Ecole Polytechnique Fédérale de Lausanne

Evaluation of monocular depth estimation for human localization

10 credit semester project

Presented to :
Alexander Alahi

Supervisor :
Anastasia Bolotnikova
Author :
Victor Dramé

January 18, 2022

Contents

1 Introduction

This project is associated to the CIS Intelligent Assistive Robotics Collaboration Research Pillar. Along with other projects, the final goal is to enable mobile furniture control for assisting indoor mobility for people with limited mobility (e.g.: wheelchair, standing wheelchair, crutch). The general project is composed of several sub-projects that are addressing the challenges related to the vision for localization of the human and furniture, design for allowing furniture to move and motion planning to determine the furniture trajectory and motors inputs.

In this works, we will focus on the human localization, by this term we intend localization of a person with respect to the camera frame, predicting (x, y, z) information in meters. Within this project, we aim to study and evaluate the use of a low-cost monocular camera image based estimation of depth using Machine Learning (ML) techniques.

Human detection and pose estimation is a well known problem in computer vision and it finds many useful applications in robotics. However, for the depth perception, many approaches are either expensive (Lidar) or sensible to calibration (stereo). In this project we want to evaluate a learning based approach using monocular depth estimation and an off-the-self 2D pose estimation. We want to asses inclusiveness of the used ML tool for specific case of the pose estimation of people in a wheelchair or exoskeleton, prediction accuracy and time performance of the selected methodology. If all the performance aspects are satisfactory, the method could be used to enable the interactive control of mobile furniture to improve quality of life of people with mobility challenges and be used in other project where human localization is useful.

The code associated with this project is available here

2 Literature review

For this work, we used OpenPifPaf as an off the shelf tool for 2D pose estimation [2]. To pass from a 2D pose to a 3D pose, a monocular depth estimation is considered in this work and thus, we present the literature review to study existing works in this area and selected the most suitable method for our application.

2.1 Monocular depth estimation

To train a monocular depth estimation networks, various approaches exist that have different specifications, detailed in table 1

Methods	**Descriptions**	**Remarks**
supervised	Groud truth (GT) depth maps are used as the supervision signal of the deep learning network	High precision, simple framework, yet heavy dependence on GT
unsupervised	Using epipolar geometric constraints instead of GT as the supervision	GT is not required, but there are problems such as scale blur, dynamic blur, and occlusion
self-supervised	Relying on virtual data, sparse depth, surface normal and other auxiliary information	Heavy dependence on the auxiliary information

Table 1: A summary of the deep learning methods for monocular estimation [4]

In this work, we will mainly focus on supervised approach as they are the one reporting the highest accuracy. An issue with many existing models is that many are trained on datasets with static shapes (such as car and furniture) and do not achieve great result on dynamic shapes such as human. However, models trained on human datasets also exist and provided state-of-the-art results. We will discuss these approaches and models in the following subsections.

2.1.1 SMAP

SMAP [7] is a single-shot network which regresses several intermediate representations such as :

- Part Affinity Fields (PAFs) of body keypoint
- root depth map, prediction of the center of the skeleton keypoints

- part relative-depth maps, skeleton position with respect to their root
- optional RefineNet can be used to further refine the recovered 3D poses and complete occluded keypoints

With all of these representations, this method provides an estimation of the absolute depth estimation. In our project, we will proceed with a similar approach where we will use OpenPifPaf [2] as an off-the-shelf networks to estimate 2D body keypoints and either Midas [6] or the Mannequin [3] method to estimate depth maps from the RGB images.
The reason we did not choose to work with SMAP directly is that it reports an average Euclidean distance between the GT and the estimation around 80cm, which would not be usable for our project as the goal is to use the human localization as a feedback signal for robot control in close proximity to a person.

2.1.2 Midas

To train robust neural networks it is important to have large and diverse annotated dataset. However, obtaining such dataset is complicated for RGB-Detph (RGBD) data since it needs specific sensor or processing of the data. To palliate this issues, Rantlf and al. [6] developed tools that enable mixing multiple datasets during training, even if their annotations are incompatible. They proposed a training objective which is invariant to changes in depth range and scale and trained their networks on multiple datasets (DIW, ETH3D, Sintel, NYU, TUM). They reported close to state-of-the-art result on multiple dataset without fine tuning and made the trained network weights available to the community. We will uses two types of the available networks for the following reasons:

- *midas_v21_small* - as it is a lightweight networks made to run fast on most machines.
- *midas_dpt* - which is a new model using deformable patch-based transformer and claiming an improvement of the accuracy of 21% compared to the previous Midas works on their preprint [5].

2.1.3 Mannequin

Another work that was considered at first in our project as potentially useful tool is the work of Zhengqi Li and al. [3]. Their unique dataset is large and trained for case with human present in the scene. They take a data-driven approach and learn human depth priors from a new source of data: thousands of Internet videos of people imitating mannequins, i.e., freezing in diverse, natural poses, while a hand-held camera tours the scene. Because people are stationary, geometric constraints hold, thus training data can be generated using multi-view stereo reconstruction.
At inference, they claimed improvement over state-of-the-art monocular depth prediction methods. However, Midas achieve better performance even with their lightweight model so we did not continue to explore this method further.

2.1.4 Monolocco

In regards to 3D human localization, the MonoLoco [1] is also a very interesting method. They first predict the COCO skeleton using OpenPifPaf as an off-the-shelf network and then infer the depth from the height of the skeleton. However in our case, since we intend to work with multiple size profile (wheelchair, standing wheelchair), we cannot use MonoLoco as an off-the-shelf networks to infer depth. However similar approach will be tried later with selected body part.

2.1.5 OpenPifPaf

For the 2D keypoint estimation we use OpenPifPaf for the following reason :

- It is a bottom-up approach with real-time estimation
- The accuracy reported is sufficient for our use-case
- It works well with people with obstruct body part, which is frequent for people in wheelchair or exoskeleton
- Qualitative test validate the use of OpenPifPaf for people in wheelchair or exoskeleton

3 Methodology

In this work, we implemented a method similar to SMAP [7]. We compute the 2D keypoints with OpenPifPaf [2] and generate a depth map using the networks of the Midas method [6]. As this method returns relative depth map, we use known element to determine the scaling factor (detailed later in this document).
We then compute the metrics to evaluate the use of both models. From this, we implemented the output in real-time streaming as a ROS2 package.

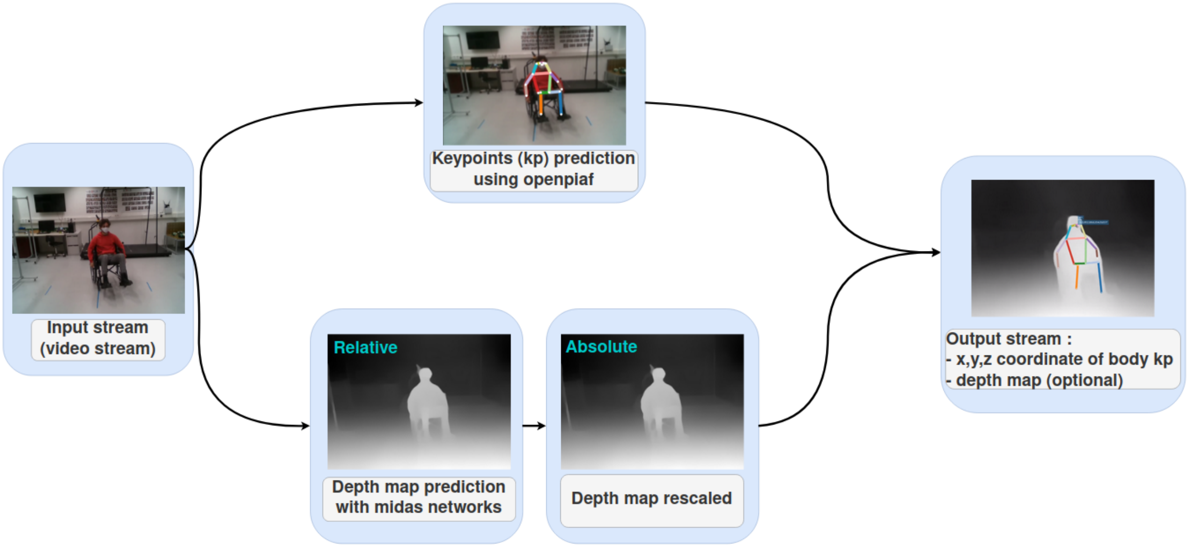

Keypoints (kp) prediction using openpiaf
Input stream (video stream)
Output stream :
- x,y,z coordinate of body kp
- depth map (optional)
Relative
Depth map prediction with midas networks
Absolute
Depth map rescaled

Figure 1: Schematic of the methodology for the project

3.1 Dataset

To evaluate metrics, GT is obviously needed. As there is no RGB-D dataset of people in wheelchair or exoskeleton in our knowledge. We made acquisition of such data in the BioRob laboratory, the acquisitions were made with an Intel depth D435i camera which has the following specifications:

- Use environment: Indoor/Outdoor
- Ideal range: .3 m to 3 m
- Depth Accuracy: <2% at 2 meters
- Depth technology: Stereoscopic
- Full spec : `https://www.intelrealsense.com/depth-camera-d435i/`

Most of our data acquisitions were be done in the range from 1m to 4m which may include bias in the evaluation. But since the project is made for indoor estimation, we assume that it is sufficient.

3.2 Metrics

For the 2D keypoints estimation, only a qualitative metrics estimation was performed as the OpenPifPaf returned similar output on people in wheelchair or exoskeleton as on people with occluded body parts, since the method was trained over crowed spaces with many samples of occluded body parts in the training set. The quantitative metrics are assessed only on the keypoint depth of observed people. The assessment metrics are the Root Mean Square Error (RMSE).

3.3 Rescaling

Midas and the Mannequin networks both return relative depth estimation where the distance estimated has no metrics and only the relative depth between the different pixel is important. Since we want the depth expressed in meters, we need to go from relative depth map to an absolute one.

Absolute depth can be returned from the relative one if we know the depth of at least two pixels in the image. We follow an optimization approach to estimate absolute depth with a method proposed by Rantlf and Al. [6]. We align the prediction to the GT based on a Least Square (LS) criterion:

$$\begin{aligned} (s,t) = \ & arg \min_{h} \sum_{i=1}^{M} (sd_i + t - d_i^*)^2, \\ & \hat{d} = sd + t, \hat{d^*} = d^* \end{aligned} \tag{1}$$

where $\hat{d}$ and $\hat{d}$ are the aligned prediction and the GT, s and t are respectively the scale and translation factor, which can be efficiently determined in closed form by rewriting 1 as a standard least-squares problem: Let $\vec{d_i} = (d_i, 1)^T$ and $h = (s,t)^T$, then we rewrite the optimization objective as:

$$h^{opt} = \ arg \min_{s,t} \sum_{i=1}^{M} (\vec{d_i}^T h - d_i^*)^2 \tag{2}$$

which leads to the close form solution:

$$h^{opt} = (\sum_{i=1}^{M} \vec{d_i}\vec{d_i}^T)^{-1} \cdot (\sum_{i=1}^{M} \vec{d_i} d_i^*) \tag{3}$$

To do the scaling we, therefore, need at least two keypoints that we consider as GT. To obtain these keypoints, we experiment with the two following approaches:

1 We use some portion of the data provided by the depth camera as GT. In the real application this GT could be estimated from known markers in the environment.

2 We use the size of a human body parts to infer the depth. Since we want to ensure inclusiveness, we will consider keypoints that are usually well visible among people in a wheelchair or exoskeleton. Therefore, we chose the distance between shoulder-hips.
To obtained a relationship between the pixel size and the depth, we compute the depth using the Intel camera over 10 videos of 300 frames and fitted a second order polynomial equation 4.

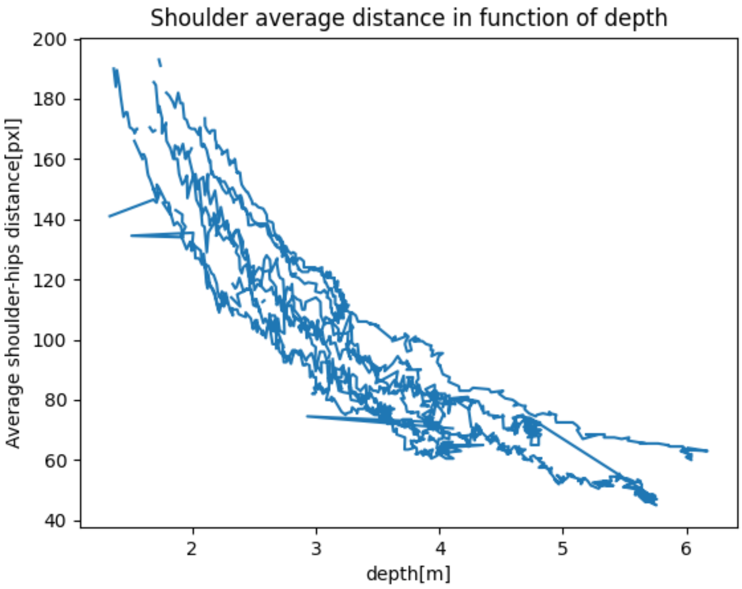


Figure 2: Depth from RGBD camera in function of torso pixel size

From the figure 2 we fitted the following second order equation

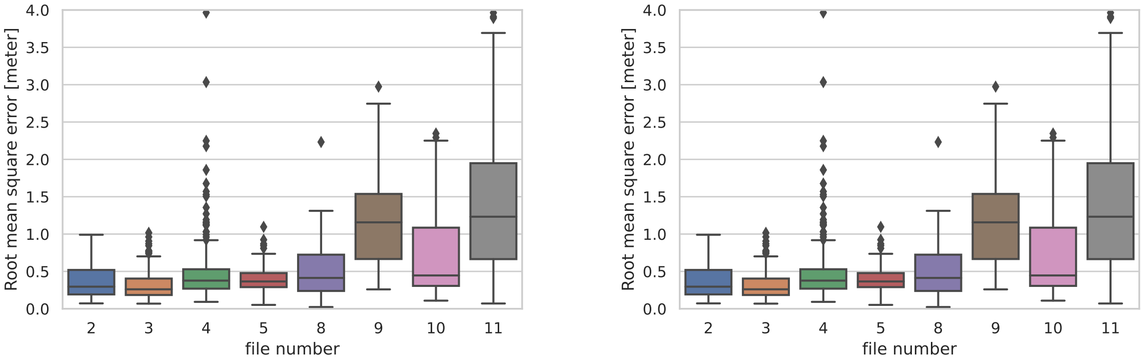
$$D = 8.01904 - 0.073300L + 0.000213L^2 \tag{4}$$

with D the depth in meter and L the the average length between shoulder and hips in pixels.

4 Results

4.1 Accuracy with depth obtained from GT

As mentioned before, the metrics are computed over the COCO keypoints. Each estimated depth frame from Midas method is rescaled using the scaling parameters computed using a small part of the depth input image where there is no subject as described in the previous section. Each file is a video of three hundred frames over 10 seconds were there is someone in a wheelchair moving around. We obtained the following metrics:

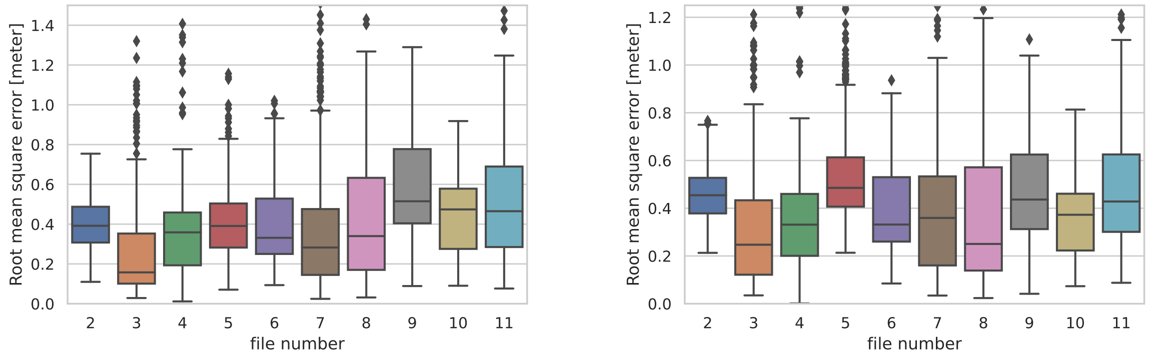


(a) ‘Midas small version 2.1’

(b) ‘Midas V2.1’

Figure 3: RMSE on ‘Midas version 2.1’

From the results presented in figure 3, we see that the Midas method achieves RMSE in between 0.3m to 1m, even though we can clearly see an outlier in the 11th file.



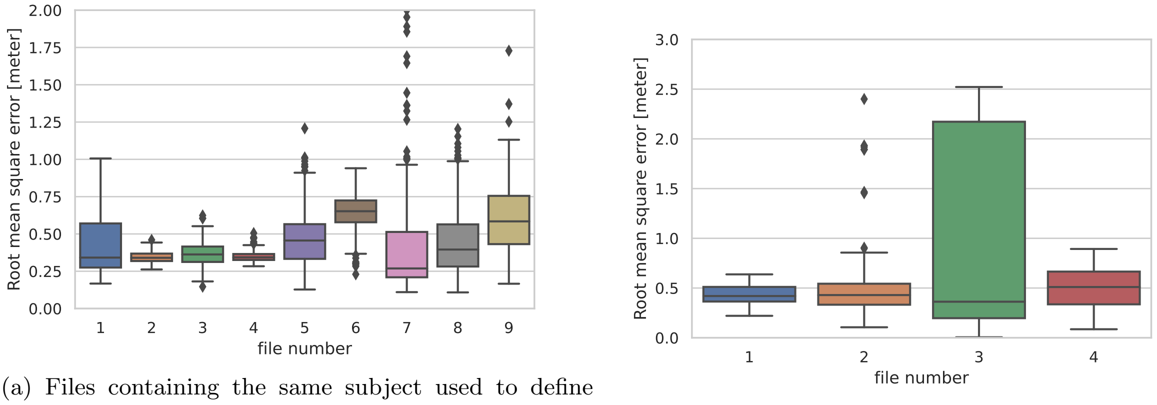
(a) Scaled over portion of image

(b) Scaled over whole image

Figure 4: RMSE on the preprint of DPT Midas (new model using transformer)

We see a clear improvement with Deformable Patch-based Transformer (DPT) in the model architecture (figure 4a) compared to without (figure 3). Yet, the average RMSE is in the range between 0.2m to 1.6m, which is too high to be efficiently and safely used as a signal for robot control in close proximity to the person.

4.2 Accuracy with depth obtained from skeleton size



(a) Files containing the same subject used to define equation 4

(b) Files containing another subject

Figure 5: RMSE when scaling Midas small with torso size

We observe better result when scaling with the torso size compared to the scaling with GT pixels, however, it should be noted that this is highly dependent on the fitting done in equation 4. And so highly relying on the initial measurement, which explains why there is a bigger error on the RMSE of the other human in figure 5. Another point is that when there is flickering on the prediction of the torso, high error seems to be measured as we can see in the file 3 of figure 5b.

4.3 Timing

To ensure real time inference, it is important to compare the time necessary to evaluate each model (OpenPifPaf and Midas). To observe the inference time, we measure it on the lab computer which has a single core *NVIDIA GeForce GT 1030*.

	mobilenetv2	resnet50	shufflenetv2k16	shufflenetv2k30
Average inference time [ms]	118	427	257	572
Average inference frequency [Hz]	8.5	2.3	3.9	1.7

Table 2: Inference time for the OpenPifPaf networks on the BioRob lab computer

	midas_v21_small	midas_v21	mannequin	DPT-Hybrid
Average inference time [ms]	61	1267	996	933
Average inference frequency [Hz]	16.5	0.8	1	1.0720
Inference time reported in the Midas paper [ms] *	-	32	-	38

* *Timings were conducted on an Intel Xeon Platinum 8280 CPU @ 2.70GHz with 8 physical cores and an Nvidia RTX 2080 GPU.*

Table 3: Inference time for depth estimation on the BioRob lab computer

As we can see from Table 2 and 3, to achieve fastest computation time on the lab computer best is to use the *Midas21small* and *mobilenetv2* networks. However, if we obtain more powerful computer capacity, we could use bigger networks. The study of different encoders done by [6] shown in figure 6 suggests that the encoders performances are directly related to networks performances.

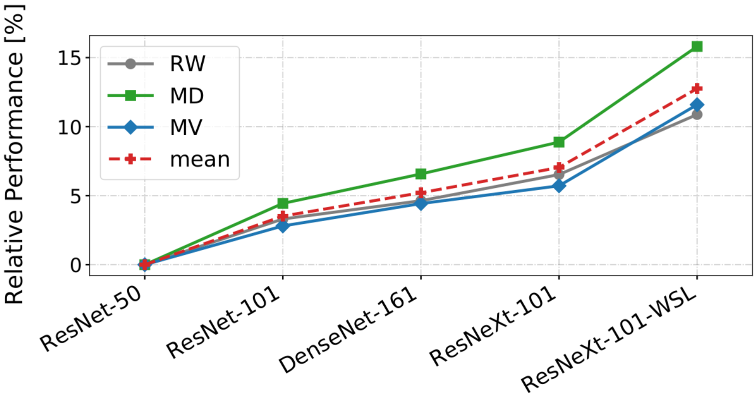


Figure 6: Relative performance of different encoders across datasets (higher is better). ImageNet performance of an encoder is predictive of its performance in monocular depth estimation.

4.4 RGBD camera data based approach

Since we want to use the method of this project to enable robot control, the accuracy obtained for both scaling methods (figure 5 and 4a) is not sufficient. Such a high inaccuracy will likely force the controller to have large uncertainty which is not safe for human-robot interaction indoor applications. To provide a more accurate solution, finally we have concluded that the best method available for depth perception at the moment is to use the Intel D435i camera as the depth map source and use the OpenPifPaf to estimate the human body keypoints (x, y, z) coordinates.

What's available to us now is the (u, v) keypoints coordinates in pixels from the OpenPifPaf and the corresponding z coordinate in meters from the RGBD camera. Our first steps is to pass from this coordinate frame to (x, y, z) in meters. For this we use the following projection matrix:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = z \begin{bmatrix} \frac{1}{f_x} & -\frac{s}{(f_x f_y)} & \frac{s c_y - c_x f_x}{f_x f_y} \\ 0 & \frac{1}{f_y} & -\frac{c_y}{f_y} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \tag{5}$$

with f_x, f_y the the focal lengths, c_x, c_y the principal points and s the skew. Every value is in pixels length, for our camera we have $[f_x, f_y] = [909.2, 909.0]$, $[c_x, c_y] = [657.7, 357.6]$, $s = 0$.

Now that we have our human body keypoints coordinates (x, y, z), we compute the rotation matrix from camera frame to the torso frame. To do this we will use rigid transformation to find the rotation and translation matrices. However in a rigid transformation approach, we need the coordinates in the new referential. To compute them we will take the torso keypoints (x, y, z) coordinates and project them on their principals axis. One issue with this approach is that even though the axis are correct it does not differentiate the direction and can lead to an axis projection in flickering direction. Once this is done, we compute the rotation matrix R and translation vector T using singular value decomposition.

4.4.1 Visualization

To have a better understanding of what is the result of our computations, a pointcloud and skeleton visualisation has been added:

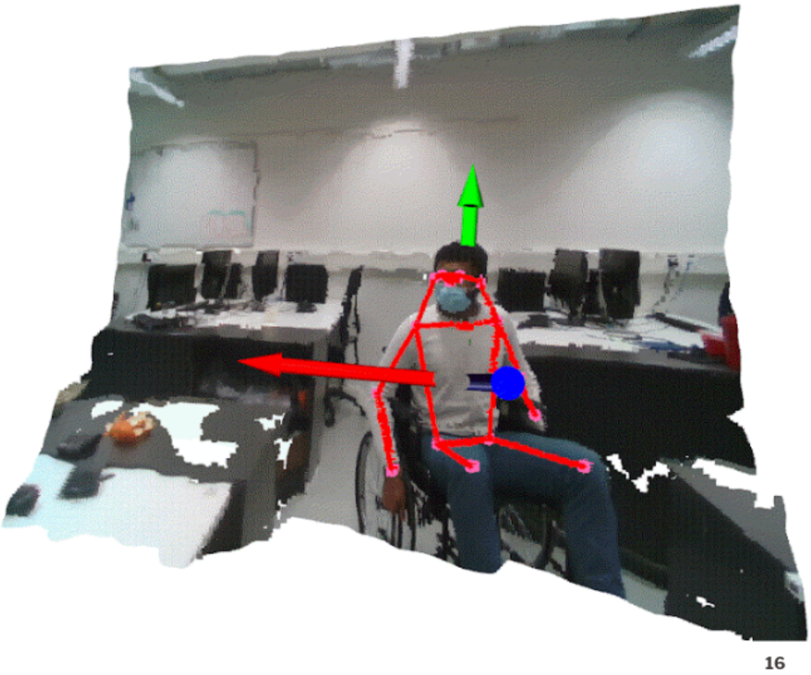


Figure 7: Pointcloud in Cartesian frame with GT from D435i camera, torso frame axis are represented with the arrows

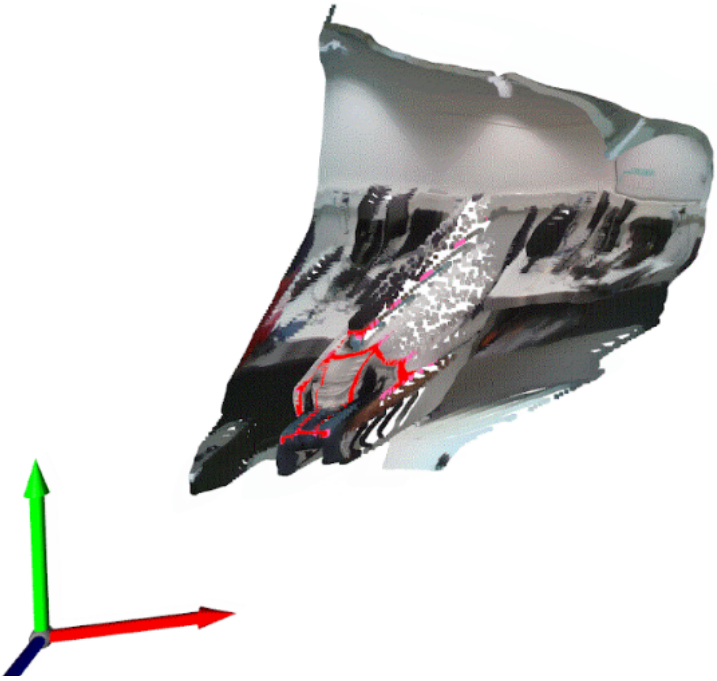


Figure 8: Pointcloud in Cartesian frame with GT from Midas, axis is in the Cartesian frame

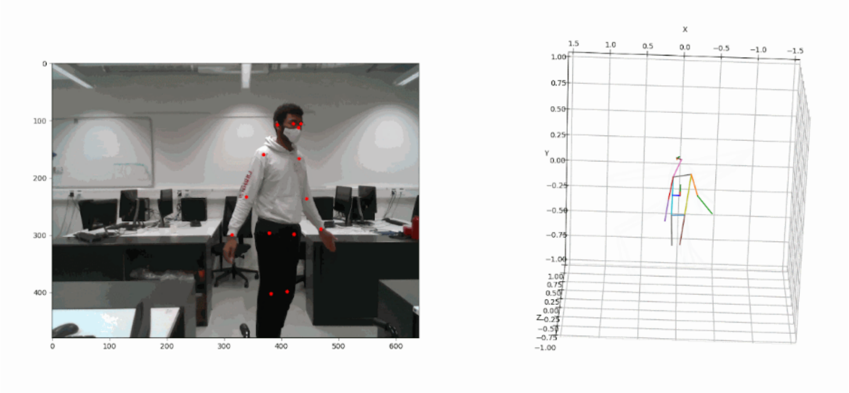


Figure 9: Input image (on the left) and body skeleton in torso frame (on the right)

5 Discussion

The main conclusion of this work is that monocular depth estimation is not suitable for this project. Even though it produces coherent depth map, the best precision is in the range of 30cm with a standard deviation around 20cm in the best cases. We proposed another approach less cost effective including a RGBD camera (D435i) and a mapping from camera to torso frame.

One limitation of this work is when the keypoints are localised but with obstruction. For example, an ankle position predicted in camera frame (u,v) but in the depth map the ankle is located behind some object, the stereo camera will provided the depth of the closest object which will be associated with the ankle. In practice this is an issue particularly when the subject is sideways and some keypoints are obstruct by the body (two shoulder aligned), this will make the projection from the camera to the torso frame inaccurate. A way to compensate for this issue would be to implement and train a neural network to refine the recovered 3D poses.

References

[1] Lorenzo Bertoni, Sven Kreiss, and Alexandre Alahi. *Perceiving Humans: from Monocular 3D Localization to Social Distancing.* 2021. arXiv: 2009.00984 [cs.CV].

[2] Sven Kreiss, Lorenzo Bertoni, and Alexandre Alahi. "Pifpaf: Composite fields for human pose estimation". In: (2019), pp. 11977–11986.

[3] Zhengqi Li et al. "MannequinChallenge: Learning the Depths of Moving People by Watching Frozen People". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43.12 (2021), pp. 4229–4241. DOI: 10.1109/TPAMI.2020.2974454.

[4] Yue Ming et al. "Deep learning for monocular depth estimation: A review". In: *Neurocomputing* 438 (2021), pp. 14–33. ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2020.12.089. URL: https://www.sciencedirect.com/science/article/pii/S0925231220320014.

[5] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. "Vision transformers for dense prediction". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision.* 2021, pp. 12179–12188.

[6] René Ranftl et al. "Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer". In: *arXiv preprint arXiv:1907.01341* (2019).

[7] Jianan Zhen et al. "SMAP: Single-Shot Multi-person Absolute 3D Pose Estimation". In: *Computer Vision – ECCV 2020* (2020). Ed. by Andrea Vedaldi et al., pp. 550–566.