Computer Vision Project Report

Marco Morandin ~ 257849

University of Trento

1 Introduction

Human pose estimation (HPE) is a core problem in computer vision with broad applications in sports analysis, augmented reality, robotics and human-computer interaction. Since many of these applications require a good motion understanding, extending HPE to three dimensions becomes important.

Multi-camera 3D pose estimation addresses this need by combining synchronized views to triangulate 3D joint locations from 2D detections. This strategy enhances accuracy and robustness, particularly in cases where parts of the body are hidden from individual camera perspectives.

This project develops a complete pipeline for multi-camera 3D HPE, beginning with image rectification to correct lens distortions, followed by 2D pose estimation using advanced deep learning models. The process then applies 3D triangulation, validates results through reprojection, and concludes with evaluation.

2 Technical Background

From a technical perspective, multi-camera 3D human pose estimation builds on several core concepts in computer vision.

2.1 Rectification

Image rectification in OpenCV is the process of transforming images so that corresponding points in two or more views align on the same horizontal line, which is especially important in multi-views applications. Using camera calibration parameters OpenCV provides `initUndistortRectifyMap()` function to remove lens distortions and warp the images so that their epipolar lines become horizontal and parallel. This simplifies feature matching and correspondence search, which are critical for computing depth and reconstructing 3D structure. By rectifying images before triangulation, errors in point matching are reduced and the accuracy of the 3D reconstruction pipeline is improved.

2.2 Triangulation

Triangulation reconstructs 3D points from corresponding 2D observations across multiple camera views. The Direct Linear Transform (DLT) method is used, which formulates 3D reconstruction as a linear system that can be solved using Singular Value Decomposition (SVD). For each keypoint, the algorithm constructs a system of equations from the camera projection matrices and 2D observations, finding the 3D point that minimizes reprojection error. The implementation includes optional bundle adjustment using SciPy optimization to refine the triangulated 3D poses by jointly optimizing all keypoints and camera parameters.

2.3 Reprojection

Reprojection validates 3D pose estimates by projecting them back onto the 2D image planes using camera intrinsic and extrinsic parameters. This process serves for quality assessment by computing reprojection errors between predicted and ground truth 2D keypoints and for visualization of 3D pose accuracy in the original camera views.

2.4 YoloPose

YOLOPose v11 is built upon the YOLO11 architecture, a unified framework designed for multiple vision tasks including detection, segmentation, classification, and pose estimation. The architecture integrates an improved backbone and neck with lightweight yet effective modules such as C3k2 (an optimized Cross-Stage Partial block), SPPF (Spatial Pyramid Pooling – Fast), and C2PSA (a parallel spatial attention mechanism), which collectively enhance feature representation and computational efficiency. The pose estimation variants are pretrained on the COCO keypoints dataset, which provides 17 annotated human body joints. This pretraining enables the model to achieve robust human pose estimation while maintaining the speed and efficiency characteristic of the YOLO family.

2.5 VitPose

ViTPose++ is a state-of-the-art human pose estimation framework based on the Vision Transformer (ViT) architecture. While models like YOLO excel at detecting objects in an image and generating bounding boxes, they do not excel on providing fine-grained information about the positions of individual body joints. ViTPose++ improved this by predicting detailed keypoint locations for the human body within detected regions. In a typical workflow, YOLO (or any other human detector) is first used to detect humans in an image, producing bounding boxes that isolate each person. These regions are then fed into ViTPose++, which leverages a non-hierarchical ViT encoder to extract high-dimensional visual features. The extracted features pass through a lightweight decoder that predicts heatmaps for each body keypoint, allowing precise localization of joints such as elbows, knees, and shoulders.

ViTPose++ extends the original ViTPose framework by incorporating a Mixture of Experts (MoE) module, which includes both task-agnostic and task-specific feed-forward networks. This design enables the model to share general knowledge across different pose estimation tasks while adapting to dataset-specific requirements, improving performance on heterogeneous datasets. Furthermore, the model benefits from masked autoencoder (MAE) pretraining on large-scale image datasets, which enhances feature representation and generalization to unseen poses or occluded keypoints. ViTPose++ has been benchmarked on multiple datasets, including COCO, MPII, AI Challenger, OCHuman, and COCO-WholeBody, achieving state-of-the-art accuracy for both single-person and multi-person pose estimation.

2.6 ProbPose

ProbPose introduces a probabilistic framework for 2D human pose estimation, addressing limitations in current models that often overlook out-of-image keypoints and rely on uncalibrated heatmaps for keypoint localization. Unlike traditional approaches that focus solely on in-image keypoint detection, ProbPose enhances robustness by predicting, for each keypoint, a calibrated probability of presence at each location within the activation window, the probability of being outside of it, and its predicted visibility. This comprehensive probabilistic representation allows for more accurate localization, especially for keypoints near the image boundaries or occluded regions. To evaluate these improvements, ProbPose introduces the CropCOCO dataset, which includes out-of-image keypoints, and the Extended OKS (Ex-OKS) metric, extending the standard OKS to account for false positive predictions. Empirical results demonstrate that ProbPose significantly improves out-of-image keypoint localization and enhances in-image localization through data augmentation techniques. Additionally, the model exhibits better flexibility in keypoint evaluation, making it a valuable tool for applications requiring precise human pose estimation in challenging scenarios.

2.7 HPE evaluation metrics

Several standard metrics are employed for human pose estimation evaluation:

- **PCK (Percentage of Correct Keypoints)**: Measures the percentage of predicted keypoints within a threshold distance of ground truth.
- **PCKh**: Head-normalized PCK that uses head size for scale normalization, making it more robust to person scale variations.
- **MPJPE (Mean Per Joint Position Error)**: Average Euclidean distance between predicted and ground truth 3D joint positions.
- **PA-MPJPE (Procrustes Analysis MPJPE)**: MPJPE after optimal alignment using Procrustes analysis, removing global pose differences.
- **MPJAE (Mean Per Joint Angular Error)**: Angular error between predicted and ground truth joint orientations.

3 Methodology

This section details the end-to-end pipeline designed for multi–camera 3D human pose estimation. The system is deliberately modular and orchestrated through Hydra configuration management, enabling independent execution and evaluation of each stage.

3.1 Pipeline Architecture

The implementation follows a modular design with two main processing modes: `DatasetPipeline` for offline processing of image sequences with ground truth annotations, and `VideoPipeline` for video processing. The dataset mode executes the following sequential stages: rectification, ground truth triangulation, pose estimation with multiple models, and comprehensive evaluation. While the video mode rectifies the input, runs pose estimation algorithms and then triangulate the results.

3.2 Rectification

Before pose estimation, all frames undergo undistortion and rectification using OpenCV's `initUndistortRectifyMap` with pre-computed camera calibration parameters. The `Rectifier` class processes both images and their corresponding COCO annotations, updating keypoint coordinates to match the rectified image space. This ensures geometric consistency across all subsequent pipeline stages.

3.3 2D Pose Estimation

The pipeline implements two pose estimation approaches through dedicated estimator classes:

- **YOLOv11 Pose** (`YOLOPoseEstimator`) performs end-to-end pose detection with integrated person detection and keypoint estimation. The model utilizes pre-trained weights (`yolo11l-pose.pt`) with a confidence threshold of 0.25 for keypoint filtering.
- **ViTPose++** (`ViTPoseEstimator`) employs a two-stage approach: YOLO11 for person detection followed by the HuggingFace ViTPose++ model for refined keypoint localization with confidence threshold 0.5.

Both estimators process rectified images and update the COCO dataset annotations in-place. Keypoint pruning removes foot keypoints based on configuration patterns, and virtual keypoints (hips, neck, spine) are computed from existing joint positions in order to convert the predicted COCO keypoints into a unified format compared to the GT. The `COCOManager` class handles all dataset operations including annotation clearing, keypoint mapping, and format conversions between model-specific outputs and standard COCO representation.

ProbPose ProbPose was not included in the current version of the pipeline. While initial exploration and testing showed that the installation worked correctly and the model produced reasonable results, issues arose when integrating it into the complete pipeline. Specifically, conflicting dependencies and the unavailability of some required packages made the installation problematic.

3.4 3D Triangulation

The `PlayerTriangulator` class implements 3D pose reconstruction from multi-view 2D keypoints. Annotations are first grouped by frame number using filename parsing to identify camera and frame indices. For each frame, visible keypoints across all camera views are collected.

For each joint with 2D observations from cameras where the joint is visible, the Direct Linear Transform (DLT) is formulated. It is solved via Singular Value Decomposition from SciPy library. The closed-form solution serves as initialization for an optional bundle adjustment stage that minimizes the reprojection error using Levenberg-Marquardt optimization through SciPy's `least_squares` function. Triangulation requires a minimum of two camera views per keypoint.

3.5 Reprojection and Validation

The resulting 3D skeleton is projected back to each camera to compute evaluations and to generate qualitative overlays that facilitate debugging. An evaluation script computes PCK, PCKh, MPJPE, MPJAE and PA-MPJPE against available ground truth.

4 Results

The evaluation process compares the performance of YOLO and ViTPose models across multiple stages of the pipeline.

Model	**Method**	**PCKh@0.2**	**PCKh@0.5**	**MPJPE**	**PA-MPJPE**	**Detection Rate**
Ground Truth	Triangulation	9%	48%	20.50	10.06	–
YOLO Pose	Prediction	39%	85%	23.71	20.65	67%
	Triangulation	17%	67%	25.86	16.99	–
ViTPose	Prediction	54%	91%	15.65	14.60	88%
	Triangulation	20%	75%	23.47	14.43	–

5 Conclusions

This project implemented and evaluated a multi-camera 3D human pose estimation pipeline comparing YOLO Pose and ViTPose architectures. The experimental results demonstrate ViTPose's clear superiority across all evaluation metrics.

ViTPose significantly outperformed YOLO Pose in direct pose prediction, achieving 91% PCKh@0.5 accuracy compared to 85%, and substantially lower error rates with 15.65px MPJPE versus 23.71px. ViTPose also demonstrated superior detection reliability at 88% compared to YOLO Pose's 67%.

The superior performance of ViTPose can be attributed to the Vision Transformer architecture's ability to model long-range spatial dependencies, enabling more robust keypoint localization in challenging scenarios. While YOLO Pose showed lower accuracy metrics, it may remain viable for real-time applications where computational efficiency is prioritized over precision.

References

1. YOLO by Ultralytics. `https://github.com/ultralytics/ultralytics`
2. Xu, Y., Zhang, J., Zhang, Q., Tao, D.: ViTPose: Simple vision transformer baselines for human pose estimation.
3. Xu, Y., Zhang, J., Zhang, Q., Tao, D.: ViTPose++: Vision Transformer for Generic Body Pose Estimation.