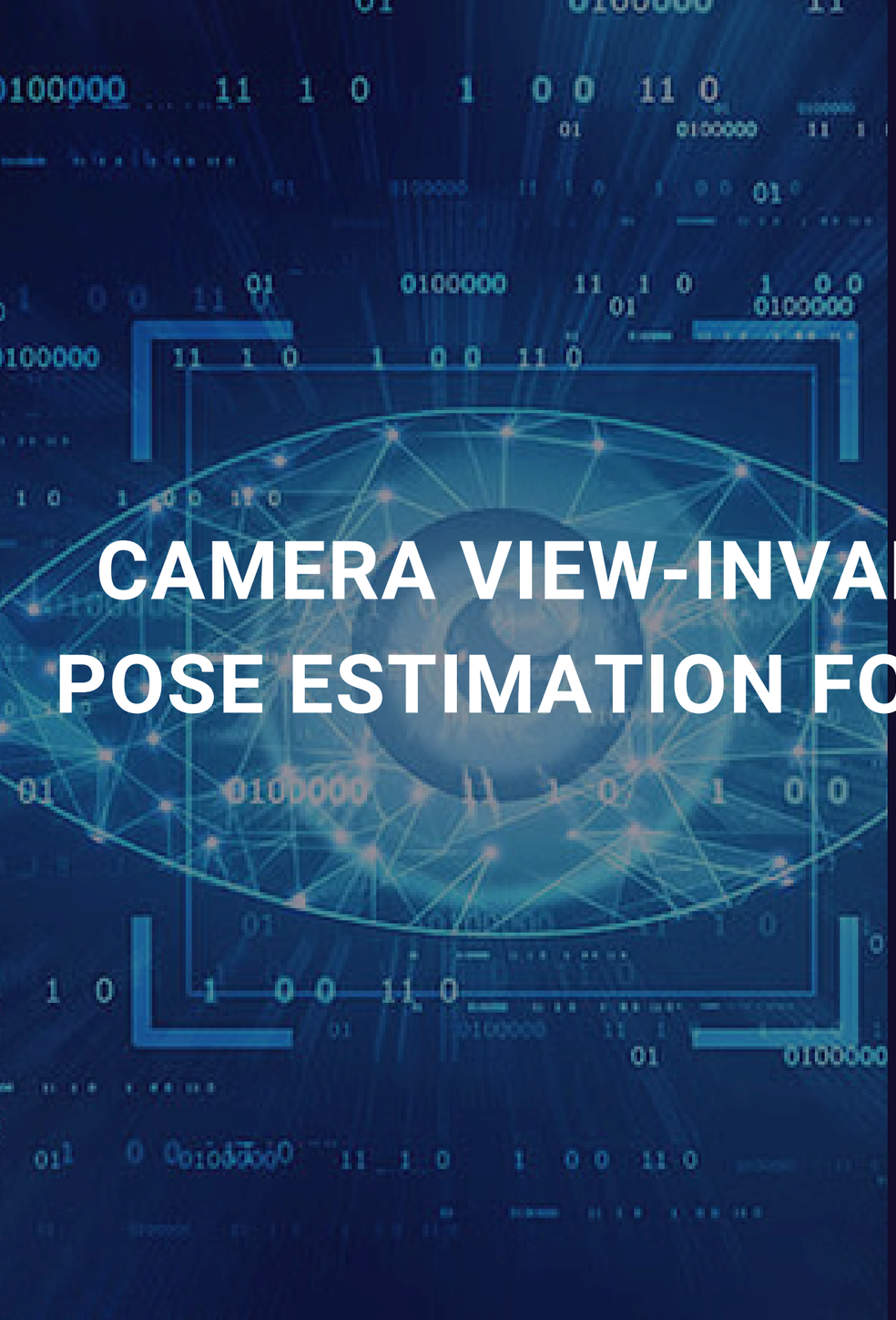


CAMERA VIEW-INVARIANT MULTI-PERSON 2D HUMAN POSE ESTIMATION FOR DEPTH AND FISHEYE CAMERAS

Zarema Balgabekova

23 April, 2024

Human Pose Estimation

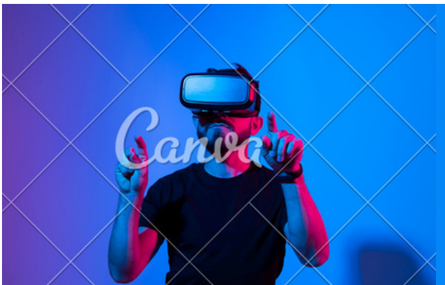
Human Pose Estimation is a computer vision (CV) task of automatically locating the human body parts/joints from images or videos.

When an image or video is given to the pose estimation model as input, it outputs the coordinates of body parts such as shoulders, elbows, knees, eyes, ears, etc.

CV practitioners often refer to body parts as ***keypoints***, while a set of keypoints is known as a ***skeleton***.

Human Pose Estimation can be divided into:

- **2D Pose Estimation:** estimates the locations (x and y coordinates) of keypoints in 2D space relative to input data.
- **3D Pose Estimation:** predicts the spatial positioning of a person by additionally estimating z-dimension.

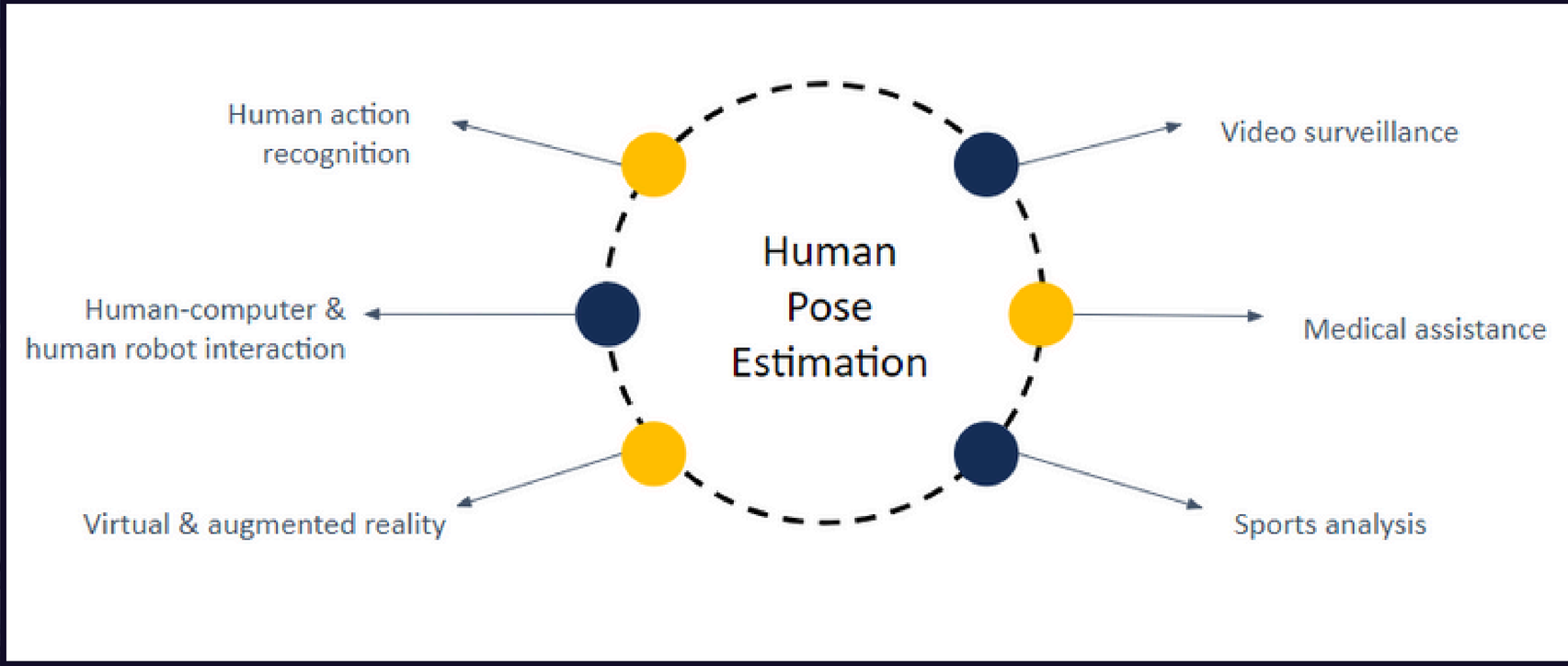


Skeleton Variations in Different Datasets

COCO keypoints (17)

MPII keypoints (16)

Why is Human Pose Estimation Important?



Fisheye Cameras

Large field of view

up to 180 degrees

Image distortion

Especially in the periphery

Reduced costs and less calibration

Fewer cameras are used for the task

Lack of datasets

Most existing datasets are related to egocentric pose estimation

Depth Cameras

Different representation

Pixels measure depth instead of intensity/color

Sensor noise

Retain only coarse appearance details

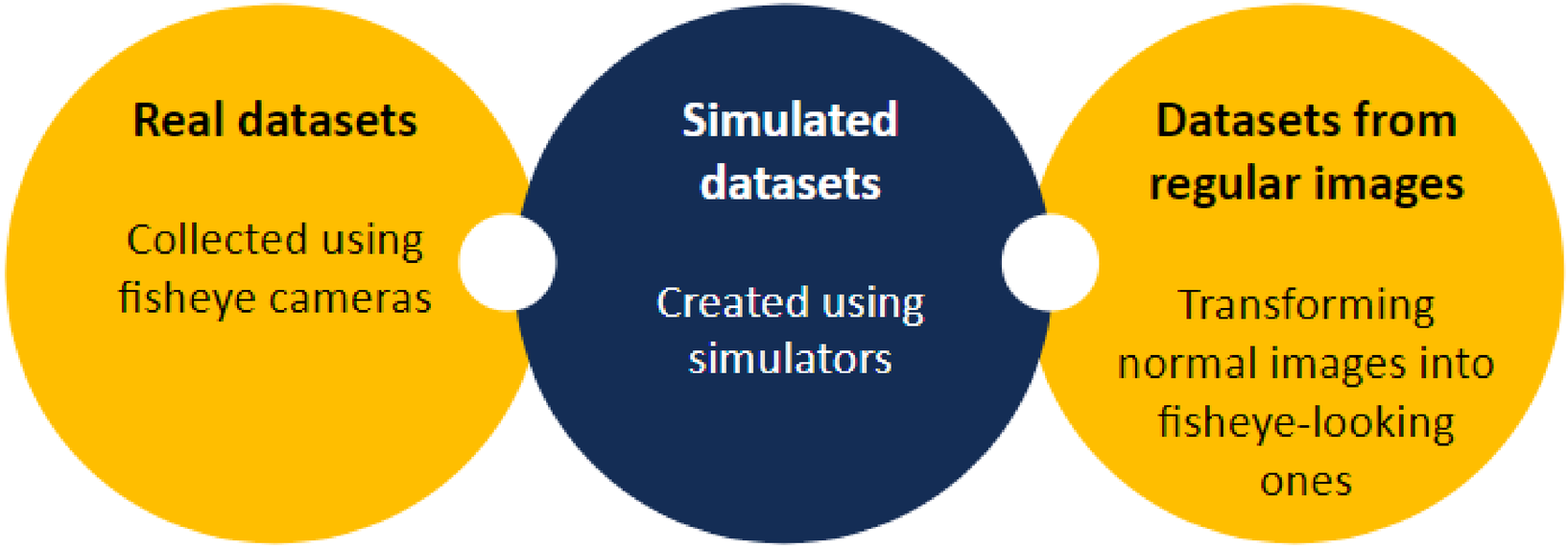
Less susceptible to environment

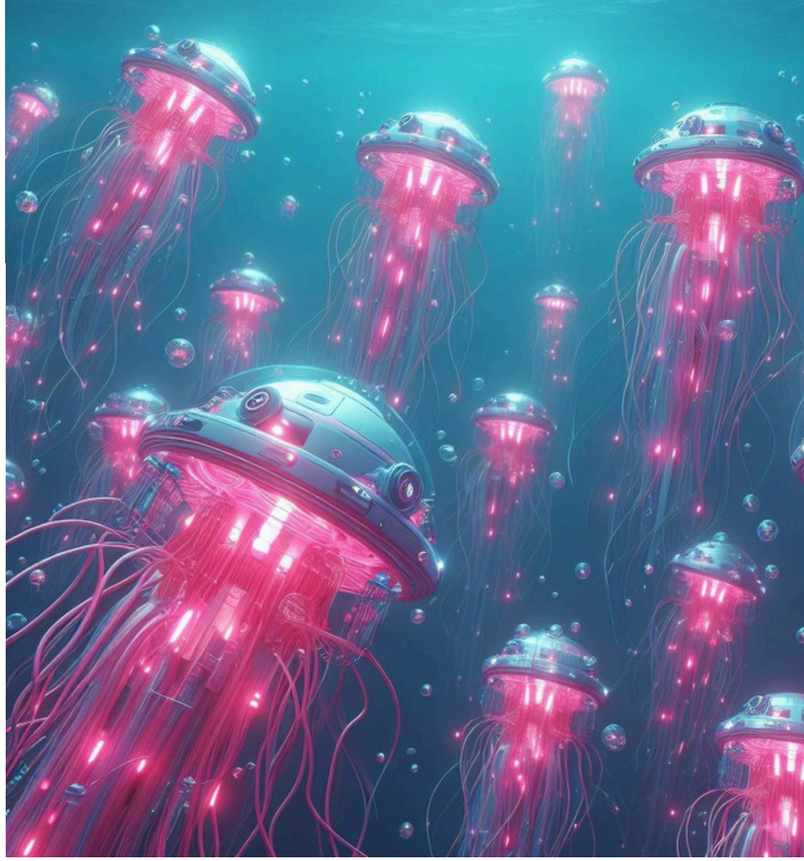
Works in low light levels, color/texture-invariant

Lack of datasets

Most existing datasets contain a single person

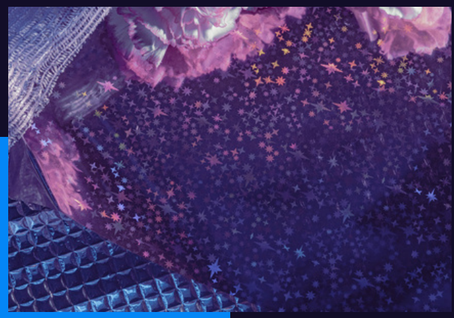
Fisheye Image Datasets





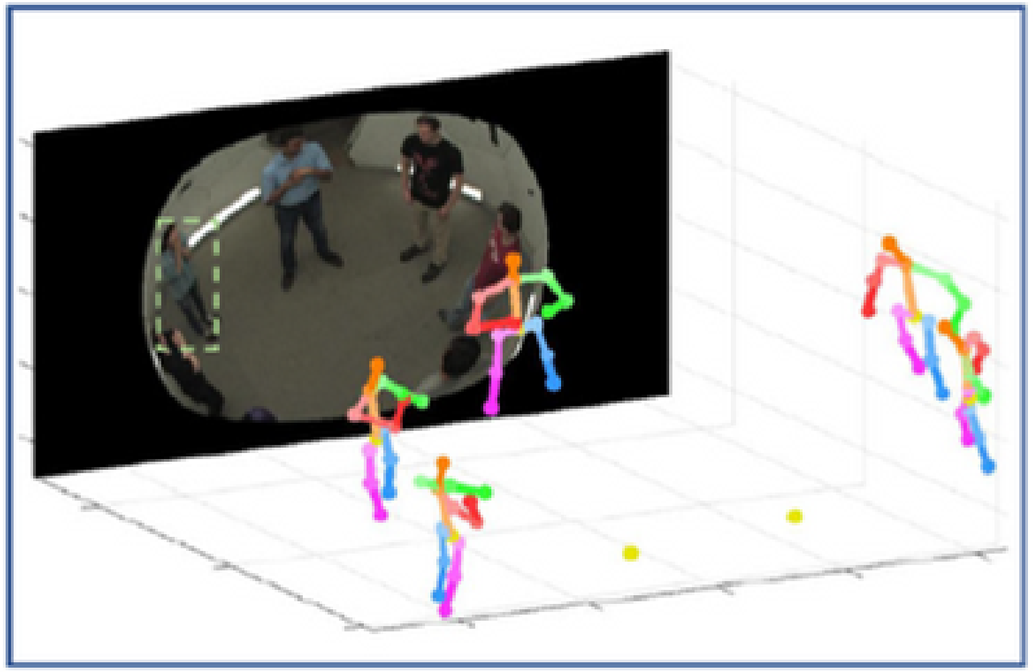
[2] Y. Qian, M. Yang, and J. M. Dolan, “Survey on fish-eye cameras and their applications in intelligent vehicles,” IEEE Transactions on Intelligent Transportation Systems, vol. 23, no. 12, pp. 22 755–22 771, 2022.

Fisheye Datasets for Egocentric Pose Estimation

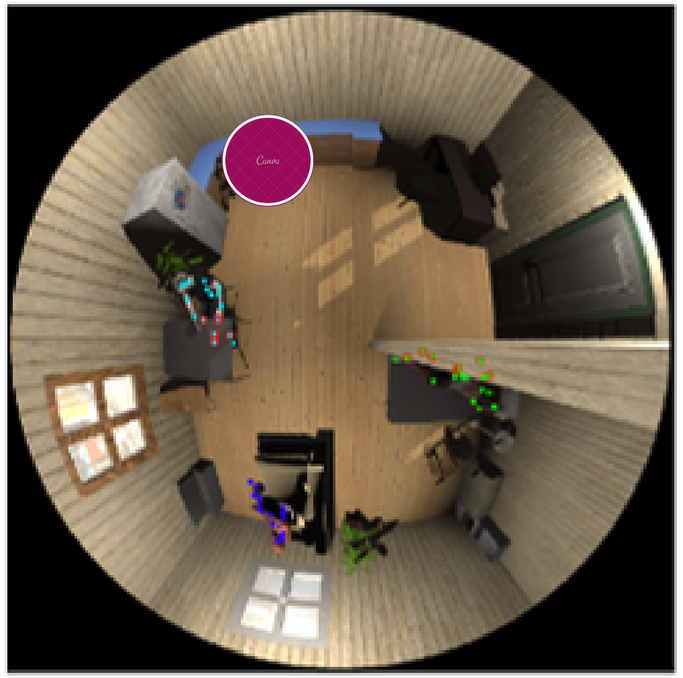


Egocentric pose estimation allows capturing a person’s motion in a large unconstrained space using head/body-mounted cameras.

Synthetic Fisheye Datasets for Pose Estimation



Zhang et al. (2022) [5] created synthetic datasets by transforming regular images (CMU Panoptic and Shelf) to evaluate their multi-person 3D pose estimation algorithm.



THEODORE+ (2023) [6]:
A top-view omnidirectional synthetic dataset
50K images, 160K 3D character instances
13 keypoints,6 environment settings, 8 animations

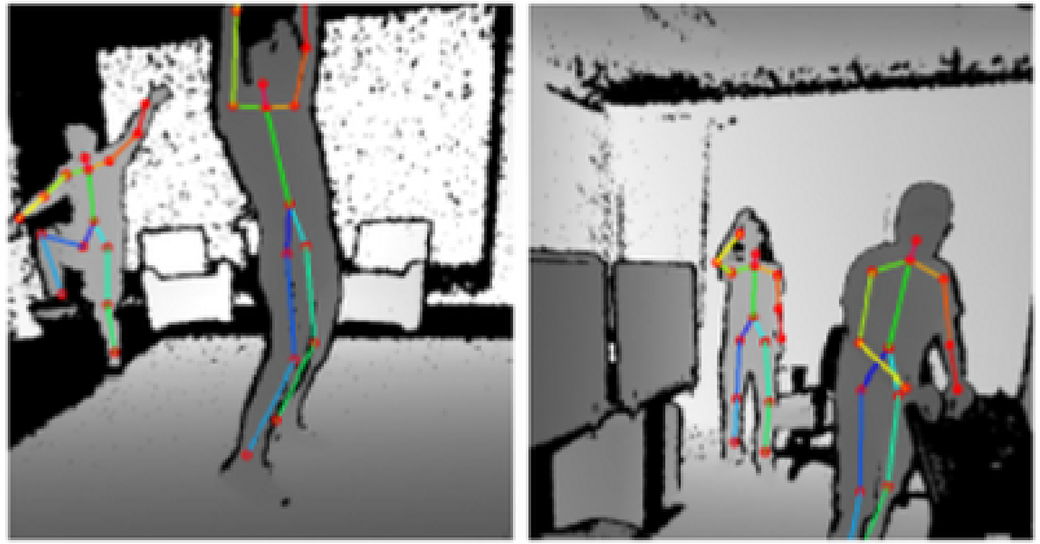
Depth Image Datasets

K2HGD – Kinect2 Human Gesture Dataset (2016) [8]:
100K real images
19 body joints
30 subjects, 10 scenes

ITOP – Invariant-Top View Dataset (2016) [7]:
100K real images
15 body keypoints
20 people, 15 action sequences
front/side and top views

Depth Image Datasets

DIH – Depth Images of Humans (2018) [9]:
170K synthetic images
17 keypoints
24 3D characters
Human-robot interaction scenario



MP-3DHP – Multi-Person 3D Human Pose Dataset (2022) [10]:
Training set: 210K images, 15 subjects, 10 actions, 8 scenes
Testing set: 4.5K images, 5 people, 4 scenes

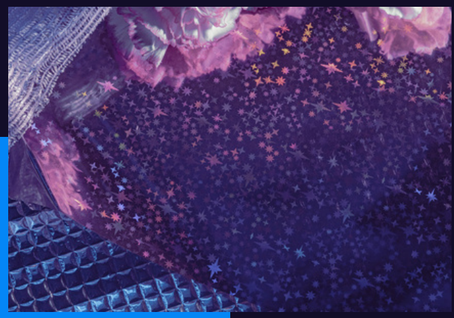
Problem Statement

Will human pose estimation models trained on synthetic fisheye and depth image datasets adapt to real fisheye and depth images?

To answer this research question, I will:

- *Create the most diverse synthetic depth and fisheye image human pose estimation datasets in terms of*
- *Camera views*
- *Number and appearance of 3D characters*
- *Scenes*
- *Train state-of-the-art human pose estimation models (e.g., YOLOv8-pose [11]) on these datasets*
- *Collect and annotate small datasets of real fisheye and depth images*
- *Evaluate the performance of the obtained models on real data*

Technological Enabler: Tools for Synthetic Dataset Generation



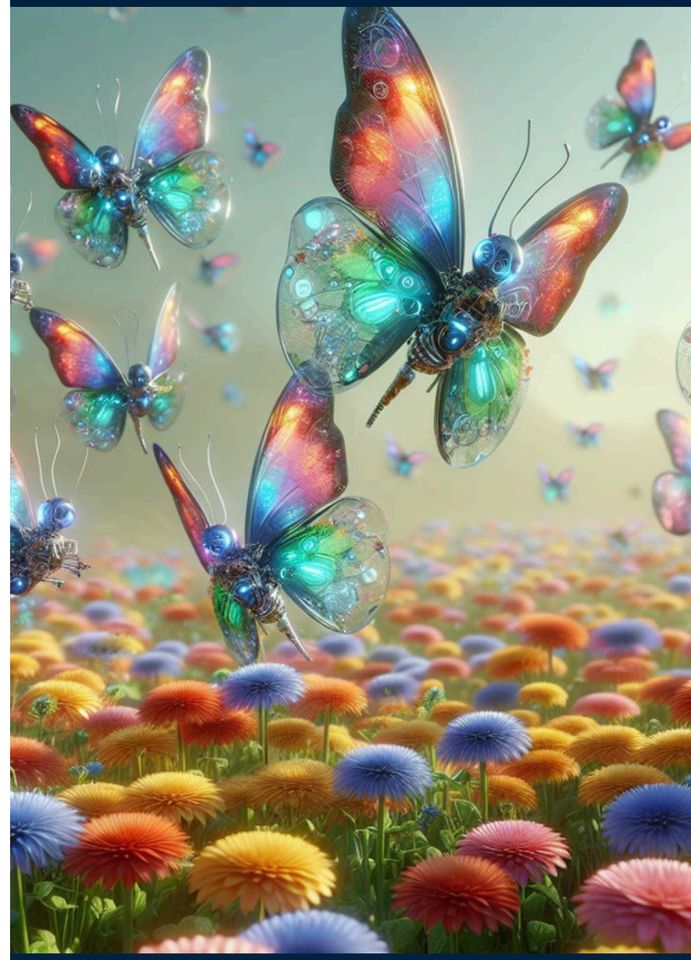
Nvidia Omniverse is a 3D graphics platform, which incorporates various apps and extensions for the creation of photorealistic environments and cinematic animations and simulations.
For this project, the following Omniverse apps and extensions are used:

- USD Composer: used for the creation of realistic scenes
- Replicator: enables synthetic data generation
- Code: allows to run Replicator from a script rather than interacting with a graphical user interface

Besides Nvidia Omniverse, the 3D character design software Character Creator and 3D computer graphics software Blender are used for the creation of 3D characters that are placed in scenes.

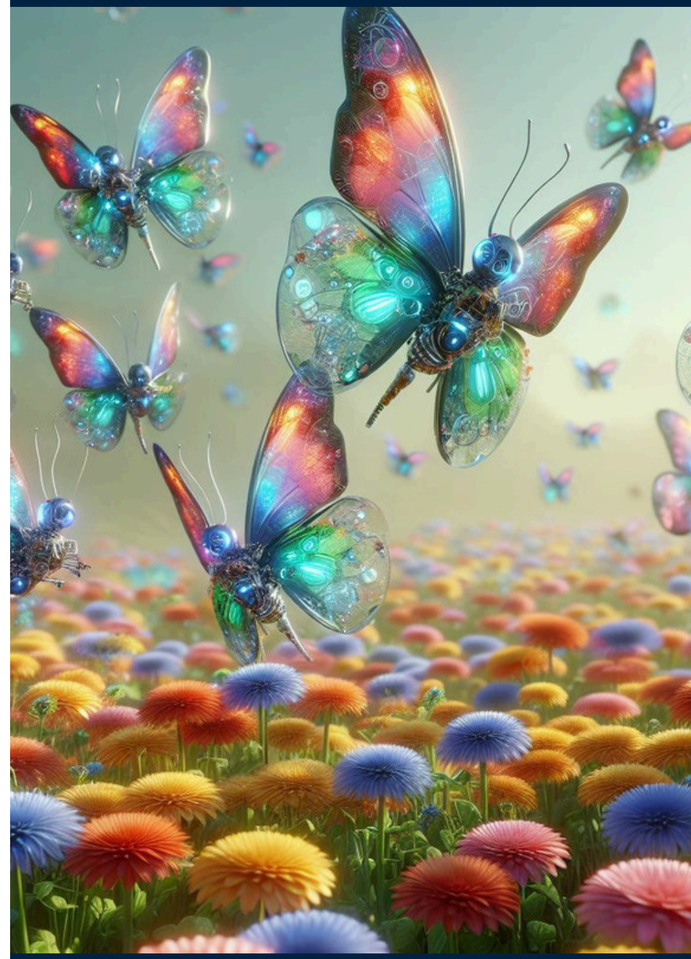
Synthetic Dataset Generation

- Scene creation: 16 different scenes were created using Nvidia Omniverse USD Composer



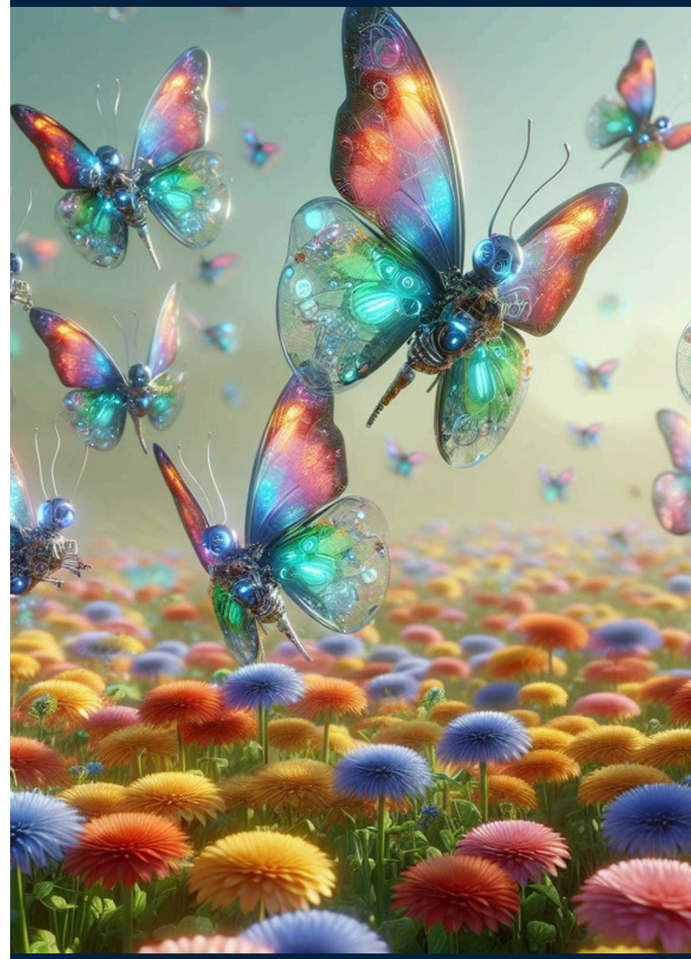
Synthetic Dataset Generation

- Scene creation: 16 different scenes were created using Nvidia Omniverse USD Composer



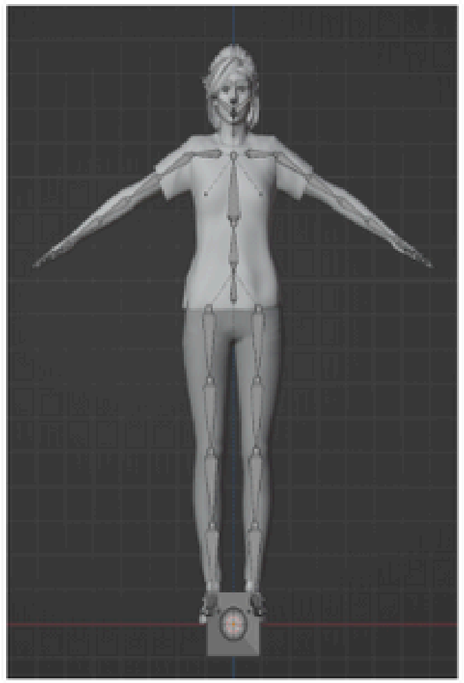
Synthetic Dataset Generation

- Scene creation: 16 different scenes were created using Nvidia Omniverse USD Composer

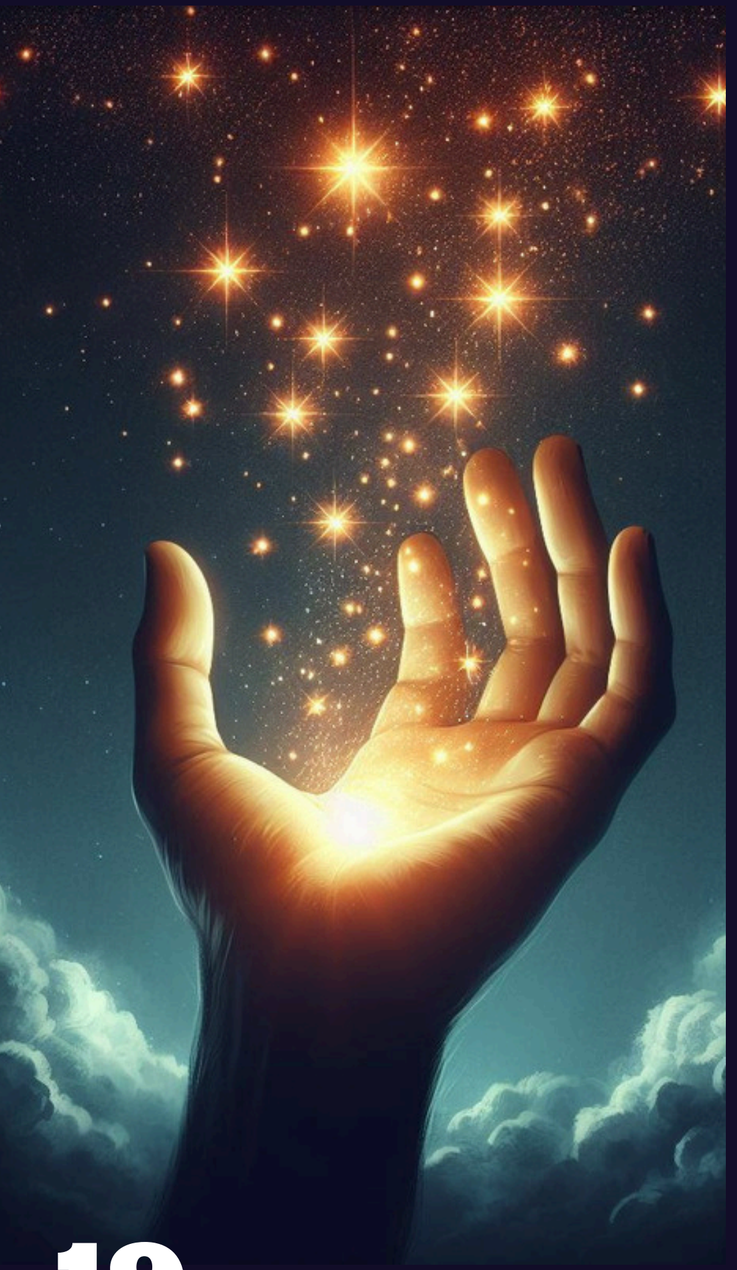


Synthetic Dataset Generation

Character creation: 38 3D characters with various appeara
created using Character Creator and Blender



Synthetic Dataset Generation



Synthetic Dataset Generation

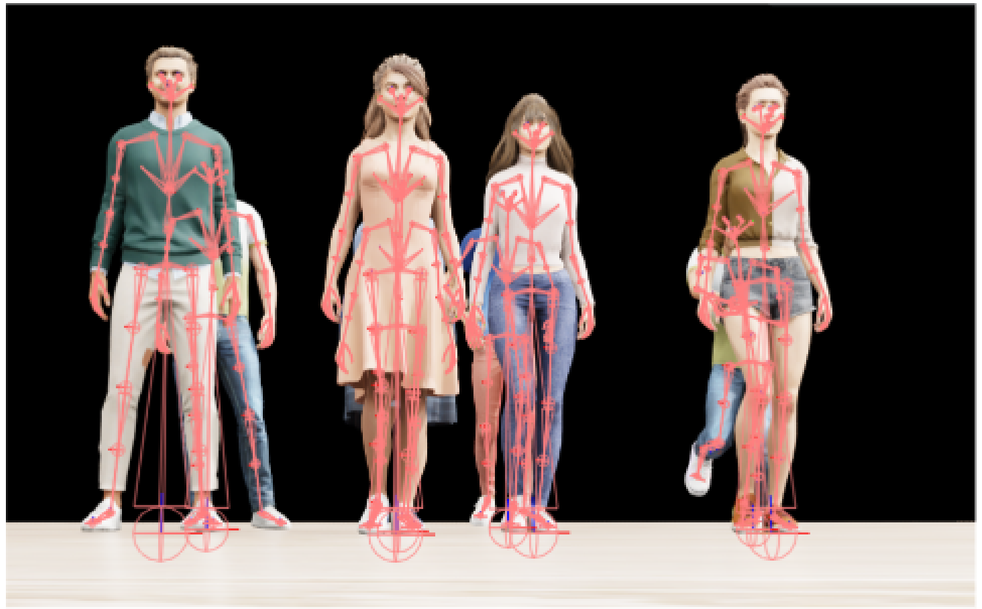
- Placing people in scenes: in groups of 3, 5, and 7 characters with different combinations

- Placing cameras: five predefined positions – front, back, left and right side, and top views + unusual positions/angles

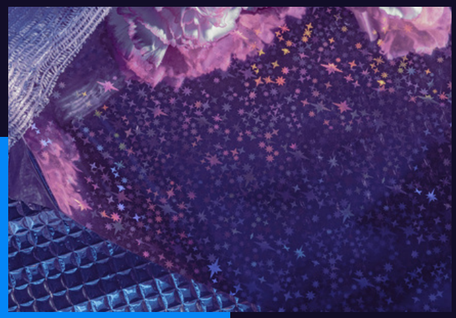
Synthetic Dataset Generation

- Recording the dataset: Nvidia Omniverse extension Replicator and app Code are used to record
  - RGB images for fisheye cameras
  - Distance to camera for depth cameras
  - Skeletal data
  - Bounding boxes

- Preprocessing the dataset:
  - Convert distance to camera into images (add noise?)
  - Convert labels (skeletal data, bounding boxes) into YOLO format



Comparison of Synthetic Fisheye Image Datasets

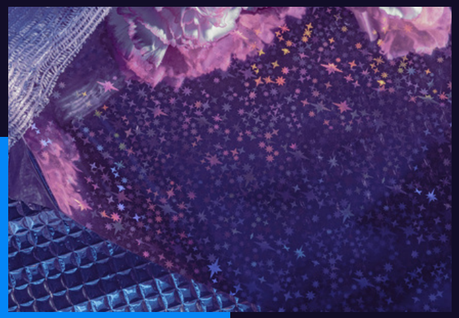


- **The first dataset: the dataset created using the Nvidia Omniverse platform**
- **The second dataset: transforming the Microsoft Common Objects in Context (MS COCO) dataset [12] into fisheye-looking images**

MS COCO is a benchmarking dataset for object detection, instance segmentation, image captioning, and human pose estimation for regular images.

[12] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, and C. L. Zitnick, “Microsoft coco: Common objects in ´ context,” in Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13. Springer, 2014, pp. 740–755.

Data collection



To test trained models, we collected two benchmark testing datasets:

- Fisheye
- Depth

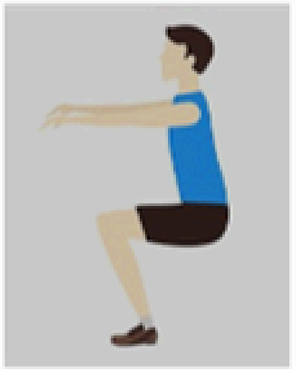
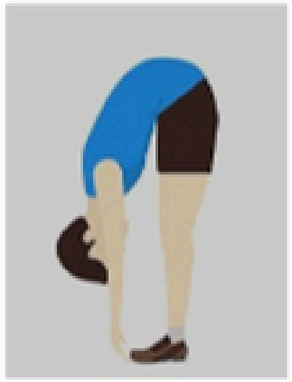
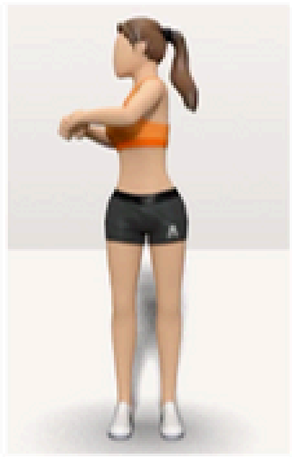
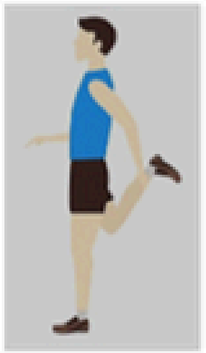
Annotations in Roboflow

- Fisheye - COCO annotations
- Depth - ITOP annotations

Data Collection Protocol

Session – 5 minutes, sub-session – 1 minute

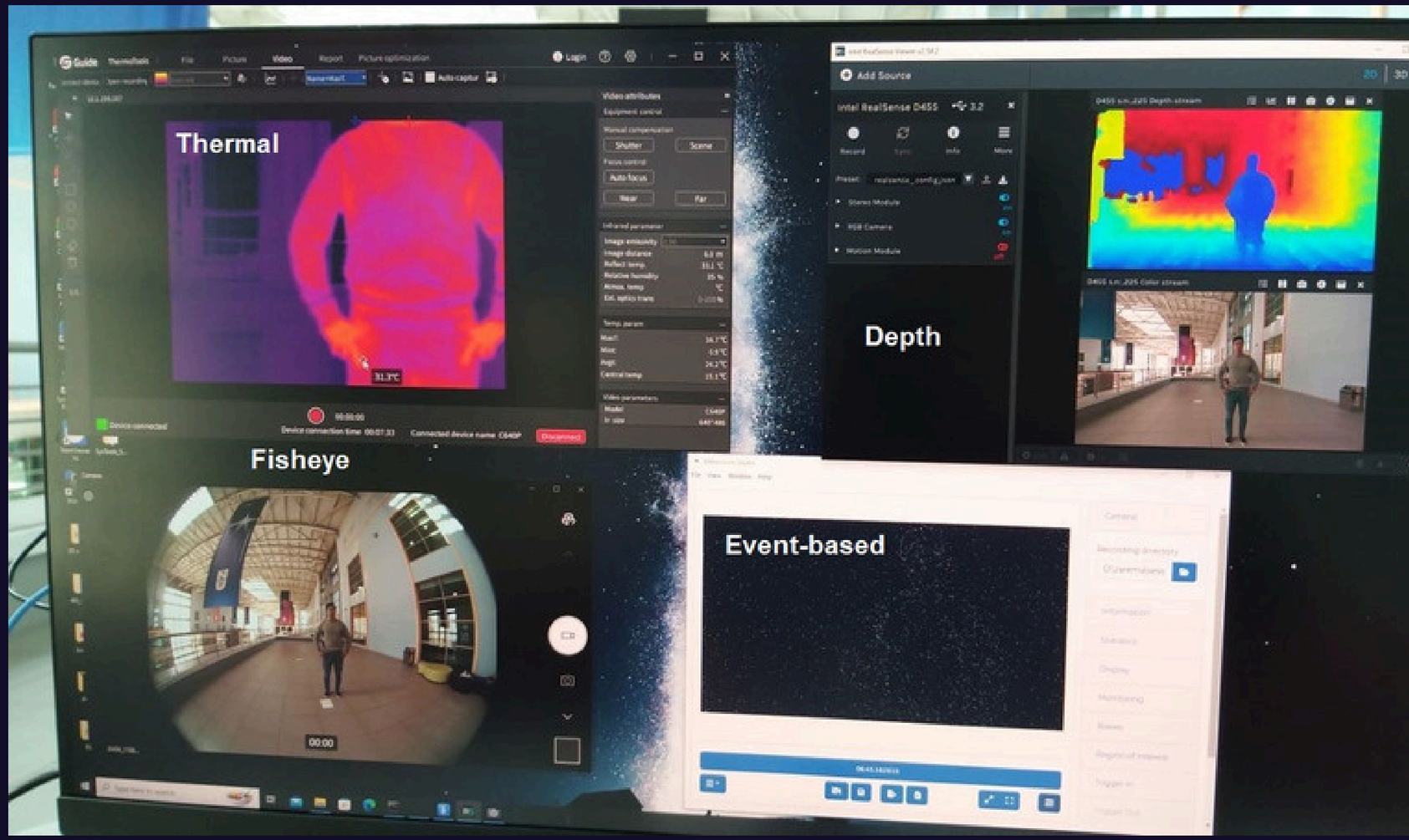
Three people – 3 sessions

	Subject 1	Subject 2	Subject 3
Sub-session 1 (front view)	 5 times  5 times (alternate) Closer to the camera	 5 times  5 times (alternate)	 5 times  5 times (alternate)

Metadata

Data	Stats
Age	20-27
Gender	male/female
Ethnicity (in descending order)	Kazakh, Russian, Nigerian, Pakistani

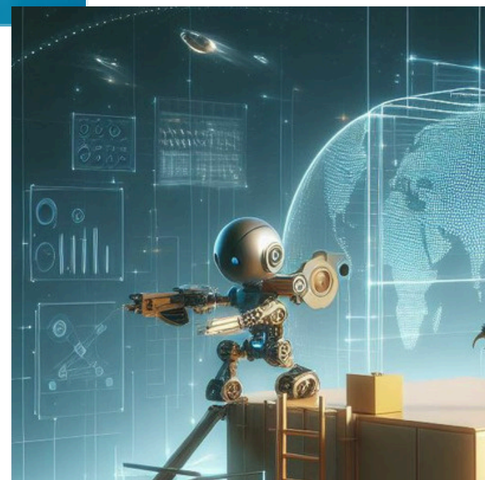
Data Collection Setup



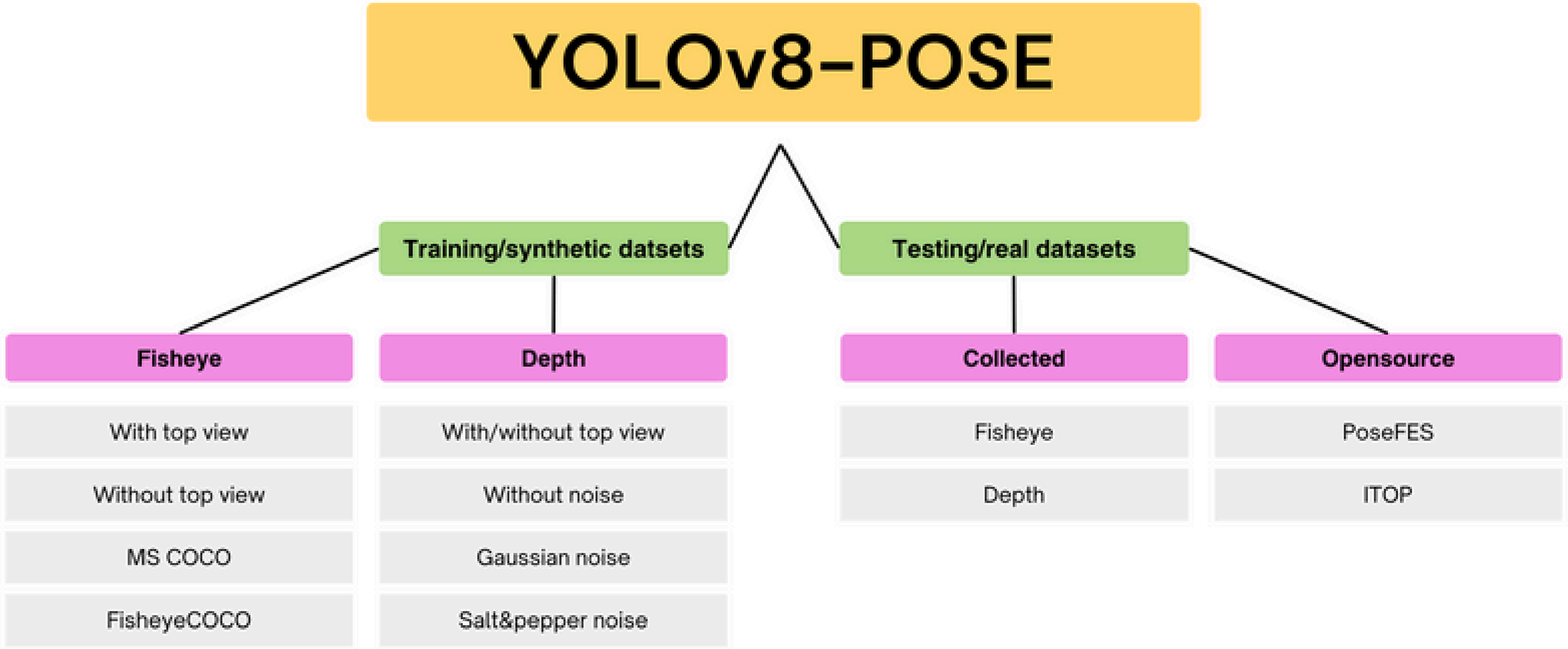
Opensource testing datasets

PoseFES

ITOP



Methodology



Results training

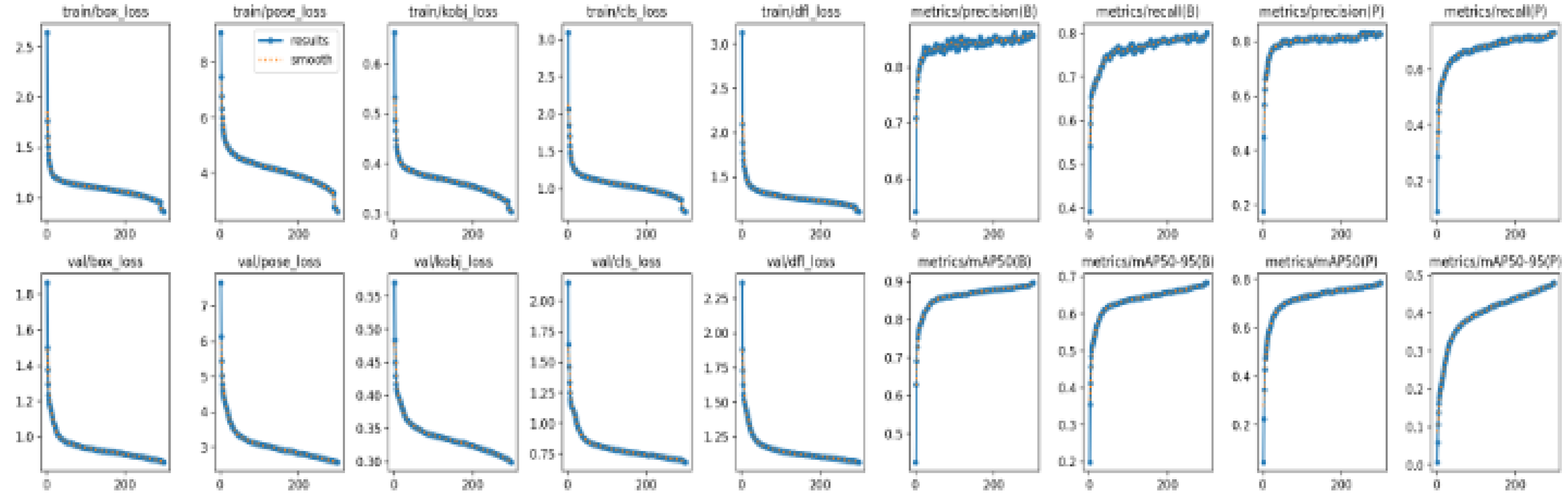
	MS COCO	FISH COCO	Synth top	Synth no top
PoseFES	0.347	0.385	0.226	0.643
Fish Real	?	?	?	?

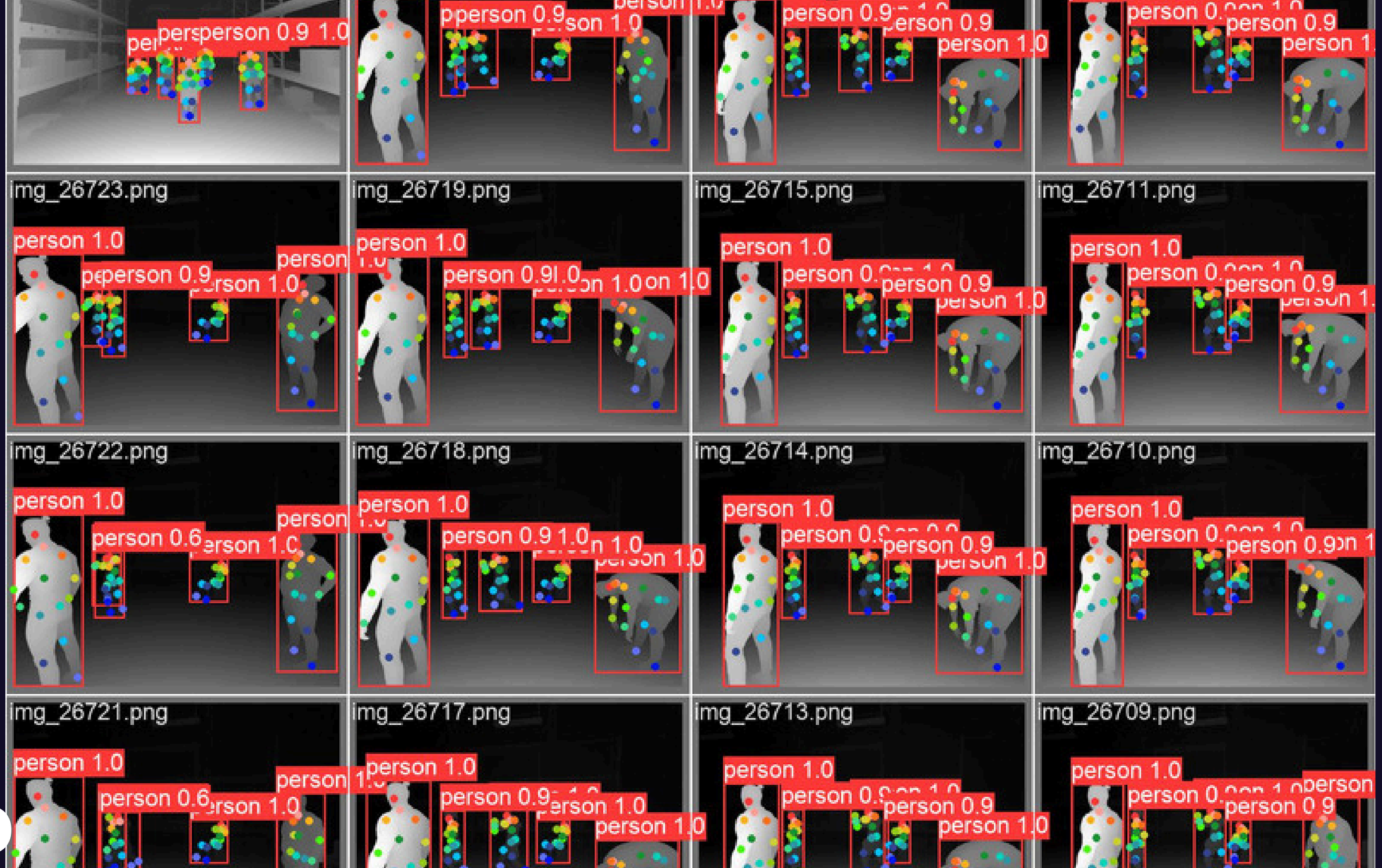
	Depth	Depth no top	Gauss	Gauss no top	Sp	Sp no top
ITOP	0.0672	0.0392	0.0173	0.000785	0	0
Depth Real	2.56	0.607	0.78	4.92	3.07	6.41

Results testing

	MS COCO	FishCOCO	Synth Top	Synth no Top
PoseFES	78.47	44.66	97.38	82.54
Fisheye Real	94.01	83.12	80.08	?
ITOP	81.62	81.34	79.71	?
Depth Real	Accomplishments: To-Do's: Blockers:	Accomplishments: To-Do's: Blockers:	Accomplishments: To-Do's: Blockers:	Accomplishments: To-Do's: Blockers:

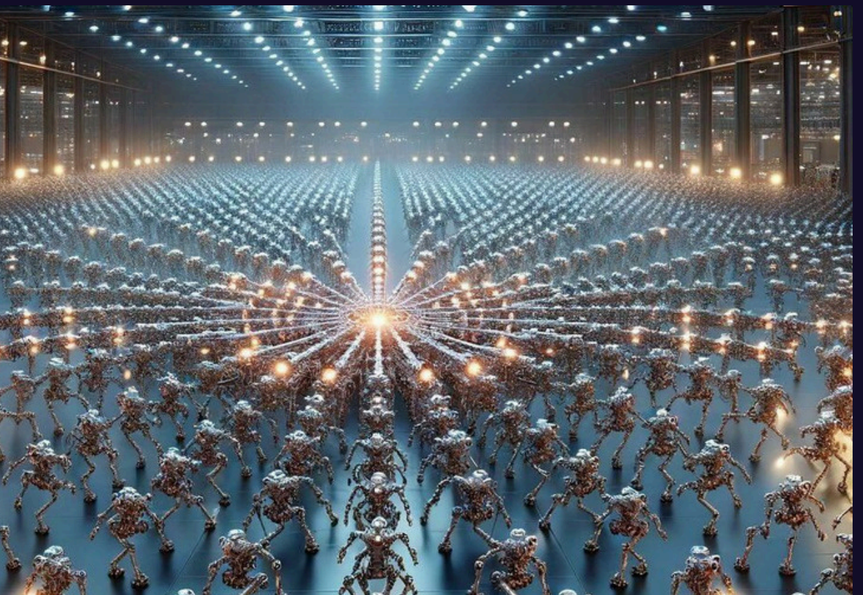
Results: Quantitative




person 1.0
person 0.9
person 0.6
person 0.7
img_26723.png
img_26719.png
img_26715.png
img_26711.png
img_26722.png
img_26718.png
img_26714.png
img_26710.png
img_26721.png
img_26717.png
img_26713.png
img_26709.png

Discussion

- Our models outperform state-of-the-art for RGB images
- There is a greater performance for fisheye images
- Exclusion of top view influences the result
- Noise is also important
- Limitation is the size of datasets
- In future work the model can be used in combination with hardware



Thank you
Any Questions???

Zarema Balgabekova
2nd Year MSc student in Robotics
AI | CV | HRI
Incoming PhD student at USC

@zarema.baloveyourall

References

- [1] T. L. Munea, Y. Z. Jembre, H. T. Weldegebriel, L. Chen, C. Huang, and C. Yang, “The progress of human pose estimation: A survey and taxonomy of models applied in 2d human pose estimation,” *IEEE Access*, vol. 8, pp. 133 330–133 348, 2020.
- [2] Y. Qian, M. Yang, and J. M. Dolan, “Survey on fish-eye cameras and their applications in intelligent vehicles,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 22 755–22 771, 2022.
- [3] W. Xu, A. Chatterjee, M. Zollhoefer, H. Rhodin, P. Fua, H.-P. Seidel, and C. Theobalt, “Mo 2 cap 2: Real-time mobile 3d motion capture with a cap-mounted fisheye camera,” IEEE transactions on visualization and computer graphics, vol. 25, no. 5, pp. 2093–2101, 2019.
- [4] H. Akada, J. Wang, S. Shimada, M. Takahashi, C. Theobalt, and V. Golyanik, “Unrealego: A new dataset for robust egocentric 3d human motion capture,” in European Conference on Computer Vision. Springer, 2022, pp. 1–17.
- [5] Y. Zhang, S. You, S. Karaoglu, and T. Gevers, “Multi-person 3d pose estimation from a single image captured by a fisheye camera,” Computer Vision and Image Understanding, vol. 222, p. 103505, 2022.
- [6] J. Yu, T. Scheck, R. Seidel, Y. Adya, D. Nandi, and G. Hirtz, “Human pose estimation in monocular omnidirectional top-view images,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 6410–6419.
- [7] A. Haque, B. Peng, Z. Luo, A. Alahi, S. Yeung, and L. Fei-Fei, “Towards viewpoint invariant 3d[illegible] estimation,” in Computer Vision– ECCV 2016: 14th European Conference, Amsterdam, The N[illegible] October 11–14, 2016, Proceedings, Part I 14. Springer, 2016, pp. 160–177.

References

- [8] K. Wang, S. Zhai, H. Cheng, X. Liang, and L. Lin, “Human pose estimation from depth images via inference embedded multi-task learning,” in Proceedings of the 24th ACM international conference on Multimedia, 2016, pp. 1227–1236.
- [9] A. Mart´ınez-Gonzalez, M. Villamizar, O. Can ´ evet, and J.-M. Odobez, ´ “Real-time convolutional networks for depth-based human pose estimation,” in 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2018, pp. 41–47.
- [10] Y. Guo, Z. Li, Z. Li, X. Du, S. Quan, and Y. Xu, “Pop-net: Pose over parts network for multi-person 3d pose estimation from a depth image,” in Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2022, pp. 1240–1249.
- [11] G. Jocher, A. Chaurasia, and J. Qiu, “YOLO by Ultralytics,” Jan. 2023. [Online]. Available: https://github.com/ultralytics/ultralytics
-
- [12] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, and C. L coco: Common objects in ´ context,” in Computer Vision–ECCV 2014: 13th European C Switzerland, September 6-12, 2014, Proceedings, Part V 13. Springer, 2014, pp