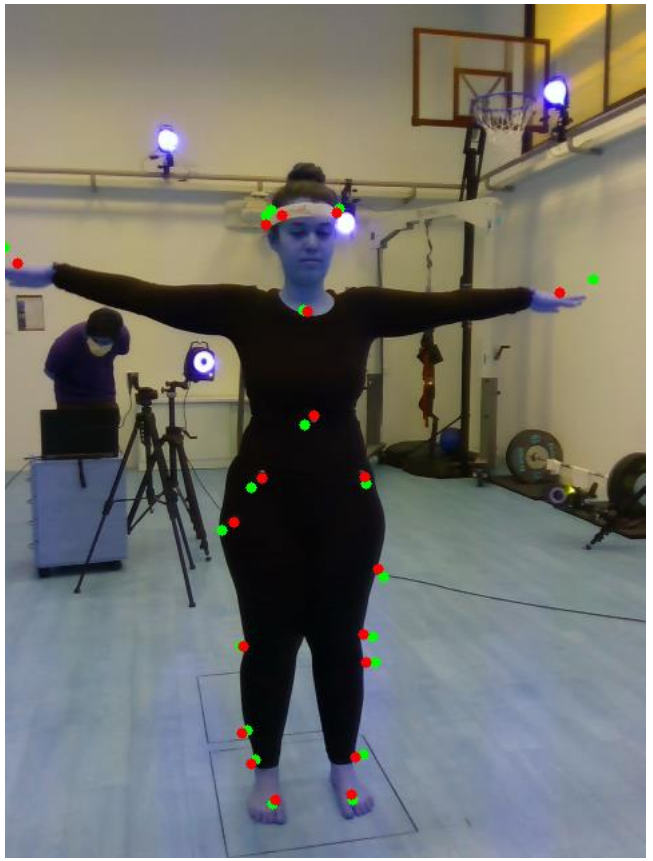
- OpenPose is a model for 2d pose estimation from RGB images.
- We had to project the Vicon points into the RealSense pixels.
- The RealSense cameras and the Vicon sensors are not calibrated.
- Kabsch algorithm is used to find the transformation between the Vicon system and the RealSense 3D system.
- Improvements to Kabsch:
  - Averaging the depth value (z) for each point with neighbor pixels.
  - Removing noisy depth values – For each point, if its depth (z) value is greater than some clipping distance, the point is removed.
  - Running Kabsch and finding error for each point, removing points with high error rates and re-calculating Kabsch.

  Current error after improvements is ~60mm.

Projecting Vicon points into RealSense pixels



without improvements



with improvements

Future work - Projecting Vicon points

- For previous recordings, try picking frames where the object is as static as possible. Maybe pick the ones after the T-pose.
- For future recordings, calculate the calibration matrix based on the calibration device was built for this task.

Age prediction from posture

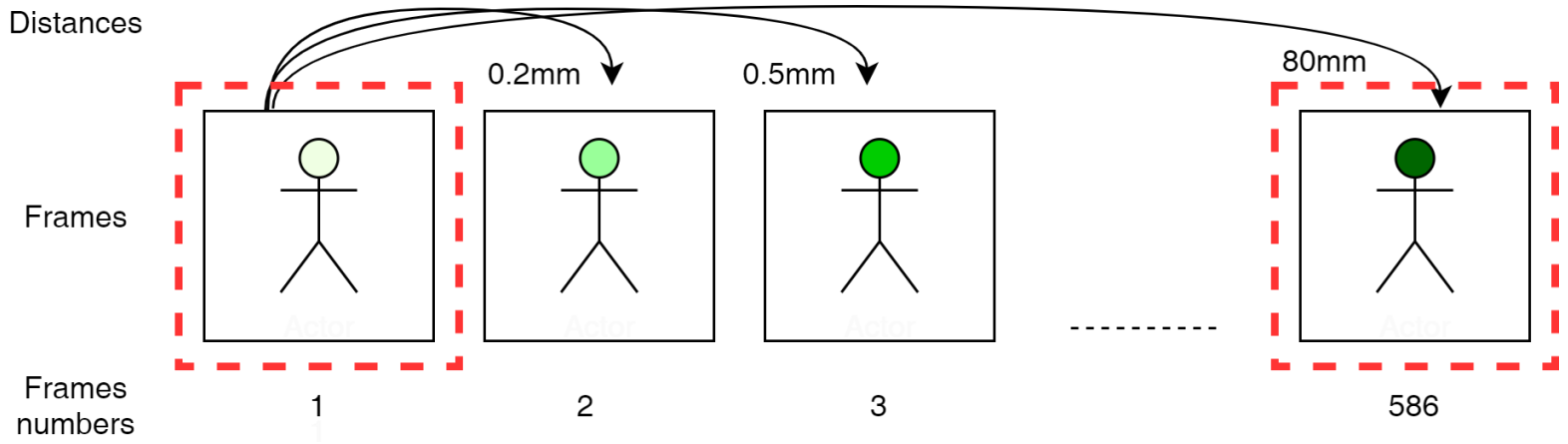
Dataset – Overview & Challenges

- ~250,000 samples, each sample has 39 3d points, and age as label.
- Dataset consists of only 23 different people.
- Many of the samples are “duplicates” due to the Vicon high FPS.
- Dataset is imbalanced: there are 7 ‘old’ people and 16 ‘young’ people.

Age prediction from posture

Dataset - Overview & Challenges

- In order to increase the variance in the data, I took only frames that their average Euclidean distance is >= 80mm.

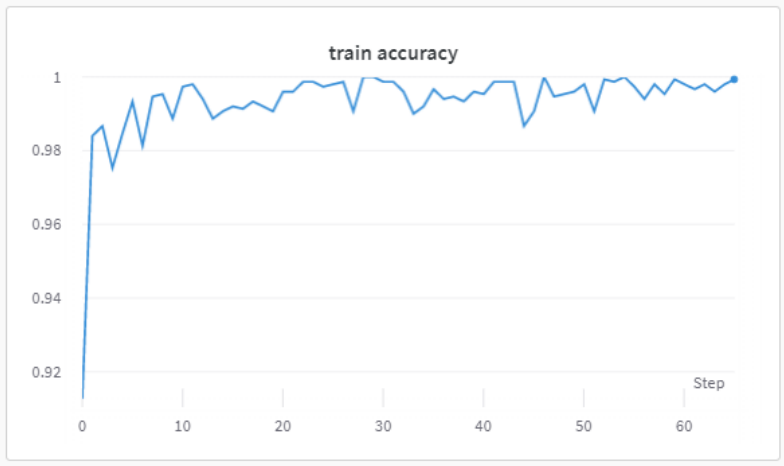


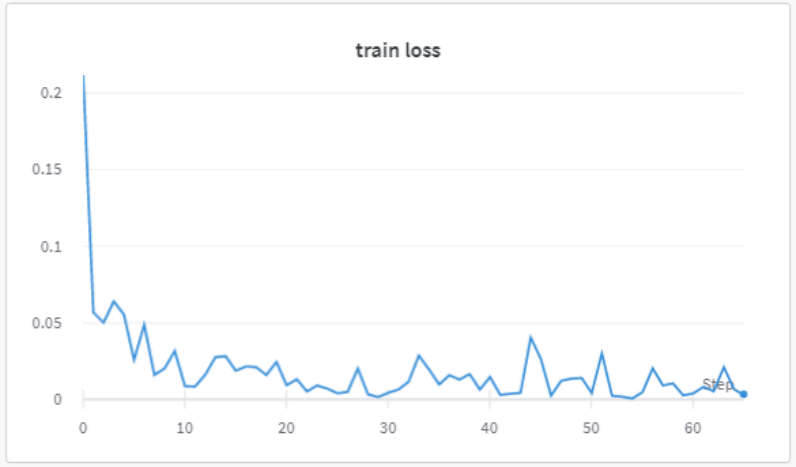
- After this process there were ~3000 samples were left.

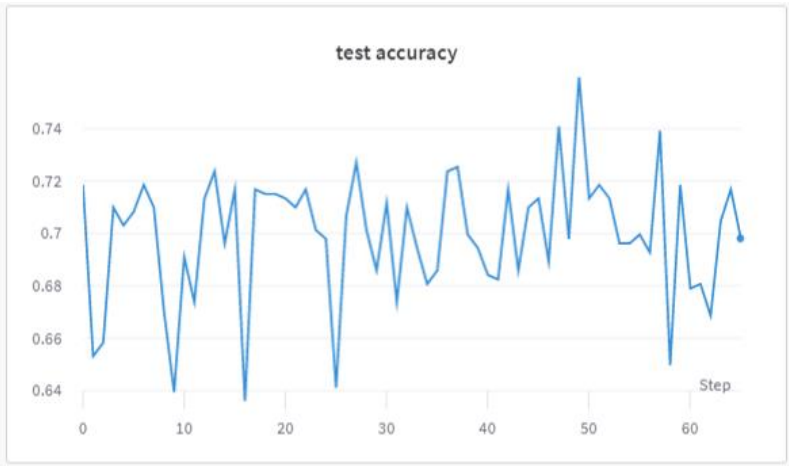
Age prediction from posture

Main Progress - Plan A

- Re-trained PointNet on our data, almost “as is” (except for the input layer and the softmax dimensions in the output layer).
- Resulted in high overfitting, the net learned to classify all samples as “young”.







Age prediction from posture

Future work - Plan A

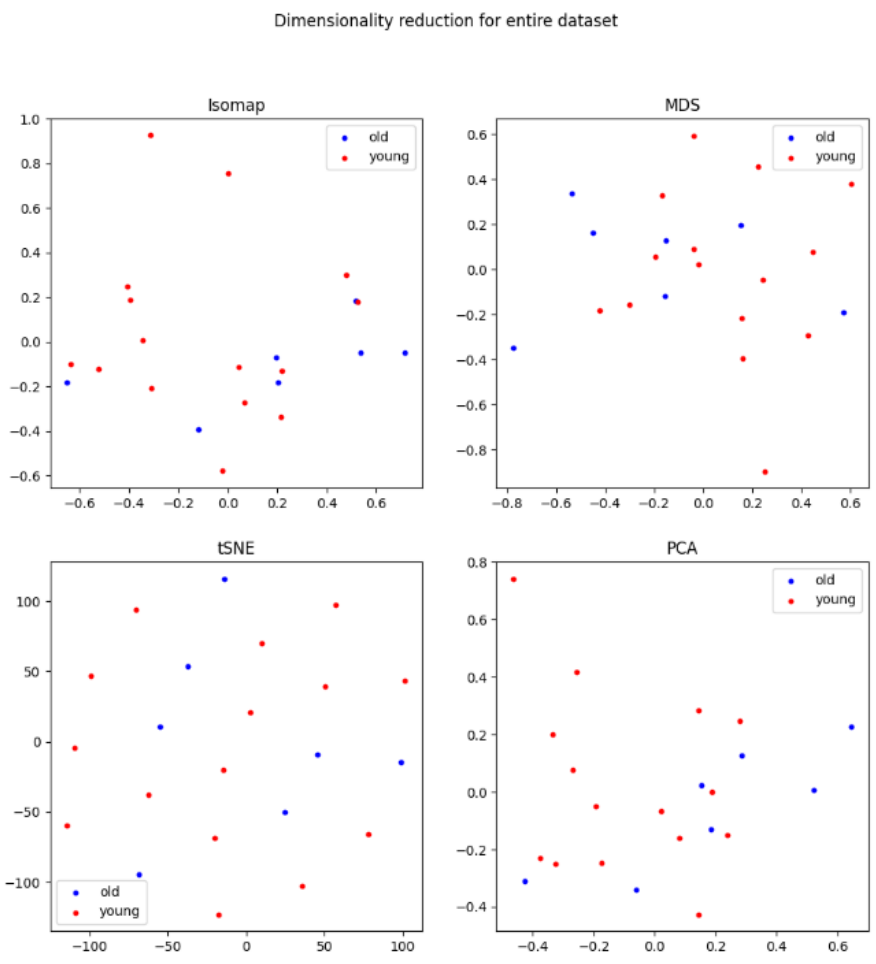
- In order to reduce the overfitting:
  - Reduce the dimensions of the embedding vector in PointNet from 1024 to 128 or 64, to make the model less complex.
  - Apply permutations on each sample. In my training the first row in each sample is always the same point, and that might affect the network's ability to generalize.
  - Data augmentations:
    - Add random noise of [-X, X] (in mm) to each point.
    - Rotate the point cloud along the Y axis.
    - "Construct" new old people, by taking old sample and switching only the points that are "hands" or "legs" with young sample.

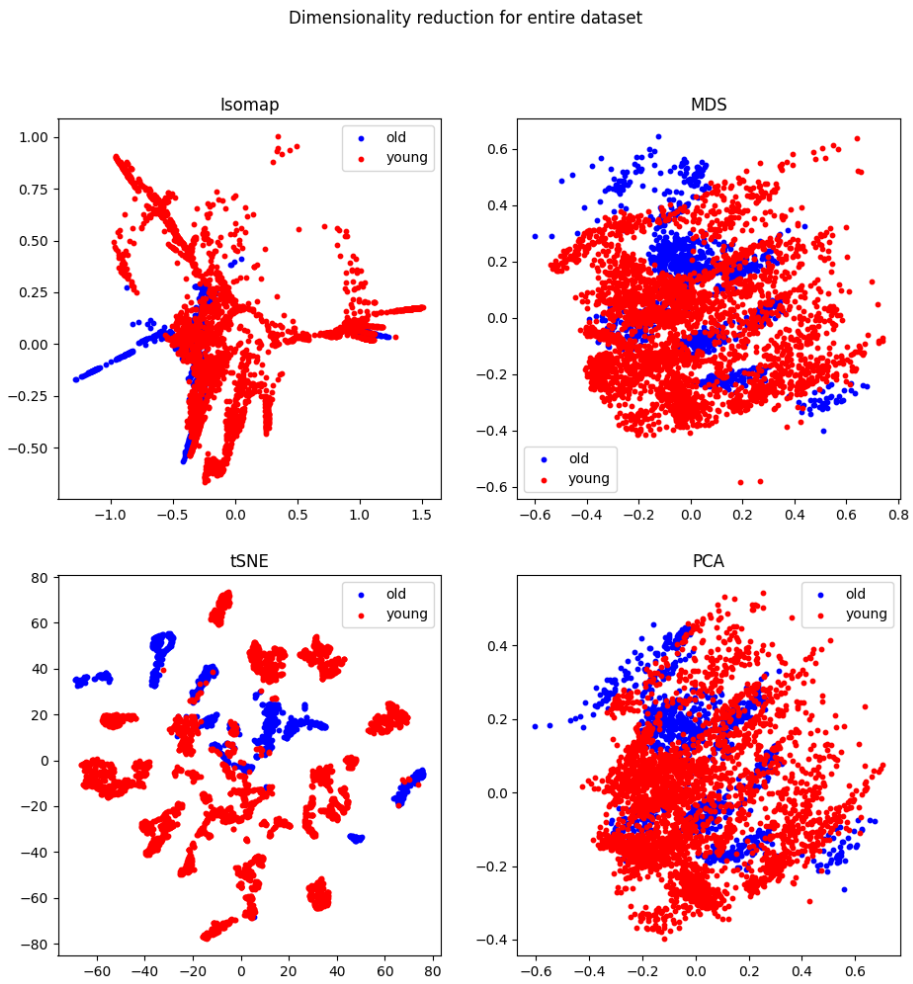
Age prediction from posture

Main Progress - Plan B

- For each sample, calculate 4 angles that has connection to the human posture. Classify the samples based on these 4 angles.
- Ran on the data several classifiers (KNN, SVM, Random forest), poor results on test set.
- Dimensionality reduction algorithms were applied on the data, to visualize it. Visualization show that there is no clear separation in the data.

Age prediction from posture

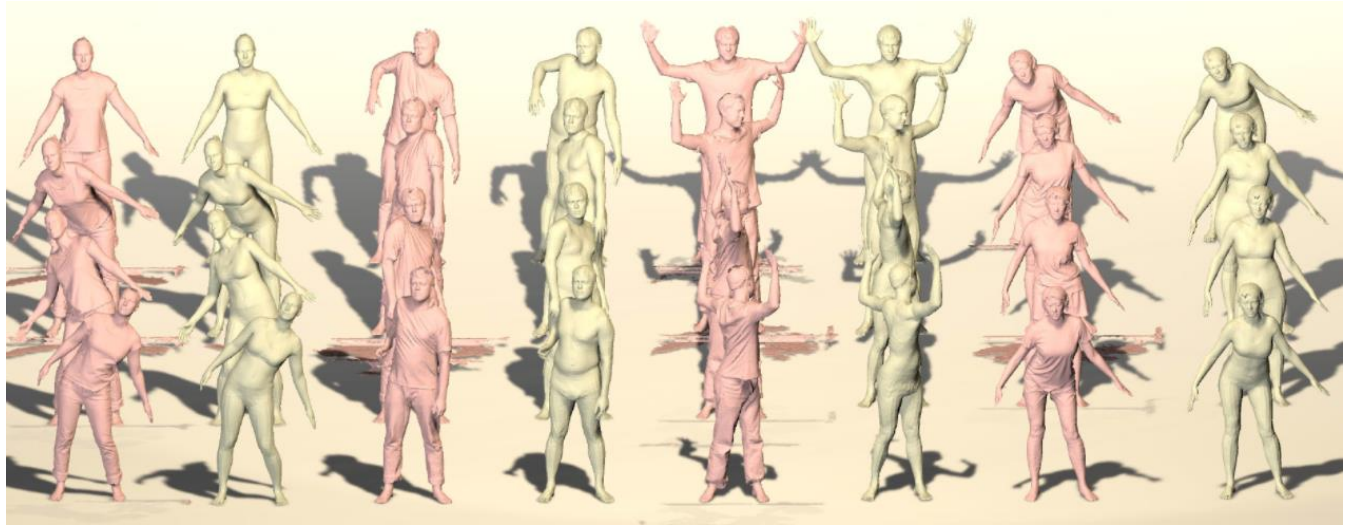




Age prediction from posture

Future work - Plan B

- Well, clearly – collect more data.
- Long shot, bear with me: perhaps 3D human body datasets can be used (?).



Thanks!