FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Efficient Estimation of the 3D Pose in a Multi-Camera Environment

Rúben Costa Viana

Mestrado em Engenharia Informática e Computação

Supervisor: Luís Filipe Teixeira

Supervisor at INESC TEC: Ana Filipa Rodrigues Nogueira

July 30, 2025

Efficient Estimation of the 3D Pose in a Multi-Camera Environment

Rúben Costa Viana

Mestrado em Engenharia Informática e Computação

Approved in oral examination by the committee:

President: Prof. António Coelho
Referee: Prof. Rui Lopes
Referee: Prof. Luís Filipe Teixeira

July 30, 2025

Resumo

A estimativa da pose humana em 3D representa um problema fundamental e de longa data no domínio da visão computacional. O objetivo desta estimativa é recuperar com precisão a configuração do corpo inteiro dos indivíduos numa determinada cena. Apesar dos progressos consideráveis na estimativa da pose 2D através da aprendizagem profunda, a reconstrução 3D robusta, nomeadamente do volume do corpo, continua a ser um desafio. Os ambientes com várias câmaras oferecem perspectivas complementares que podem ser utilizadas para atenuar problemas como oclusões e pontos de vista limitados que são inerentes às abordagens de visão única. No entanto, existem desafios persistentes, incluindo a escassez de conjuntos de dados anotados em ambientes não controlados com múltiplas perspectivas, dificuldades na deteção de poses em cenas com muita gente e a complexa fusão de poses em múltiplas perspectivas 2D, o que pode levar a cálculos volumétricos ineficientes. A procura de uma estimativa de pose 3D precisa, eficiente e implementável em tempo real em diversos cenários continua a ser crítica para uma variedade de aplicações, incluindo veículos autónomos, reabilitação, avaliação desportiva, interação homem-computador, realidade aumentada/virtual e sistemas de vigilância avançados.

A presente dissertação explora o desenvolvimento de um novo sistema distribuído de estimação da pose humana em 3D. Este sistema foi concebido para enfrentar os desafios previamente identificados, equilibrando a precisão e a eficiência em ambientes reais com várias câmaras. Além disso, explora os compromissos inerentes entre precisão, tempo de processamento e escalabilidade. A arquitetura proposta distribui estrategicamente a carga computacional, tirando partido da computação periférica para a extração inicial do mapa térmico da pose 2D e de um servidor centralizado para a subsequente reconstrução da pose 3D, utilizando o modelo de última geração, FasterVoxelPose. Uma análise exaustiva do desempenho demonstra as capacidades do sistema em tempo real, alcançando taxas de fotogramas competitivas tanto nos dispositivos periféricos como no servidor central, juntamente com uma precisão de ponta. Assim, este trabalho de investigação estabelece uma estrutura distribuída robusta e eficiente para a estimação da pose humana em 3D, oferecendo uma solução viável para aplicações exigentes em tempo real, como vigilância, reabilitação física e avaliações desportivas.

Abstract

3D human pose estimation represents a pivotal and long-standing problem in the field of computer vision. The objective of this estimation is to accurately recover the full body configuration of individuals within a given scene. Despite considerable progress in 2D pose estimation via deep learning, robust 3D reconstruction, particularly of the body volume, remains challenging. Multi-camera environments offer complementary perspectives that can be used to mitigate issues such as occlusions and limited viewpoints that are inherent to single-view approaches. However, there are persistent challenges, including the scarcity of annotated "in-the-wild" multi-view datasets, difficulties in detecting poses in crowded scenes, and the complex fusion of identities across multiple 2D views, which can lead to inefficient volumetric computations. The demand for accurate, efficient, and real-time deployable 3D pose estimation across diverse scenarios remains critical for a variety of applications, including autonomous vehicles, rehabilitation, sports assessment, human-computer interaction, augmented/virtual reality, and advanced surveillance systems.

The present dissertation is an exploration of the development of a novel distributed 3D human pose estimation system. This system has been designed to address the challenges previously identified by balancing accuracy and efficiency in real-world multi-camera environments. Furthermore, it explores the inherent trade-offs between accuracy, processing time, and scalability. The proposed architecture strategically distributes computational load, leveraging edge computing for initial 2D pose heatmap extraction and a centralised server for subsequent 3D pose reconstruction, utilising the state-of-the-art FasterVoxelPose model. A thorough performance analysis demonstrates the system's real-time capabilities, achieving competitive frame rates at both edge devices and the central server, coupled with state-of-the-art accuracy. Therefore, this research work establishes a robust and efficient distributed framework for 3D human pose estimation, offering a viable solution for demanding real-time applications such as surveillance, physical rehabilitation, and sports assessments.

UN Sustainable Development Goals

The United Nations Sustainable Development Goals (SDGs) provide a global framework to achieve a better and more sustainable future for all. It includes 17 goals to address the world's most pressing challenges, including poverty, inequality, climate change, environmental degradation, peace, and justice.

This work contributes to multiple United Nations SDGs through the development of a distributed 3D human pose estimation system. By leveraging edge computing and efficient AI models, the system enables real-time, privacy-conscious human motion analysis applicable to healthcare, workplace safety, smart cities, and sustainable infrastructures.

Therefore, aligning with the following specific Sustainable Development Goals:

SDG 3 Ensure healthy lives and promote well-being for all at all ages

SDG 8 Promote sustained, inclusive and sustainable economic growth, full and productive employment and decent work for all

SDG 9 Build resilient infrastructure, promote inclusive and sustainable industrialization and foster innovation

SDG 11 Make cities and human settlements inclusive, safe, resilient and sustainable

SDG	Target	Contribution	Performance Indicators and Metrics
3	3.d	Enables real-time health monitoring and posture assessment in healthcare and rehabilitation contexts.	Reduction in injury/rehabilitation time; accuracy of posture analysis; number of users benefiting from remote care.
	3.4	Supports early detection of musculoskeletal issues by analyzing posture and movement patterns.	Improvement in diagnosis lead time; decrease in fall/injury incidents; feedback loop usage statistics.
8	8.2	Improves productivity and safety in industrial environments through motion tracking and worker posture analysis.	Number of workplaces adopting system; reduction in work-related strain/injuries; processing latency.
9	9.5	Contributes to R&D in computer vision and distributed systems through novel system architecture.	Research publications; citation metrics; collaboration with innovation-focused institutions.
11	11.5	Enables safer public environments through smart crowd and posture analysis in urban settings.	Incident detection rate; real-time alert responsiveness; integration in smart city pilot programs.

Acknowledgements

I would like to express my deepest gratitude to my supervisors, Professor Luís Teixeira and Ana Filipa, for their guidance, patience, and support throughout this research.

I sincerely thank INESC TEC for providing me with the opportunity and valuable resources during this project.

Special thanks to my colleagues and friends, whose encouragement and companionship were instrumental in maintaining my motivation throughout this endeavour. The provision of support during challenging periods, in addition to the moments of laughter and inspiration, has been of the utmost value.

Finally, I would like to thank my family, particularly my parents, for their unwavering support, trust, and understanding throughout this amazing journey.

Rúben Costa Viana

"Everything you can imagine is real."

Pablo Picasso

Contents

List of Figures

List of Tables

Listings

List of Acronyms

AP	Average Precision.
CPN	Cuboid Proposal Network.
DLT	Direct Linear Transform.
DPD	Depth-wise Projection Decay.
EDN	Encoder-Decoder Network.
FPS	Frames Per Second.
GUI	Graphical User Interface.
HDN	Human Detection Network.
HPE	Human Pose Estimation.
IP	Internet Protocol.
JLN	Joint Localization Network.
LSTM	Long Short Term Memory.
MAE	Mean Average Error.
MPJPE	Mean Per Joint Position Error.
NTP	Network Time Protocol.
PCP	Percentage of Correct Parts.
PRISMA	Preferred Reporting Items for Systematic Reviews and Meta-Analyses.
PRN	Pose Regression Network.
SDGs	Sustainable Development Goals.
SOTA	State-of-the-art.
UDP	User Datagram Protocol.

Chapter 1

Introduction

This chapter provides a brief introduction to the topic of 3D pose estimation, highlighting the need for a multi-camera solution that can effectively operate in real-time. Hence, presenting the problem that this thesis aims to tackle, as well as the pressing research questions. Afterwards, a summary of the content of each chapter is presented.

1.1 Context and Motivation

3D Human Pose Estimation (HPE), which is one of the most important computer vision tasks in the past few decades [1] and has been a longstanding problem [13], aims to recover the body configuration of all humans in a scene. The successful resolution of 3D pose estimation can benefit many real-world applications. Some examples include autonomous vehicles, rehabilitation, sports assessments, gaming, assisted living, and surveillance systems; for example, in public places like airports, illegal behaviour (e.g. assault, stealing) can be easily identified, leading to a faster intervention of the authorities. In addition, other areas are human-computer interaction, augmented reality (AR), and virtual reality (VR).

In recent years, considerable advancements have been observed in the domain of 2D pose estimation through the utilisation of deep learning methodologies [1]. However, these approaches have not been optimally suited for the reconstruction of the body volume, a crucial aspect in 3D pose estimation

In the early stages of development, the methodologies employed for 3D pose estimation were based on a single view. These approaches were confronted with a multitude of challenges, including the lack of depth information, severe occlusions, and the scarcity of labelled datasets.

Multi-camera environments provide an opportunity to overcome some of the challenges associated with occlusion and limited viewpoints [13, 15]. This is achieved by providing complementary perspectives that enrich the available information. However, the lack of annotated in-the-wild datasets and the difficulty of detecting poses in crowded scenes are problems that still remain unsolved. Also, the fusion of information from multi-view images is challenging since the identity

of the 2D poses from each camera view is unknown [11], and the computation of the volumetric space is of high complexity, making the current methods inefficient.

Despite efforts to create precise methods with close to real-time inference, further research is still necessary. Therefore, developing a method that can accurately and efficiently estimate the 3D poses across different scenarios and that can be deployed in real-time would be of great value for many applications. For instance, its integration into current surveillance systems would cause significant improvements over traditional ones, allowing for better monitoring of crowded places with fewer human efforts by the automated detection of suspicious behaviours and threats. This could improve security in places such as airports and festivals.

1.2 Problem

The main challenge in accurate 3D pose estimation in a multi-camera environment is the efficiency of the models. The current methods are highly computationally complex and expensive, making them unsuitable for real-world scenarios where real-time inference is necessary.

This problem is particularly relevant for areas like security and surveillance, where real-time pose detection is needed, autonomous systems, where 3D awareness is key, medical rehabilitation, where a precise analysis of the patient movements is required, and many others.

That being the case, this research focuses on trying to make the current methods more efficient or developing a new, more efficient method that is capable of estimating the 3D pose of a human in a multi-camera environment that can be used in real-time.

1.3 Research Questions

With this research, the objective is to tackle the following research questions:

- **RQ1:** How can 3D pose estimation in a multi-camera environment balance accuracy and efficiency in real-world scenarios?
- **RQ2:** What are the trade-offs between accuracy, processing time, and scalability in 3D pose estimation?

1.4 Outline of the Thesis

The rest of this thesis is structured as follows:

- **Chapter 2 - Literature Review:** This chapter presents a systematic literature review conducted using the PRISMA methodology. It details the criteria and procedures followed in each stage of the methodology, ensuring a comprehensive and rigorous survey of existing research relevant to 3D HPE.

- **Chapter 3 - State-of-the-Art:** This chapter provides an in-depth overview of the current state-of-the-art in 3D HPE, including a discussion of recent methods, commonly used datasets, and the standard evaluation metrics adopted by the research community.
- **Chapter 4 - Distributed System:** This chapter describes the design and configuration of the distributed system developed for this dissertation. It outlines the architecture, computational setup, pose estimation models, datasets used for training and evaluation, and the metrics employed to assess both accuracy and computational efficiency.
- **Chapter 5 - Experiments & Results:** This chapter presents the experimental procedures and results. It details the conducted experiments, evaluates the performance of the proposed system, and compares it against state-of-the-art approaches in terms of accuracy, efficiency, and scalability.
- **Chapter 6 - Conclusion & Future Work:** This chapter concludes the thesis by summarizing the key findings, discussing the system's limitations, and outlining directions for future research.

Chapter 2

Literature Review

A systematic literature review was conducted using the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) methodology. This strategy allows existing literature to be reviewed in a thorough, methodical and accurate manner. Therefore, this chapter presents the criteria followed in the different stages of the PRISMA methodology.

2.1 Research Domain

Methods to accurately and efficiently estimate the 3D human pose have been widely explored over the years. This, alongside with the increase availability of multi-camera environments and better hardware lead to the development and research of new solutions that could be used in real time.

The goal of this review is to retrieve and assess as much information as possible about the methods that were developed related to the 3D pose estimation of a human using multiple views. To ensure that the processes of collecting, screening and inclusion are consistent and systematic, the PRISMA methodology is used. Figure 2.1 shows in detail the steps taken at each phase of PRISMA.

2.2 Identification Phase

In the identification phase, a search is performed in known scientific digital libraries in order to collect knowledge regarding the working topic.

To start, four digital libraries are chosen as the data sources, IEEE Xplore, Scopus, ScienceDirect and ACM Digital Library. This is followed by the definition and refinement of the keywords that will formulate the search strings.

2.2.1 Search Strings Formulation and Refinement

The digital library IEEE Xplore was used for the process of testing and refining the search strings.

The initial search string used was ("3D pose estimation") AND (multi-camera). Executing this search string results in few records. After inspection, was concluded that many of the papers use

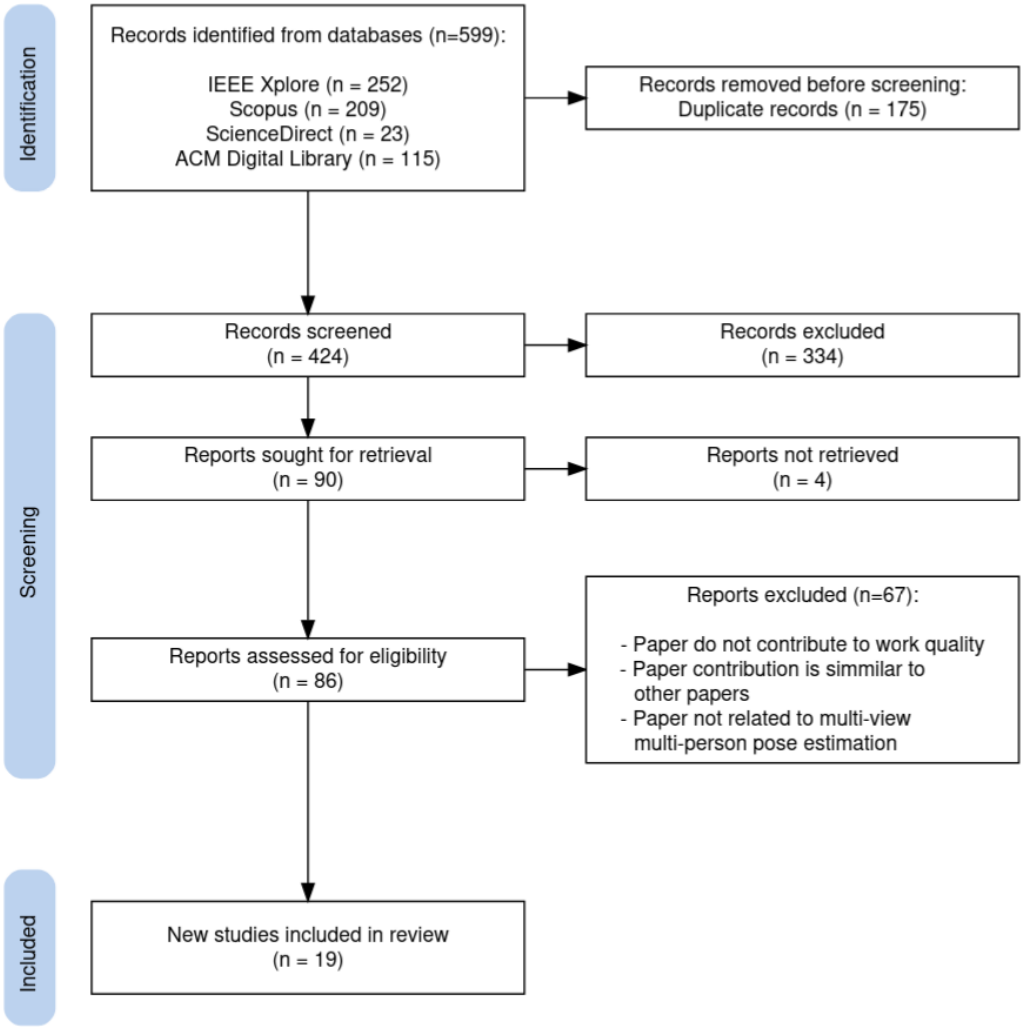

Figure 2.1: PRISMA Flow Diagram

the term "human pose estimation" instead of "3D pose estimation". The same happens with the term "multi-camera" being replaced by "multi-view". So, the solution found was to add this new terms as synonyms.

Although the goal of this research is to find an efficient method to perform the 3D HPE in a multi-camera environment, "efficient" is not used as a keyword because the objective of the review is to find as much related works with different approaches without that restriction.

The refinement process results in the following search string:

SS1. ("3D pose estimation" OR "human pose estimation") AND (multi-view OR multi-camera)

In the end, after executing the search query in all data sources, a total of 590 records were retrieved, the number of papers by library is shown in Table 2.1. These results were registered in Zotero, the references management system used.

Table 2.1: Search string results per search database

	SS1
IEEE Xplore	252
Scopus	209
ScienceDirect	23
ACM Digital Library	115

2.2.2 Search Criteria

Since 3D HPE has been explored for years, the search results are filtered to match the ones published between 2020 and 2024 (inclusive), that are conference papers or article journals and in the area of "Computer Science" or "Engineering". These search criteria are adapted according to each data source capabilities:

- IEEE Xplore: 2020-2024 conferences and journals (search within all metadata);
- Scopus: 2020-2024 English conference papers and articles in the subject area of "Computer Science" or "Engineering" (search within Article title, Abstract, Keywords);
- ScienceDirect: 2020-2024 articles in the subject area of "Computer Science" or "Engineering" (search within Title, Abstract, Keywords);
- ACM Digital Library: 2020-2024 publications (search within Title, Abstract).

2.2.3 Remove Duplicates

The process of finding and removing duplicates is performed using Zotero's Zoplicate plugin, and the criteria for merging duplicates is modified to retain the record with the most information. The result of this process is a dataset of 424 unique papers and 175 duplicates.

2.3 Screening Phase

The PRISMA screening phase entails the definition of inclusion and exclusion criteria and a meticulous assessment of the selected records. Initially, each distinct paper is screened based on its title and abstract, followed by a verification of its full-text accessibility. Later, a full-text screening is conducted on the remaining records to confirm their eligibility for a final inclusion.

2.3.1 Inclusion and Exclusion Criteria

A total of 424 records are transposed from the previous phase. Thus, the need for defining inclusion and exclusion criteria, Table 2.2, is essential to find the most relevant works.

Table 2.2: Inclusion and exclusion criteria

Phases	Inclusion/Exclusion Criteria
1	Search engine results after executing a search query
2	Year of the publication since 2020 (inclusive)
3	Published in the English language
4	Not duplicated
5	Article or Conference Paper
6	3D pose estimation of a human
7	Not single-view nor single-camera settings

2.3.2 Screened Records

In this process, 424 unique records are inspected by their title and abstract in order to exclude the ones that do not match the criteria outlined in Table 2.2, such as language, type, etc. This results in the exclusion of 334 records, keeping 90 records.

2.3.3 Full-text Access

This step consists on the retrieval of the PDFs for the remaining records with the help of Zotero's "find full text" tool. This tool found and imported 40 PDFs. The other PDFs were retrieve manually. Only 4 records did not have full-text access. Therefore, the total number of papers at this stage of the literature review is 86.

2.3.4 Eligibility

During the eligibility phase, a comprehensive assessment and analysis of the paper content is conducted for all 86 papers. In addition to the criteria highlighted in Table 2.2, other criteria are employed to facilitate the selection of papers for inclusion in this literature review, as illustrated in the "Reports excluded" box in Figure 2.1.

In order to facilitate and guide the content analysis, a set of data was defined as being necessary for retrieval. These parameters are as follows:

- Year
- Research Purpose
- Application Domain
- Datasets used
- Tools used

2.4 Included Phase

This represents the final phase of the PRISMA framework. The content assessment and analysis conducted in the preceding stage is presented here for the 19 remaining papers, which are deemed relevant to this study. This is illustrated in the tables 2.3, 2.4, 2.5, 2.6, 2.7 and 2.8, which are organised by published year.

Table 2.3: 2020 Paper's Content Assessment and Analysis

Papers	Research Purpose	Approach/Domain	Datasets
[13]	Proposes VoxelPose, a novel approach for 3D multi-person pose estimation from multiple camera views. The method directly operates in 3D space using voxel-based feature aggregation to address occlusion and correspondence challenges.	Combines a Cuboid Proposal Network (CPN) for localizing people and a Pose Regression Network (PRN) for detailed pose estimation. Operates in a voxelized 3D space to fuse data from multiple views.	Campus, Shelf, CMU Panoptic

Table 2.4: 2021 Paper's Content Assessment and Analysis

Papers	Research Purpose	Approach/Domain	Datasets
[3]	Proposes a real-time framework for 3D HPE using distributed smart edge sensors with semantic feedback to improve pose accuracy and computational efficiency in multi-view setups.	Implements 2D pose detection locally on edge sensors using MobileNetV3 and semantic feedback loops from a central backend to refine local detections. Uses triangulation and a skeleton model for 3D pose estimation.	Human3.6M, Campus, Shelf
[11]	Proposes a novel plane-sweep-based approach for multi-view multi-person 3D pose estimation, addressing cross-view fusion and 3D pose reconstruction in a single unified framework without requiring explicit cross-view matching.	Implements a plane sweep stereo method to regress depth for each joint in 2D poses, with person-level and joint-level depth regression in a coarse-to-fine manner. Avoids explicit correspondence matching, enhancing efficiency.	Campus, Shelf, CMU Panoptic

Table 2.5: 2022 Paper's Content Assessment and Analysis

Papers	Research Purpose	Approach/Domain	Datasets
[18]	Proposes ER-Net, a recalibration network using a channel attention mechanism and optimal calibration strategy to improve the accuracy of 3D pose estimation in multi-view multi-person scenarios, especially for occluded joints.	Employs a coarse-to-fine framework using Coarse-Net and Fine-Net for person-level and joint-level depth regression, supported by recalibration modules to enhance feature learning and optimize joint dependencies.	Campus, Shelf
[5]	Proposes a novel, fast, and robust approach for multi-person 3D pose estimation and tracking using multiple calibrated views. Aims to improve efficiency and robustness by leveraging multi-way matching and temporal tracking.	Introduces a multi-way matching algorithm combining geometric and appearance cues for clustering 2D poses into 3D poses. Extends with temporal tracking and filtering for multi-view videos.	Campus, Shelf, CMU Panoptic
[15]	Proposes Faster VoxelPose, a method for multi-view multi-person 3D pose estimation. It aims to reduce computational costs by replacing 3D CNNs with 2D CNNs to achieve real-time performance in large scenes.	Utilizes orthographic projection to re-project 3D volumetric features into 2D coordinate planes for lightweight processing. Introduces Human Detection Network (HDN) for bounding box generation and Joint Localization Network (JLN) for fine-grained pose estimation.	Campus, Shelf, CMU Panoptic
[12]	Proposes an automatic calibration method for multi-camera systems using human body joints as calibration points instead of traditional calibration patterns. Aims to simplify the process, reduce dependency on specific tools, and improve flexibility.	Utilizes a bottom-up 2D pose detector (OpenPose) to identify human joints as calibration points. Iteratively refines intrinsic and extrinsic camera parameters through reprojection error minimization.	-

Table 2.6: 2023 Paper's Content Assessment and Analysis

Papers	Research Purpose	Approach/Domain	Datasets
[4]	Introduces TEMPO, a method for efficient multi-view, multi-person 3D pose estimation. Aims to enhance pose accuracy, enable tracking, and predict future poses using temporal context while maintaining computational efficiency.	Combines spatio-temporal features using a recurrent architecture for 3D pose estimation, tracking, and forecasting.	CMU Panoptic, Human3.6M, Campus and Shelf, EgoHumans
[10]	Proposes a distributed real-time 3D pose estimation system for multi-view settings, leveraging asynchronous cameras and timestamp-based synchronization to ensure constant frame rates and low latency.	Implements a distributed framework with edge devices for 2D pose detection and a central server for 3D pose reconstruction via geometric triangulation using the Direct Linear Transform (DLT) method.	-
[6]	Proposes a camera-selecting device-edge collaborative inference (CDC-MCTPE) framework to reduce inference latency and energy consumption for real-time multi-camera 3D pose estimation while maintaining estimation accuracy.	Integrates a device-edge co-inference strategy, model splitting, and a Random-Ordered Greedy Algorithm (ROGA) to optimize latency, energy use, and camera selection.	MS COCO, Campus
[17]	Proposes VoxelTrack, a framework for multi-person 3D pose estimation and tracking in challenging environments with wide baseline cameras, leveraging voxel-based representations to improve robustness and accuracy.	Introduces a voxel-based 3D representation for pose estimation directly in 3D space, avoiding noisy 2D associations. Uses sparse 3D CNNs for efficient processing and introduces occlusion-aware Re-ID features for tracking.	Campus, Shelf, CMU Panoptic
[7]	A distributed framework for real-time 3D HPE using asynchronous multi-view cameras. Aims to address computational load, bandwidth limitations, and synchronization issues in multi-camera systems.	Combines edge devices for 2D pose detection and a central server for 3D pose reconstruction using geometrical triangulation (DLT method). Incorporates a time-synchronization algorithm to mitigate issues with asynchronous cameras.	EasyMoCap Dataset

Table 2.7: 2023 Paper's Content Assessment and Analysis

Papers	Research Purpose	Approach/Domain	Datasets
[20]	Proposes FasterVoxelPose+, a voxel-based method for real-time multi-person multi-view 3D pose estimation. It addresses depth ambiguity and performance degradation with fewer cameras by introducing Depth-wise Projection Decay (DPD).	Uses DPD to add depth information to voxel features, combined with an Encoder-Decoder Network (EDN) for feature processing. The method is real-time and optimized for efficiency.	Campus, Shelf, CMU Panoptic
[19]	Proposes a 3D Associative Embedding approach to improve 3D multi-view HPE in crowded scenes, reducing interference from other people's joints and improving estimation accuracy using a coarse-to-fine framework.	Utilizes a volumetric paradigm with a Voxel Hourglass Network to predict 3D heatmaps and tagmaps. Implements 3D Associative Embedding to distinguish target joints from others' joints, combined with a refinement layer to enhance accuracy.	CMU Panoptic
[9]	Proposes a real-time system for multi-person 3D pose estimation and live streaming of 3D animation using multiple RGB-D cameras and edge devices for distributed processing.	Combines 2D pose detection and depth sensing on edge devices with multi-view triangulation using Direct Linear Transform (DLT) on a central server. Clusters poses based on depth data to reduce occlusion errors.	-
[16]	Proposes E3Pose, an energy-efficient, edge-assisted multi-camera system for real-time multi-human 3D pose estimation, aiming to minimize energy consumption while maintaining high accuracy.	Employs adaptive camera selection based on energy states and occlusion levels using an attention-based Long Short Term Memory (LSTM) for occlusion prediction and a Lyapunov optimization framework for scheduling.	Shelf, Panoptic

Table 2.8: 2024 Paper's Content Assessment and Analysis

Papers	Research Purpose	Approach/Domain	Datasets
[14]	Proposes a real-time multi-view multi-person pose estimation system that integrates a high-synchronization hardware acquisition system with a distributed computing platform for robust and real-time estimation in various environments.	Combines hardware and software components, leveraging distributed computing for 2D keypoint detection and 3D pose reconstruction with optimizations for real-time performance.	3DMPB, CMU Panoptic, Human3.6M
[8]	Proposes a noise-robust 3D pose estimation framework using appearance similarity and geometrical affinity to address calibration errors in multi-camera setups, ensuring accuracy in noisy and real-world environments.	Combines geometrical affinity and appearance similarity for matching 2D poses across views, enabling self-correction of camera calibration errors. Includes edge devices for pose detection and a central server for 3D reconstruction.	Campus, Shelf
[2]	Proposes BeFine, a real-time multi-camera 3D HPE system tailored for industrial applications. Focuses on achieving high scalability and accuracy while addressing network bandwidth, synchronization, and occlusion challenges.	Implements distributed processing with 3D pose estimation performed on edge devices using lightweight models. A central server synchronizes and aggregates data for multi-view triangulation. The system includes two levels of synchronization to mitigate network delays and occlusions.	-

2.5 Summary

This chapter presents a comprehensive overview of the literature review undertaken, establishing the foundation for the state-of-the-art depicted in the subsequent chapter. This Systematic Literature Review on 3D Human Pose Estimation provides a review of the current methodologies employed for the accurate reconstruction of human poses in three-dimensional space from multi-view systems. The PRISMA framework is utilised to systematically identify, evaluate, and synthesise relevant studies in this research domain. This review has identified challenges related to occlusion, ambiguous poses, and the generalisation across diverse datasets. This review aims to provide a comprehensive insight into the current state-of-the-art and future directions for 3D HPE research.

Chapter 3

State-of-the-Art

This chapter offers a comprehensive overview of the current state-of-the-art of 3D HPE approaches, datasets, and evaluation metrics used.

3.1 Datasets

The accessibility of high-quality datasets has been a crucial factor in the advancement of research in the field of 3D HPE. This section presents a review of the key datasets commonly used in multi-camera, multi-person pose estimation. The CMU Panoptic, Campus and Shelf datasets. While Human3.6M is also frequently employed in multi-view settings, its primary focus is on single-person environments.

CMU Panoptic represents one of the most comprehensive resources available for 3D pose estimation. It features a dome-shaped studio equipped with 480 VGA, 31 HD and 10 Kinetic cameras, which captures a wide range of multi-person activities, such as group discussions and dancing, in a controlled environment. This dataset includes 3D ground truth annotations derived from marker-based motion capture systems.

Campus comprises footage captured by three outdoor cameras, which record individuals engaged in various activities and interactions within the campus environment. In contrast to indoor datasets, this outdoor setting introduces additional variables, such as variable lighting conditions. For cameras 1 and 2, annotations were made for the three people's body joints. The annotations for the third perspective were then created by triangulating and projecting these annotations.

Shelf consists of synchronised videos captured from five cameras positioned in an indoor environment where people were disassembling a shelf. The footage features four people disassembling a shelf with severe occlusions, making it an ideal test case for algorithms designed to handle occlusions in cluttered environments. Triangulating the body joint annotations from the camera views 2^{nd}, 3^{rd}, and 4^{th} produced the 3D ground truth annotations.

3.2 Evaluation Metrics

The evaluation metrics unique to 3D HPE are presented in this section. The Percentage of Correct Parts (PCP), Mean Per Joint Position Error (MPJPE), and Average Precision (AP) are the most widely used metrics to assess the quality of the 3D pose estimations. In addition to the aforementioned specific metrics, other metrics, such as Frames per Second (FPS) and Inference Time, are employed for the evaluation of the methods' computational efficiency.

PCP determines the limb detection accuracy. If the difference between the ground-truth limb joint positions and the predicted limb ends is less than or equal to a specific value, α, of the limb's length, then the limb is deemed correct (Eq. 3.1).

$$\frac{\|s_p - s'_p\| + \|e_p - e'_p\|}{2} \leq \alpha \|s_p - e_p\| \tag{3.1}$$

s_p, e_p are the 3D ground-truth coordinates of the beginning and ending points of the body part p.

s'_p, e'_p are the estimated 3D coordinates of the beginning and ending points of the body part p.

α is the threshold.

MPJPE is the average euclidean distance between the estimated joints and the respective ground-truth (Eq. 3.2).

$$\text{MPJPE}(S) = \frac{1}{N_S} \sum_{i=1}^{N_S} \|G_i - P_i\|_2 \tag{3.2}$$

G_i is the ground-truth position of the i-the joint.

P_i is the predicted position of the i-the joint.

N_S is the number of joints of the S skeleton.

AP is computed as the thresholding metric between the predicted joint and the ground-truth using MPJPE (Eq. 3.2).

$$AP@k = \frac{1}{P} \sum_{p=1}^{P} \frac{TP_p}{TP_p + FP_p} \tag{3.3}$$

TP corresponds to True Positive.

FP corresponds to False Positive.

P is the number of people in the scene.

k is the threshold in *mm*.

3.3 3D Human Pose Estimation

3D HPE in multi-view, multi-person environments has become a central topic in computer vision due to its applications in sports analytics, human-computer interaction, surveillance, and augmented reality. The task involves reconstructing 3D human joint coordinates from multiple 2D views captured by spatially distributed cameras. This process becomes particularly challenging

due to factors such as occlusions, inter-person interactions, and dynamic environmental conditions. Several approaches have been proposed in the literature, categorized into three primary methodologies: geometric-based methods, voxel-based representations, and plane-sweep stereo techniques.

3.3.1 Geometric Approaches

In a recent publication, Dong et al. [5] put forth a novel methodology for rapid and reliable multi-person 3D pose estimation. The paper's pipeline commences with the utilisation of an off-the-shelf 2D pose detector, which is employed to obtain the bounding boxes and keypoint locations of individuals in each view. Taking into account the noise and incompleteness of the 2D detections, the authors employ a multi-way matching algorithm based on convex optimisation to identify cycle-consistent correspondences of bounding boxes across multiple views. The algorithm exploits both geometric consistency and appearance similarity. Subsequently, a 3D Pictorial Structure (3DPS) model is employed to reconstruct the 3D pose of each individual person from the corresponding 2D poses. The proposed method demonstrates superior performance compared to previous 3DPS-based methods, achieving state-of-the-art results on the Campus and Shelf datasets. Furthermore, the authors propose an efficient tracking method for the 3D poses that are detected throughout the multi-view video.

Hwang and Kim [8] proposed a framework for estimating the 3D pose of multiple people in a noise-robust way by exploiting appearance similarity based on multiple distributed views. The framework comprises multiple edge AI devices and a central server. The function of each edge AI device is to detect 2D poses using OpenPose and to extract appearance features from RGB images. In particular, the edge AI devices gather RGB colour samples between the 2D coordinates of the body joints, thereby obtaining a concise representation of the individual's appearance. Subsequently, the 2D pose data and appearance characteristics are transmitted to the central server. On the central server, the 2D poses are grouped using both geometric affinity, based on an iterative greedy matching algorithm, and appearance similarity. The central server then compares these two groups in order to identify any potential camera calibration errors. In the event of an error being identified, the central server corrects it by utilising the reconstructed 3D skeletons and the erroneous 2D poses. The proposed framework has been evaluated on the Campus and Shelf datasets, demonstrating resilience against geometric errors and achieving enhancements in 3D pose estimation accuracy. The utilisation of appearance similarity to create a noise-robust system capable of correcting camera calibration errors in real time made it stand out.

3.3.2 Voxel-based Approaches

Tu, Wang, and Zeng [13] introduced VoxelPose (Figure 3.1), a novel approach to estimating three-dimensional human pose that circumvents the complex problem of two-dimensional association by performing association directly in three-dimensional space. The VoxelPose methodology entails the projection of 2D pose heatmaps from a multitude of camera views onto a unified 3D feature

volume. Subsequently, the volume is, firstly, processed by a Cuboid Proposal Network (CPN), followed by a Pose Regression Network (PRN). The CPN is tasked with localising instances of people within the 3D volume, producing cuboid proposals that encapsulate each individual. Subsequently, the PRN refines these proposals, estimating a complete 3D pose for each person. The method was evaluated on three datasets that are commonly employed for multi-person pose estimation: The datasets used were the Campus, Shelf, and CMU Panoptic. VoxelPose exhibited state-of-the-art performance, surpassing existing methods on the Campus and Shelf datasets and demonstrating encouraging outcomes on the challenging CMU Panoptic dataset. One of the principal advantages of VoxelPose is its resilience to occlusion. By performing 3D association, the method is able to effectively aggregate information from multiple camera views, even when some joints are occluded in one or more views.

Choudhury, Kitani, and Jeni [4] put forth a novel approach, termed TEMPO, which offers a highly efficient methodology for multiview pose estimation, tracking, and prediction. The method learns a spatio-temporal representation, thereby improving the accuracy of pose estimation while maintaining speed during inference. The limitations of existing volumetric methods are addressed by TEMPO, which is more accurate than existing methods but less computationally expensive and optimised for single time step prediction. The model's architecture comprises three stages. Initially, the location of each individual within the scene is identified through the projection of image features derived from each viewpoint onto a unified three-dimensional volume. Subsequently, the 3D bounding boxes centred on each person are returned, and tracking is performed, whereby the centres of the boxes are matched with the detections from the previous time step. Ultimately, for each individual identified, a representation of the spatio-temporal pose is calculated through the recursive combination of the characteristics observed at the current and previous time points. This representation is then decoded into an estimate of the current pose, as well as poses in future time steps. This approach allows the model to utilise the spatio-temporal context to predict more accurate human poses without sacrificing efficiency. The efficacy of this approach was evaluated on several pose estimation benchmarks, including CMU Panoptic Studio, Human3.6M, Campus, Shelf and a new multiview dataset composed of highly dynamic scenes, EgoHumans. It achieved cutting-edge results, notably a 10% better MPJPE with a 33x improvement in FPS compared to TesseTrack on the CMU Panoptic Studio dataset. TEMPO also demonstrated the ability to generalise between datasets without scene-specific fine-tuning.

In their study, Ye et al. [15] presented Faster VoxelPose, an extension of VoxelPose [13] designed to facilitate faster inference without compromising accuracy. Faster VoxelPose employs a two-stage architectural configuration comprising a Human Detection Network (HDN) and a Joint Localisation Network (JLN). The HDN operates on a three-dimensional feature volume derived from projected two-dimensional pose heatmaps and is responsible for localising instances of people in three-dimensional space. Subsequently, the Joint Localisation Network (JLN) refines the joint localisation using features extracted from the Human Detection Network (HDN). A significant advancement in Faster VoxelPose is the utilisation of feature volume re-projection, which replaces the computationally demanding 3D convolutions. This modification results in a signifi-

cant reduction in inference time, with Faster VoxelPose approximately 10x faster than the original VoxelPose [13], particularly on larger scenes. Faster VoxelPose exhibits comparable performance to VoxelPose [13], with a mean per joint error (MPJPE) of 18.26 mm on the CMU Panoptic dataset, while maintaining an impressive 30.5 FPS.

Zhang et al. [17] proposed VoxelTrack, an innovative approach for estimating and tracking multi-person 3D poses in challenging scenarios. This method takes advantage of the dispersion of 3D representations to enhance inference speed. VoxelTrack builds upon the VoxelPose [13] framework by incorporating a temporal component and addresses the challenge of estimating and tracking 3D poses in a unified manner within a single frame. The methodology is based on a multi-branch network that jointly estimates three-dimensional poses and re-identification (Re-ID) features for all individuals present in the scene. The Re-ID features are of great importance for the association of 3D poses over time, thus enabling precise tracking. A significant advantage of VoxelTrack is its capacity to perform tracking directly in three-dimensional space. By circumventing the issues inherent to 2D association, such as ambiguous matching and occlusion, VoxelTrack attains remarkable robustness to occlusion. VoxelTrack exhibits superior performance on benchmark datasets, surpassing existing methods by a notable margin. In particular, it attains a percentage of correct parts (PCP3D) of 96.7% on the Campus dataset and 97.1% on the Shelf dataset. The method operates at 15 FPS with 5 camera views as input on a single 2080Ti GPU, rendering it well-suited for diverse real-world applications.

Zhu et al. [19] addressed the challenges of 3D HPE in crowded scenes, where occlusion and the overlapping of multiple people pose significant difficulties, by introducing an approach called 3D Associative Embedding. This novel approach integrates the concept of tag-maps, previously utilized in two-dimensional pose estimation, into the three-dimensional domain. The methodology employs a Voxel Hourglass Network (VHN), which jointly predicts both 3D heatmaps and 3D tag-maps. The introduction of 3D tag-maps is crucial for the distinction of joints belonging to different individuals, effectively mitigating interference from joints of nearby people. This capability is of particular importance in crowded scenes, where traditional methods are unable to correctly differentiate and associate joints with corresponding individuals. In addition to their new embedding strategy, Zhu et al. [19] propose a three-stage structure comprising coarse-to-fine stages, designed to deal with quantisation errors inherent in discretising 3D space into voxels. This progressive structure reduces the size of the search space at each stage while increasing resolution, thereby facilitating more accurate 3D pose estimates. The 3D Associative Embedding method has been demonstrated to outperform state-of-the-art methods on the CMU Panoptic dataset, particularly in crowded scenes where it has been shown to significantly outperform top-down methods.

In order to address the issues of accuracy drops and slow inference speed, as outlined in references [13, 15], Zhuang and Zhou [20] have proposed FasterVoxelPose+, an innovative technique designed to enhance the precision and efficiency of voxel-based 3D HPE, particularly in contexts with a restricted number of cameras. The cornerstone of their approach lies in the proposed Depth-wise Projection Decay (DPD), a new projection method designed to build voxel features by incorporating depth information during the projection process. This incorporation of depth information

is crucial for mitigating the depth ambiguity inherent in projecting 2D heat maps into 3D space, especially when only a limited number of camera views are available. In addition to DPD, Zhuang and Zhou [20] propose a powerful but efficient Encoder-Decoder Network (EDN) for processing 2D features. The EDN incorporates inverse bottleneck blocks, known for their computational efficiency, parallel decoders for effective fusion of multi-scale information and large convolution kernels to increase the network's receiver field. These architectural choices collectively contribute to the improved performance and efficiency of FasterVoxelPose+. This fact is evidenced by the promising results on benchmark datasets, achieving 96.4% and 97.7% PCP3D on the Campus and Shelf datasets, respectively. On the CMU Panoptic dataset, the method achieves a MPJPE of 17.42 mm and a speed of 30.5 FPS, highlighting its ability to maintain accuracy while operating in real time. The introduction of DPD is particularly impactful, leading to significant improvements in accuracy when using a smaller number of cameras, making FasterVoxelPose+ a practical solution for real-world scenarios where cost or logistical constraints may limit the number of cameras available.

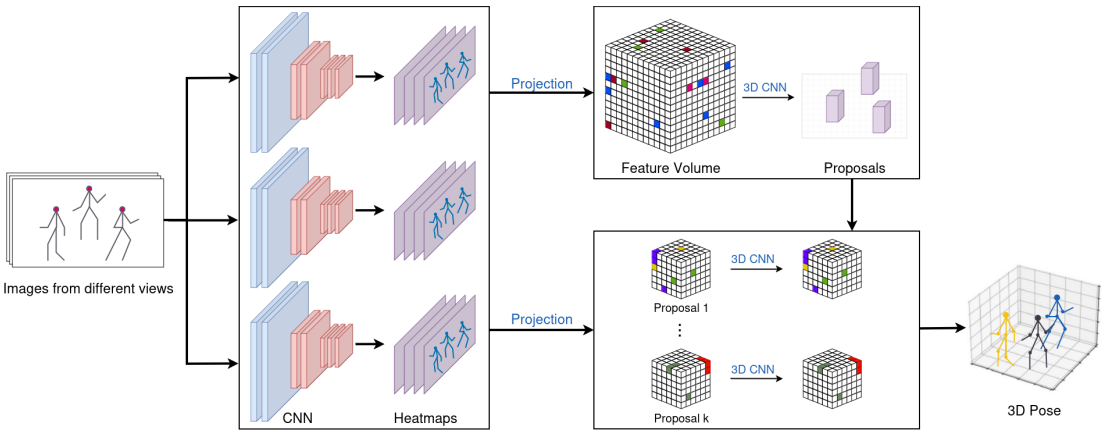

Figure 3.1: VoxelPose Architecture (adapted from [13])

3.3.3 Plane Sweep Stereo-based Approaches

Lin and Lee [11] addressed the computational limitations of VoxelPose [13] by introducing Plane Sweep Stereo (Figure 3.2), an efficient method that exploits the plane sweep algorithm for depth regression. In contrast to the reliance on computationally demanding 3D convolutions, Plane Sweep Stereo employs a regression approach to estimate the depth of each joint for each individual within a specified target camera view. Subsequently, data from multiple reference camera views is aggregated in order to implicitly impose consistency constraints between views using the plane sweep algorithm, thereby facilitating more accurate depth regression. Plane Sweep Stereo employs a two-stage coarse-to-fine architectural approach. In the initial stage, the depth at the person level is estimated. Subsequently, a relative depth estimation at the joint level is performed in relation to the estimated depth at the person level. The final 3D poses are recovered through a direct back-projection procedure utilising the estimated depths. The efficacy of Plane Sweep Stereo

was assessed using benchmark datasets, including Campus and Shelf. It demonstrated superior performance compared to state-of-the-art methods while exhibiting enhanced computational efficiency compared to voxel-based methods. This enhanced efficiency can be attributed to the fact that Plane Sweep Stereo solely processes the target 2D poses and employs 1D convolutions, which are less computationally demanding than 3D convolutions.

Zhou et al. [18] presented the Efficient Recalibration Network (ER-Net), which was developed to enhance the precision of joint localisation in multi-person, multi-view pose estimation tasks, particularly in scenarios characterised by occlusions. A recalibration module based on the channel attention mechanism was introduced to capture long-range dependency relationships between joints. The network employs a coarse-to-fine approach, analogous to that used in Plane Sweep Stereo [11]. Initially, depth is estimated at the person level, and then, through a process of refinement, at the joint level. The innovation resides in its recalibration module, which recalibrates the characteristics of each channel by multiplying them by the learned weight coefficients, which represent the relative importance of each channel. This recalibration enables the network to focus on relevant characteristics, thereby reducing the impact of occlusions and noise. ER-Net was evaluated on reference datasets, Campus and Shelf, and demonstrated improvements in accuracy, particularly for joints affected by occlusions, thereby highlighting the effectiveness of its recalibration strategy.

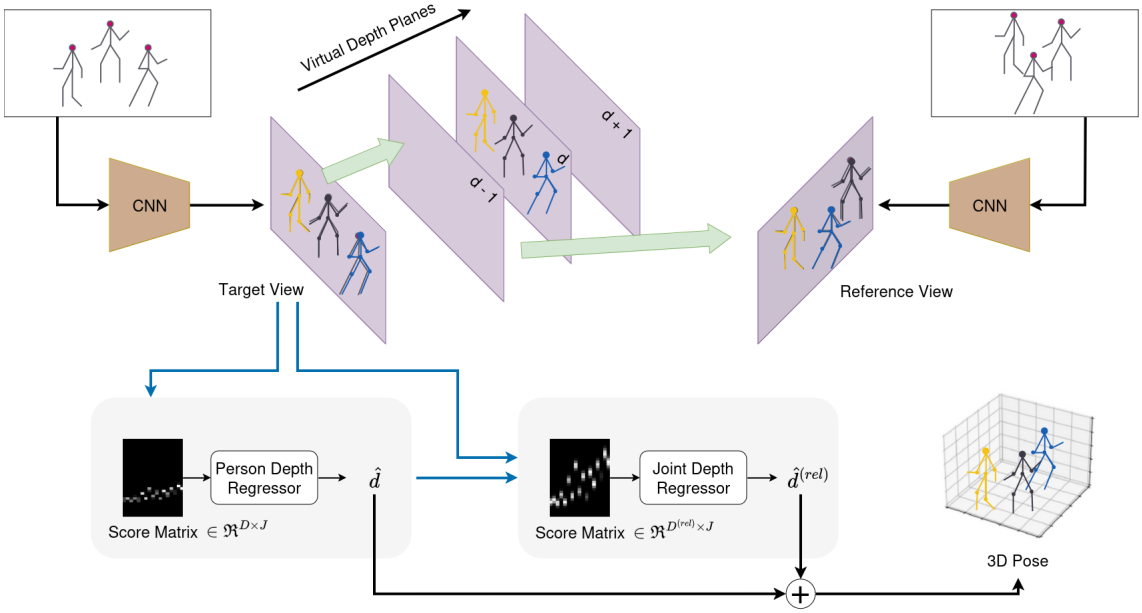

Figure 3.2: Plane Sweep Stereo Architecture (adapted from [11])

3.4 Distributed Systems for Human Pose Estimation

In order to address the challenges of scalability and high inference time of current human pose estimation models, many authors [2, 3, 6–10, 12, 14, 16] approached these problems from a

Table 3.1: Performance of the revised approaches in the Campus and Shelf datasets

Paper	Campus — PCP (%) ↑				Shelf — PCP (%) ↑			
	Actor 1	**Actor 2**	**Actor 3**	**Average**	**Actor 1**	**Actor 2**	**Actor 3**	**Average**
[4]	97.7	**95.5**	97.9	**97.3**	99.0	96.3	98.2	98.0
[5]	97.6	93.3	98.0	96.3	98.8	94.1	97.8	96.9
[8]	83.0	89.0	89.0	87.0	97.0	91.0	94.0	94.0
[11]	98.4	93.7	99.0	97.0	99.3	96.5	98.0	97.9
[13]	97.6	93.8	98.8	96.7	99.3	94.1	97.6	97.0
[15]	96.5	94.1	97.9	96.2	99.4	96.0	97.5	97.6
[16]	-	-	-	-	**99.9**	91.9	**99.4**	97.1
[17]	98.1	93.7	98.3	96.7	98.6	94.9	97.7	97.1
[18]	**98.6**	94.0	**99.2**	**97.3**	99.4	**97.3**	98.2	**98.3**
[20]	97.4	93.6	98.1	96.4	99.0	96.3	97.7	97.7

different perspective. Instead of focusing on the method itself, they developed a whole system for HPE.

Boldo et al. [2] created BeFine, this system uses a distributed approach comprised of edge devices (one per RGB-D camera) and a central computing unit. Each edge device perform 3D HPE, next they send pose data to the central aggregator. This aggregator merges the data through filtering, clustering, and association algorithms. BeFine uses a MQTT-based publisher/subscriber approach for data communication. It also synchronizes data to handle network delays and bandwidth problems by using a two-level synchronization mechanism. The first level synchronizes data from the various cameras while the second level uses a temporal alignment algorithm to aggregate the data.

As a way to solve the referred problems, Bultmann and Behnke [3] solution divides computation between smart edges sensors, which perform image analysis for each camera view, and a backend server. The backend fuses the semantic interpretations, uses a fast skeleton model, and uses timestamps to perform the synchronization of the estimated 2D poses. This system presented a novel 2D/3D feedback architecture that enables bidirectional communication between the smart sensors and backend.

There is a need to properly integrate device-edge co-inference with multi-camera 3D pose estimation due to multiple DNN models and device heterogeneity, so Du et al. [6] introduced a camera-selecting device-edge co-inference framework. The system is composed by edge devices that do some computations, and an edge server. It jointly optimizes model split points and bandwidth with the use of a Random-Ordered Greedy Algorithm (ROGA).This system reduces inference tasks by selecting a subset of cameras.

Table 3.2: Performance of the revised papers in the CMU Panoptic dataset

#Views	Paper	AP_{25} ↑	AP_{50} ↑	AP_{100} ↑	AP_{150} ↑	MPJPE ↓
2	Zhu et al. [19]	25.51	62.03	86.14	92.39	47.42 mm
	Zhuang and Zhou [20]	**31.25**	**65.63**	**93.51**	**96.12**	**42.53 mm**
3	Tu, Wang, and Zeng [13]	**58.94**	**93.88**	**98.45**	**99.32**	**24.29 mm**
	Zhang et al. [17]	49.09	92.44	97.62	-	24.93 mm
	Ye et al. [15]	53.68	91.89	97.40	98.30	26.13 mm
	Zhu et al. [19]	34.98	76.72	93.87	97.24	33.03 mm
	Zhuang and Zhou [20]	57.23	92.21	97.83	98.32	24.98 mm
4	Zhang et al. [17]	66.20	96.34	**99.47**	-	**20.35 mm**
	Ye et al. [15]	73.95	97.02	99.21	99.35	21.12 mm
	Zhu et al. [19]	52.84	90.80	97.85	98.88	23.89 mm
	Zhuang and Zhou [20]	**75.92**	**97.86**	99.32	**99.40**	20.95 mm
5	Choudhury, Kitani, and Jeni [4]	89.01	**99.08**	99.76	**99.93**	**14.68 mm**
	Tu, Wang, and Zeng [13]	83.59	98.33	99.76	99.91	17.68 mm
	Lin and Lee [11]	**92.12**	98.96	**99.8**1	99.84	16.75 mm
	Zhang et al. [17]	85.88	98.31	99.54	-	16.97 mm
	Ye et al. [15]	85.22	98.08	99.32	99.48	18.26 mm
	Zhu et al. [19]	61.28	95.10	99.39	99.87	18.88 mm
	Zhuang and Zhou [20]	86.25	98.45	99.77	99.82	17.42 mm

Hwang and Kim [8] system consists of multiple edge AI devices that detect 2D poses and extract appearance features. This information is sent to a central server that groups the 2D human poses using geometric affinity (greedy matching algorithm) and appearance similarity (clothing color). It utilizes a Perspective-n-Point (Pnp) solving algorithm to estimate the rotation matrix.

Following the same base architecture of other distributed systems [2, 3, 6–10, 12, 14, 16], Hwang, Kim, and Kim [7] and Kim and Hwang [10] presented similar distributed systems composed by multiple edge-devices that perform some computations and a central server that synchronizes and aggregates all the data.

Hwang et al. [9] proposed a similar system as others [2, 3, 6–10, 12, 14, 16], but uses RGB-D cameras instead of RGB.

Liu et al. [12] addresses the challenge of multi-camera calibration, proposing an automatic method that uses human joints as a calibration pattern, eliminating the need for specific calibration objects in denied environments. The system starts with binocular camera calibration, followed by joint optimization. The multi-camera calibration happens in three steps: parameter initialization, extrinsic parameter optimization, and joint optimization of intrinsic and extrinsic parameters. This system utilizes RANSAC algorithm to reduce outliers.

Yang and Wang [14] focused in building a flexible multi-view hardware data acquisition system and a distributed computing platform.

Existing approaches do not consider energy constraints, particularly for battery-powered cameras. With the goal to reduce the energy usage and extend the battery life of the system, Zhang and Xu [16] presented E3Pose, an energy-efficient edge-assisted multi-camera system for real-time multi-human 3D pose estimation. The system uses an attention-based LSTM (Long Short-Term Memory) network to predict the occlusion information of each camera view and a camera selection algorithm based on the Lyapunov optimisation framework to make long-term adaptive camera selection decisions. E3Pose was implemented on a 5-camera testbed consisting of an NVIDIA Jetson TX2 edge server and NVIDIA Jetson Xavier NX camera devices. The evaluation was conducted using the Shelf and CMU Panoptic datasets. The results show that E3Pose achieves significant energy savings while maintaining high 3D pose estimation accuracy comparable to state-of-the-art methods. The authors innovated by introducing an adaptive camera selection scheme that considers view quality and camera power states for an energy-efficient 3D pose estimation system.

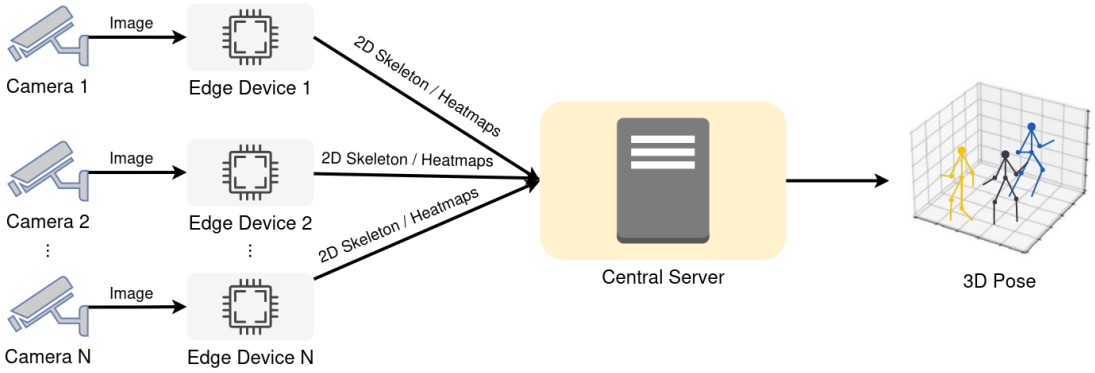

Figure 3.3: Distributed Systems Overall Architecture

3.5 Summary

The reviewed methodologies highlight three distinct approaches to 3D HPE in multi-view, multi-person environments. Geometric approaches excel in calibrated settings, voxel-based methods demonstrate robustness against occlusions and tracking challenges, while plane-sweep methods

balance computational efficiency and depth accuracy. Table 3.1 and Table 3.2 show the performance of highlighted methods in the three most used Datasets.

Distributed systems for human 3D pose estimation offer several advantages over traditional centralised approaches, mainly by leveraging edge computing and parallel processing. These systems are able to minimise latency by processing data in closer proximity to the source, a factor that is of particular importance in the context of real-time applications. Rather than transmitting raw video data to a central server, edge devices extract 2D poses and transmit only that data, thereby reducing network load and delays.

Chapter 4

Distributed System

The present chapter delineates the experimental configuration that has been developed for the purpose of evaluating the proposed 3D HPE system within a distributed environment. The components that constitute the system, the models employed, the selected datasets, and the metrics defined to analyse the quality of the estimation and the viability of the system in real-time are described in detail.

In Section 4.1, the architecture of the distributed system is introduced, including a description of the computational resources utilised. These resources encompass framework versions, such as PyTorch and CUDA, as well as the distribution of GPUs between the central server and the edge devices. In the subsequent Section 4.2, the estimation models employed are presented, with particular emphasis on FasterVoxelPose, a model that is favoured over other contemporary approaches, and a discussion of possible experiments with alternative models.

Section 4.3 focuses on the datasets used for training and evaluation, providing a justification for the selection of each dataset based on their characteristics, applicability and relevant differences between them. Finally, Section 4.4 provides a detailed description of the metrics utilised in the evaluation, encompassing the quality of the 3D pose estimation and the computational efficiency of the distributed system. This includes vital indicators such as inference time, which are instrumental in analysing the viability of the system in real-time contexts.

4.1 Developed System

The development of an accurate 3D HPE model that can operate in real-time is a complex undertaking. A comprehensive review of the extant literature and subsequent analysis of state-of-the-art systems has revealed that current distributed systems do not employ the most advanced pose estimation models. In order to address this gap, a 3D HPE distributed system with multiple cameras and a central server is proposed.

4.1.1 System Architecture

The proposed system for distributed 3D HPE, Figure 4.1, consists of a central server and a network of edge devices, with each edge device associated with a dedicated camera. The architecture emphasises real-time processing, scalability, and modularity, supporting both live camera feeds and pre-recorded datasets. The primary objective of the system is to offload computationally intensive tasks appropriately between the edge and central components to achieve low-latency and high-accuracy pose estimation.

Each edge device is responsible for capturing video frames and computing the corresponding 2D joint heatmaps locally. These heatmaps encode the likelihood of body joint locations in the image plane and serve as the essential input to the central server, where the 3D pose estimation is performed using the FasterVoxelPose model. This model takes the synchronized multi-view heatmaps and computes the 3D position of the body joints by aggregating spatial information across camera views.

To ensure time consistency across the various cameras a Network Time Protocol (NTP) server is set locally. This server Internet Protocol (IP) is set in the cameras' network settings at startup.

4.1.1.1 Communication Design

Efficient and reliable communication between edge devices and the central server is critical for the system's real-time performance. To achieve this, the system employs a custom communication protocol implemented over the User Datagram Protocol (UDP), a transport layer protocol that enables fast, connectionless communication between devices on a network, chosen for its low-latency characteristics and tolerance of occasional packet loss. This makes it particularly suitable for time-sensitive applications where dropping a frame is less critical than incurring delay.

At the core of this communication layer is the Packet class, which encapsulates the logic for serializing, parsing, and validating the various types of messages exchanged in the system. This abstraction allows consistent and extensible message handling while ensuring minimal overhead. Four main packet types are defined: the frame packet, the configuration packet, the synchronization packet and the connection packet. The frame packet contains the low-resolution image (240x128 pixels with low-quality compression, used mainly for visualization), the associated heatmap, a timestamp, and a sequence number. The configuration packet carries the camera calibration parameters, including both intrinsic and extrinsic values, which are essential for accurate 3D pose triangulation. The connection packet facilitates the initial handshake and registration process between the edge device and the server at startup.

4.1.1.2 Frame Synchronization Mechanism

Accurate 3D pose reconstruction requires temporally aligned multi-view heatmaps. To address this requirement, the system includes a robust frame synchronization mechanism implemented at the server side. This mechanism operates over per-device frame buffers, each storing recent frames received from the respective edge device. When synchronized frames are needed for inference, the

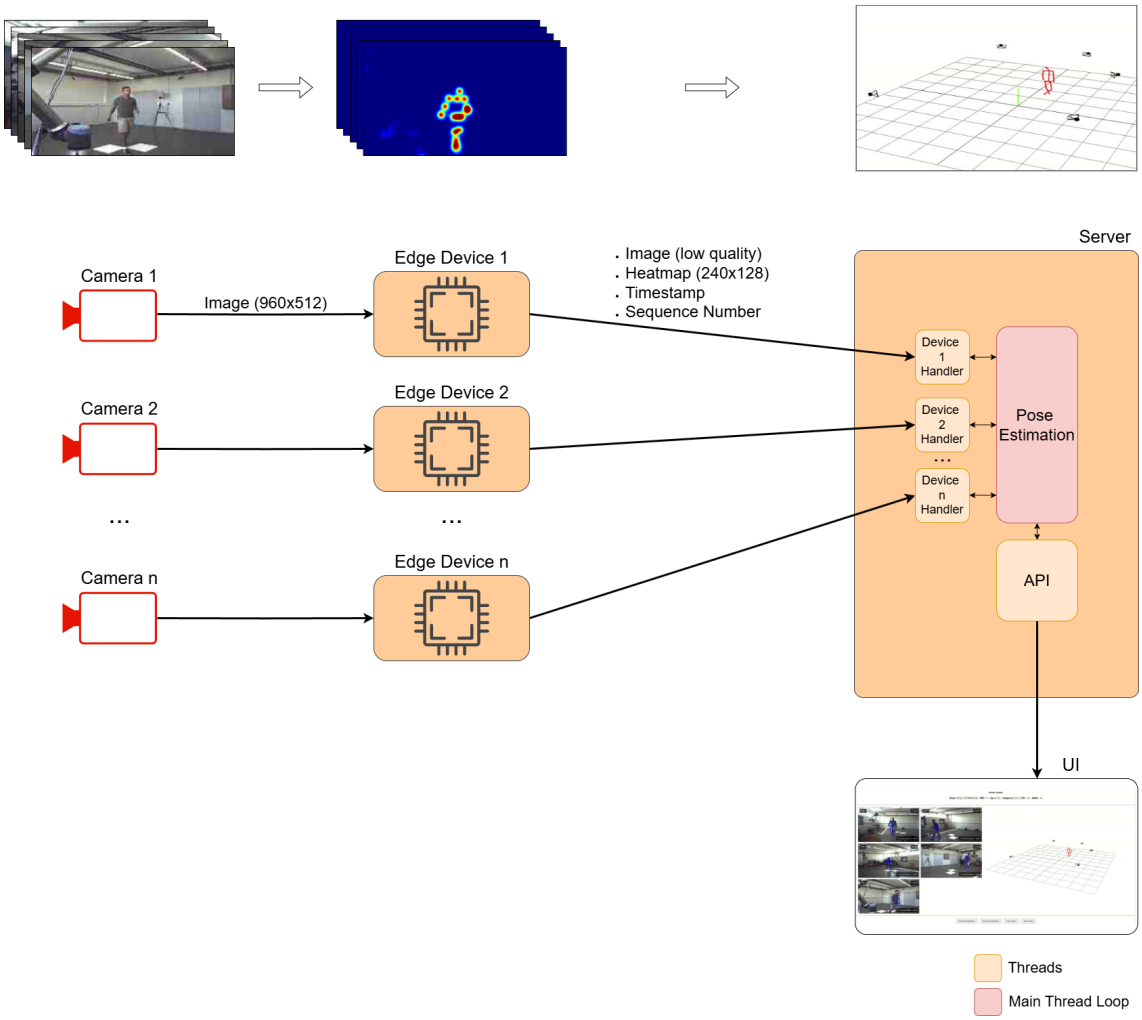

Figure 4.1: Propose System Architecture

server invokes a synchronization function (see Figure 4.2) that identifies the best-aligned group of frames based on either timestamp or sequence number, depending on the option selected by the user. Normally, sequence numbers are used when the input is dataset images, while timestamps are preferred for live feeds. When using the timestamp for synchronization, a synchronization window - the difference in milliseconds between the oldest and newest timestamps - must be defined. For example, if the server is running at 30 Frames Per Second (FPS) the synchronization window must be at most 33.33ms to ensure that all camera frames correspond to the same instant. The synchronization window is 1/FPS.

This synchronization method is robust to moderate temporal drift between devices and ensures that 3D pose estimations are performed only when frames are sufficiently aligned in time or order.

4.1.1.3 Edge Device

Upon initialization, each edge device is configured through a set of user-defined parameters. These include the IP address of the associated camera or dataset path, the IP address and port of the edge

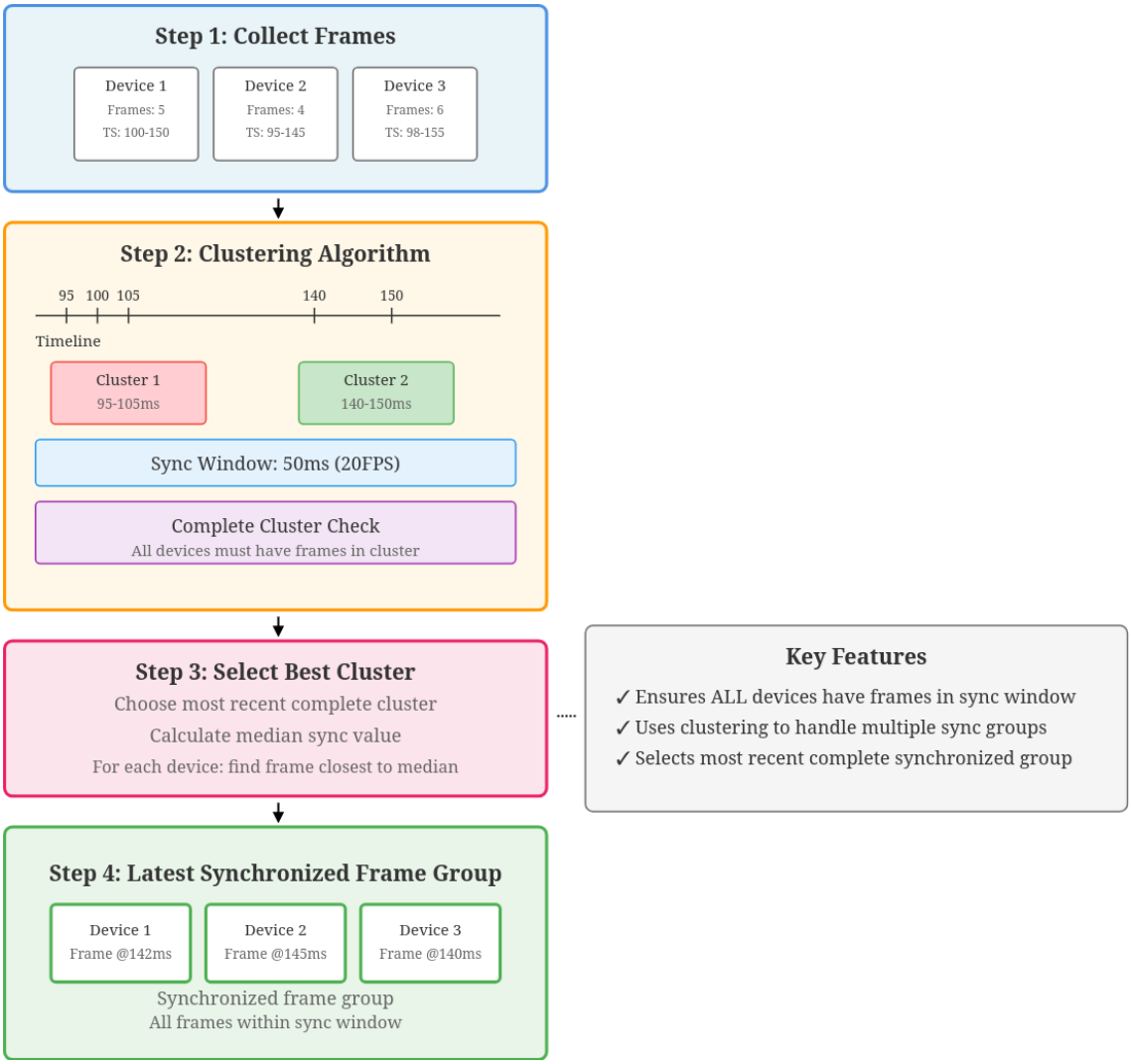

Figure 4.2: Get Synchronized Frames Function Diagram

device itself and the processing mode (camera or dataset), as shown in Listing 4.1. Having these configurable parameters ensures flexible deployment and facilitates coordination in multi-camera systems.

Listing 4.1: Example of Edge Device 1 run command

```
1 python edge_device.py --ip 127.0.0.1 --port 5005 --id 192.168.221.11 --type camera
```

Following initialization, the edge device attempts to connect to its corresponding camera, where it uploads a defined configuration to the camera and downloads the camera calibration parameters. If this connection is successful, the edge device proceeds to wait for a connection from the central server. Once connected to the server, the device uploads its configuration metadata and the previously obtained camera calibration parameters, which are subsequently used by the server

to configure the 3D pose estimation pipeline. The Figure 4.3 represents these execution flow.

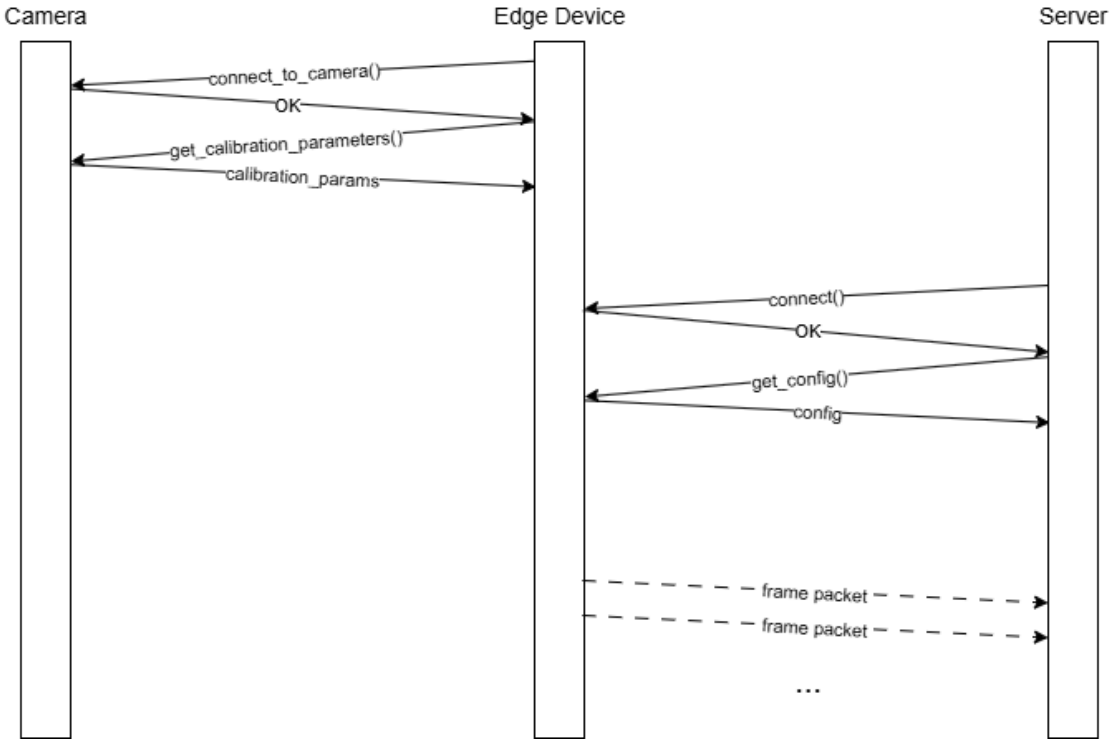

Figure 4.3: Edge Device Initialization

To support development, evaluation, and debugging, edge devices are designed to operate in two modes: real-time streaming camera or pre-recorded datasets. In dataset mode, each edge device loads the images from the correspondent view mimicking the behaviour of a live camera, maintaining the same packet generation and transmission logic. This design allows for consistent testing without requiring changes to the rest of the system, facilitating both reproducibility and regression testing. In addition, all events are logged to provide traceability and facilitate debugging.

4.1.1.4 Central Server

The central server orchestrates the overall operation of the distributed system. At startup, it loads a configuration file that specifies key runtime parameters, including the network addresses of the edge devices, the communication protocol (in this case, UDP), the target frame rate in FPS, and the path for the pose estimation model configuration file, an example of this file is shown in Listing 4.2. The server configuration file enables users to customize the system behaviour without modifying the source code, providing a flexible and user-friendly means of adapting the server to different deployment scenarios or testing environments.

Listing 4.2: Example of Server Configuration File

```
1 edge_devices:
```

```
  protocol: udp
  ips:
    CAM 1: "127.0.0.1:5005"
    CAM 2: "127.0.0.1:5006"
    CAM 3: "127.0.0.1:5007"
    CAM 4: "127.0.0.1:5008"
    CAM 5: "127.0.0.1:5009"
    FPS: 30
pose_estimation_model: FasterVoxelPose
pose_estimation_model_config_file: "./models/Faster-VoxelPose/demo/config.yaml"
```

Using this configuration, the server attempts to establish connections with all specified edge devices. Upon successful connection, it retrieves and stores the camera calibration parameters from each device, which are subsequently loaded into the pose estimation model. This calibration synchronization is repeated whenever a new connection or disconnection event occurs, ensuring that the server maintains an up-to-date view of the active system configuration.

The server is also designed to handle disconnections and reconnections of edge devices gracefully. In the event of a temporary communication failure, the server continues to operate and attempts to restore lost connections, enabling robust and uninterrupted system functionality. This resilience to dynamic network conditions makes the system well-suited for deployment in real-world environments, where device availability and connectivity may vary over time.

When all edge devices are connected and synchronized, the server starts estimating the 3D poses using the defined model, in this case, FasterVoxelPose.

In addition to real-time inference, the server includes comprehensive logging mechanisms. All events — ranging from successful inferences to connection errors — are logged to provide traceability and facilitate debugging. Each inference result is stored in a structured JSON Lines (.jsonl) file, with entries that include the timestamp, sequence number, inference time, and the estimated 3D joint positions. This persistent log serves as a valuable dataset for system evaluation and downstream tasks such as performance analysis, dataset generation, or benchmarking.

To support integration with external systems, including user interfaces or monitoring tools, the server exposes a lightweight RESTful API implemented using Flask. This API provides endpoints for querying system status, retrieving recent frames or pose estimations, and inspecting device connectivity. The simplicity of this interface ensures that it can be easily incorporated into front-end dashboards or automated pipelines, enhancing the usability and accessibility of the system.

4.1.2 Hardware vs Docker Emulation

To facilitate development, testing, and evaluation of the proposed distributed 3D pose estimation system, the architecture was initially implemented in a fully containerized environment. This approach leveraged Docker and Docker Compose to emulate the behaviour of multiple independent physical edge devices interacting with a central server. The use of containerization provided

several practical benefits, including system reproducibility, modular deployment, simplified orchestration, and efficient resource isolation.

Each emulated edge device was deployed as an independent Docker container, running its own instance of the edge software. These containers simulate the behaviour of actual physical units, including camera interfacing, heatmap extraction, and communication with the central server. Docker Compose was used to coordinate the simultaneous launch and configuration of multiple edge containers alongside the central server container, closely mimicking a real-world multi-device setup.

Crucially, the emulated edge devices were provisioned with resource limits and architectural constraints that approximate those of embedded AI platforms commonly used in state-of-the-art (SOTA) systems, such as the NVIDIA Jetson series. These platforms typically feature low-power ARM CPUs, limited system memory, and embedded GPUs. While the emulated devices operate on x86-based hardware during testing, their computational and memory constraints, along with shared access to a common GPU, were designed to reflect the behaviour and limitations of actual Jetson-class edge devices.

This emulation strategy allows for realistic evaluation of system latency, synchronization robustness, network communication, and inference throughput in a distributed setting—without requiring access to multiple physical embedded devices. The architecture was designed from the outset to be portable across both containerized environments and real hardware, ensuring that the transition from Docker-based emulation to deployment on physical edge devices requires minimal modifications.

In summary, the use of Docker and Docker Compose to emulate distributed edge devices enabled efficient development while preserving fidelity to the hardware constraints and architectural design of embedded, real-world edge systems. This approach provides a practical foundation for subsequent deployment on physical hardware and validates the system's readiness for real-time, multi-camera 3D pose estimation in resource-constrained environments.

4.1.3 Resources

The resources that were used by the system components in order to emulate the closest as possible distinct physical components are the following:

- **Server**
  - CPU 8 Cores, 3.6GHz
  - 32GB RAM
  - GPU rtx2080ti 11GB
- **(Virtual) Edge Device**
  - CPU 2 Cores, 3.6GHz
  - 4GB RAM

  - GPU rtx2080ti 11GB - Restricted to only 1466MiB

- **Camera**
  - Nerian Ruby 3D Depth Camera

- **Network**
  - Switch
  - Ethernet Cables

Figure 4.4 illustrates the schematics of the connections between all the components previously mentioned.

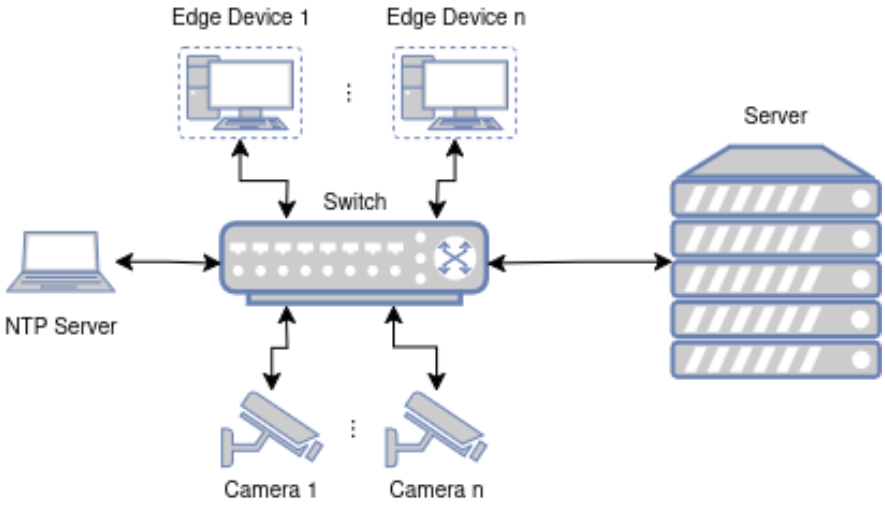

Figure 4.4: Network Diagram

4.2 Pose Estimation Models

The system presented in this work adopts a modular, two-stage approach to 3D HPE, with clear separation between local 2D feature extraction on the edge devices and centralized volumetric 3D inference on the server. The choice of models for each of these stages was guided by both computational efficiency and architectural compatibility with a distributed system design.

For the extraction of 2D joint heatmaps from each camera view, a ResNet-based convolutional neural network was employed (the same model used by Ye et al. [15] for the CMU Panoptic dataset). This model is well-established in the literature for its high accuracy in keypoint detection tasks and strikes a favourable balance between inference speed and prediction robustness. Given its relatively lightweight architecture, ResNet is well-suited for deployment on edge devices or embedded platforms, aligning with the system's goal of offloading initial computation to the camera units.

For centralized 3D pose inference, the FasterVoxelPose model by Ye et al. [15] was selected. This model is a volumetric fusion-based architecture that processes synchronized multi-view 2D heatmaps to estimate 3D joint positions. It includes a human detection module and a joint localization module, allowing it to generate highly accurate full-body 3D pose predictions. A key reason for choosing FasterVoxelPose lies in its compatibility with distributed system design: the model explicitly accepts 2D heatmaps as input, making it possible to decouple the feature extraction stage (on edge devices) from the 3D inference stage (on the server). This separation enables scalable deployment across multiple edge units and minimizes bandwidth requirements, as heatmaps are significantly more compact than raw images or video frames.

In addition to its architectural suitability, FasterVoxelPose demonstrated competitive performance in terms of both inference time and accuracy across different numbers of camera views, and other State-of-the-art (SOTA) metrics.

Alternative models were considered, such as approaches based on Plane Sweep Stereo by Lin and Lee [11], a depth estimation technique commonly used for 3D reconstruction from multiple views. However, such models were ultimately excluded due to several limiting factors. First, Plane Sweep Stereo exhibits a monolithic architecture that tightly couples image processing, depth reasoning, and triangulation, making it significantly more difficult to distribute across heterogeneous devices. Second, it incurs substantial computational overhead and latency, rendering it unsuitable for real-time applications with strict timing requirements. These constraints conflict with the system's design goals of scalability, low-latency inference, and modular deployment.

To facilitate extensibility and future development, the system was implemented in a modular and decoupled manner. Both the 2D heatmap extraction component and the 3D pose estimation module are encapsulated as interchangeable components. This allows researchers and developers to easily integrate alternative models in the future, whether for improving performance, reducing inference cost, or accommodating new hardware capabilities. As more efficient or accurate models are developed in the field of human pose estimation, they can be incorporated into the system with minimal architectural changes.

In summary, the selection of ResNet and FasterVoxelPose reflects a design philosophy grounded in practical deployment considerations, real-time performance, and architectural modularity. This combination allows for accurate and efficient 3D pose estimation while maintaining a flexible and future-proof system structure.

4.3 Datasets

In order to compare the proposed system with other distributed system and models, the CMU Panoptic, Campus and Shelf datasets, referred in Section 3.1, were used. The metrics collected in each of these datasets are presented in Section 4.4.

In addition, in order to assess and test the real-time performance of the system, a live run was performed in a faculty room. The room has a size of 10x10 meters, its layout and camera positions are shown in Figure 4.5.

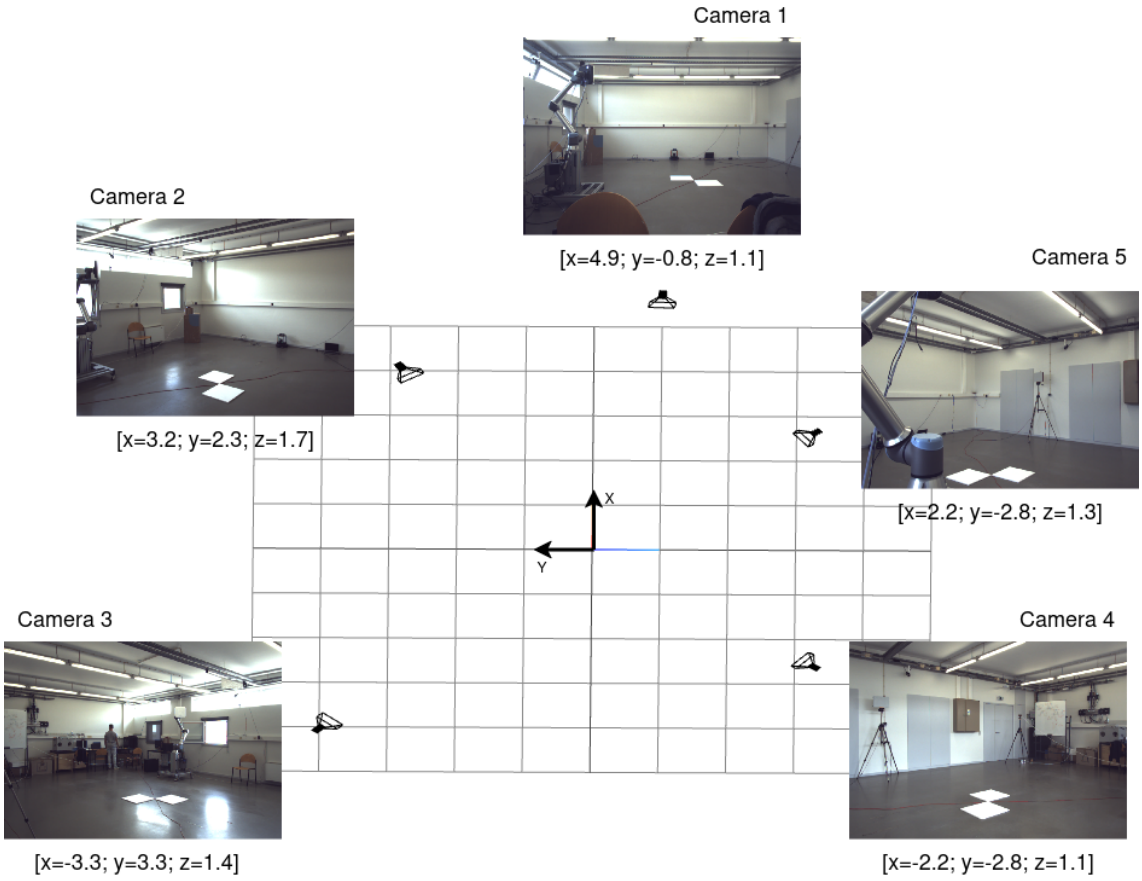

Figure 4.5: Room Layout

Before running the system in this new environment, a camera calibration process was needed. To perform the calibration, a set of steps were followed: Firstly, a set of reference points, visible for all cameras, were chosen, in this case, the 8 corners of 2 foam boards (0.5x0.5 meters) placed in the middle of the room. Secondly, using a self-made calibration program, the point matching between the image coordinates and the real-world coordinates was performed through OpenCV's solvePnP function, which returned the rotation and translation matrices. Finally, the cameras' calibration parameters were passed to the corresponding edge devices.

4.4 System Assessment Criteria

- **Accuracy of 3D HPE**
  - **PCP** (Campus and Shelf)
  - **MPJPE** (CMU Panoptic)
- **Efficiency and Performance**

  - **Frame Rate (FPS)**: Measures the processing speed of the system, often indicating real-time capability.
  - **Inference Time/Runtime (ms)**: Measures the time taken for pose estimation, either per frame set or for 2D/3D pose inference, in milliseconds.
  - **Model Size (MB)**: Refers to the memory footprint of the deployed models, typically in MB or GB.
  - **Workload (%CPU; %GPU)**: Indicates the percentage of CPU and GPU resources utilized by the software.
  - **Memory (MB)/GPU Memory (MB)**: Represents the memory consumption of the system components.
  - **Used Bandwidth (kbps; MB/s)**: Measures the data transmission rate over the network.
  - **Frames/Keypoints Lost due to Time Sync (%)**: Quantifies data loss due to temporal misalignment.
  - **Latency/Pipeline Delay (ms)**: Measures the time delay in the system, including processing, network, and synchronization delays.

- **Robustness**

  - **Robustness to communication issues**: Evaluated by analysing the impact of lower available network bandwidth, significant end-to-end latency, and packet loss rate on accuracy.
  - **Noise robustness**: Evaluated by simulating environments where camera calibration parameters change and assessing the system's ability to maintain accuracy.

Chapter 5

Experiments & Results

This chapter is an exposition of the experiments that were performed and the subsequent analysis of the results obtained. In addition, it provides a comparison with state-of-the-art systems.

In Sections 5.1 and 5.2, a detailed account of two experiments that were carried out, along with the results of these experiments.

In Section 5.3, the overall system performance and its evaluation against other distributed systems and pose estimation models is performed.

5.1 Effect of Varying Camera Number

To evaluate the scalability and performance of the proposed distributed 3D HPE system under varying camera configurations, a series of experiments was conducted using three benchmark datasets — CMU Panoptic, Shelf, and Campus — as well as a custom real-time stream. The system was tested with 3 to 5 cameras to reflect typical multi-view setups used in both academic and real-world environments.

The primary objective of this experiment was to understand how the number of active cameras impacts both individual edge device performance and the central server's workload. The results, presented in Tables 5.1 through 5.4, provide a breakdown of the timings and hardware utilization across system components under different conditions.

Table 5.1: Edge Device Iteration Times (in milliseconds) for each Dataset

Dataset	**Read**	**Transforms**	**Inference**	**ReTransforms**	**Send**	**Total**
CMU Panoptic	15.244	2.440	16.832	0.224	3.178	37.918
Campus	2.042	2.777	16.845	0.224	1.728	23.616
Shelf	11.087	2.458	16.825	0.225	3.479	34.074
Custom (Real-Time)	0.100	2.403	16.788	0.223	2.538	22.052

Table 5.2: Edge Device Resources Usage for each Dataset

Dataset	**CPU (%)**	**RAM (MiB)**	**GPU (%)**	**GPU Mem (MiB)**
CMU Panoptic	107	2550	39	1322
Campus	78	2526	43	1350
Shelf	107	2538	42	1306
Custom (Real-Time)	101	2563	43	1342

Table 5.3: Server Iteration Times (in milliseconds) for each Dataset

#Cameras	**Dataset**	**Read Sync**	**Transforms**	**Inference**	**Total**
5	CMU Panoptic	0.153	3.813	17.077	21.043
	Shelf	0.164	3.536	14.704	18.404
	Custom (Real-Time)	0.167	3.383	14.279	18.832
4	CMU Panoptic	0.128	2.856	16.993	19.997
	Shelf	0.131	2.707	14.915	17.753
	Custom (Real-Time)	0.130	2.750	13.401	16.281
3	CMU Panoptic	0.106	2.043	16.028	18.177
	Campus	0.104	2.022	10.763	12.890
	Shelf	0.103	2.073	16.477	18.653
	Custom (Real-Time)	0.104	2.018	12.330	14.454

Since the edge devices run in parallel and operate independently, the number of active cameras has no impact on the performance of each edge device. For all datasets tested — CMU Panoptic, Campus, Shelf, and a custom real-time stream — the per-frame iteration times on edge devices remain approximately constant across experiments. Notably, variations in the Read time between datasets are a result of the read from disk operation. The lower reading times for Campus derive from its smaller dataset size compared to both CMU Panoptic and Shelf. In offline datasets, frames are read from disk, incurring higher I/O latency. Conversely, the real-time streaming, does not exhibit this disk-read delay.

In contrast, the server-side performance is directly influenced by the number of cameras. For the CMU Panoptic dataset, an increase from 3 to 5 cameras results in a measurable rise in total iteration time, from 18.199 ms to 21.070 ms (Table 5.3). This increase is mainly concentrated in the transforms stage - converts the received heatmaps to the correspondent model input format - which scales with the number of multi-view heatmaps received. Similar behaviour is observed for

Table 5.4: Server Resources Usage for each Dataset

#Cameras	Dataset	CPU (%)	RAM (MiB)	GPU (%)	GPU Mem (MiB)
5	CMU Panoptic	76	3267	33	2725
	Shelf	78	3229	28	2956
	Custom (Real-Time)	86	3355	25	2914
4	CMU Panoptic	73	3170	32	2768
	Shelf	72	3141	28	2770
	Custom (Real-Time)	79	3321	21	2900
3	CMU Panoptic	70	3027	30	2888
	Campus	54	3020	18	2602
	Shelf	69	3058	31	2941
	Custom (Real-Time)	71	3116	20	2898

the other datasets. The number of people is the variable that most influences the inference times, CMU Panoptic is the dataset with higher number followed by Shelf and Campus.

From a resource utilization perspective, server resource consumption stably increases with the number of active cameras, as shown in Table 5.4. This is a result of limiting the server refresh rate at 30FPS. The increasing usage of CPU and RAM matches the increasing in the number of active cameras, since the server has more connections to handle.

In terms of edge device resource usage (Table 5.2), the CPU, GPU, and memory consumption remain relatively stable and independent from the dataset used, reaffirming the modular and isolated design of the edge-side computation pipeline.

These results validate the system's scalable architecture, where edge devices operate asynchronously and independently, and server-side processing scales linearly and predictably with the number of incoming views. This ensures that the system can adapt to deployments with varying numbers of cameras, maintaining efficiency while preserving real-time capabilities in live settings.

5.2 Robustness to Connectivity Issues

In distributed systems for 3D pose estimation, the maintenance of continuous connectivity among edge devices and the central server is of paramount importance for ensuring accurate and real-time processing. However, it is important to acknowledge that network disruptions are an unavoidable aspect of real-world scenarios. In order to evaluate the robustness of the proposed system under

```
2025-06-30 15:56:27,956 - INFO - Server state -> State.INIT
2025-06-30 15:56:27,956 - INFO - Loading model config file
2025-06-30 15:56:38,674 - INFO - Server state -> State.SETUP
2025-06-30 15:56:38,674 - INFO - Starting server
2025-06-30 15:56:38,675 - INFO - Connected to CAM 1 : 127.0.0.1:5005
2025-06-30 15:56:38,676 - INFO - Connected to CAM 2 : 127.0.0.1:5006
2025-06-30 15:56:38,676 - INFO - Connected to CAM 3 : 127.0.0.1:5007
2025-06-30 15:56:38,677 - INFO - Connected to CAM 5 : 127.0.0.1:5009
2025-06-30 15:56:38,677 - INFO - Connected to CAM 4 : 127.0.0.1:5008
2025-06-30 15:56:43,227 - INFO - Server state -> State.POSE_ESTIMATION
2025-06-30 15:56:43,245 - INFO - Inference Time: 17.46 | Sequence Number: 18848 | Cameras: 5
2025-06-30 15:56:43,290 - INFO - Inference Time: 14.48 | Sequence Number: 18849 | Cameras: 5
...
2025-06-30 15:56:48,677 - INFO - Inference Time: 20.30 | Sequence Number: 18874 | Cameras: 5
2025-06-30 15:56:48,750 - INFO - Inference Time: 18.00 | Sequence Number: 18874 | Cameras: 5
2025-06-30 15:56:50,741 - ERROR - CAM 1: Streaming Error
2025-06-30 15:56:50,742 - ERROR - CAM 1: Edge Device Disconnected
2025-06-30 15:56:50,803 - INFO - Inference Time: 19.36 | Sequence Number: 18875 | Cameras: 4
2025-06-30 15:56:50,865 - INFO - Inference Time: 18.77 | Sequence Number: 18875 | Cameras: 4
...
2025-06-30 15:56:53,745 - ERROR - CAM 1: Edge Device Disconnected
2025-06-30 15:56:53,802 - INFO - Inference Time: 19.07 | Sequence Number: 18881 | Cameras: 4
2025-06-30 15:56:53,865 - INFO - Inference Time: 19.65 | Sequence Number: 18882 | Cameras: 4
2025-06-30 15:56:54,745 - ERROR - CAM 1: Edge Device Disconnected
2025-06-30 15:56:54,801 - INFO - Inference Time: 20.11 | Sequence Number: 18883 | Cameras: 4
2025-06-30 15:56:54,864 - INFO - Inference Time: 20.38 | Sequence Number: 18884 | Cameras: 4
2025-06-30 15:56:54,930 - INFO - Inference Time: 21.34 | Sequence Number: 18884 | Cameras: 4
2025-06-30 15:56:56,160 - INFO - Reconnected to CAM 1 : 127.0.0.1:5005
2025-06-30 15:56:56,381 - INFO - Inference Time: 18.45 | Sequence Number: 18980 | Cameras: 5
2025-06-30 15:56:56,454 - INFO - Inference Time: 18.24 | Sequence Number: 18980 | Cameras: 5
2025-06-30 15:56:56,531 - INFO - Inference Time: 20.04 | Sequence Number: 18981 | Cameras: 5
```

Figure 5.1: Server Logs

such conditions, a controlled disconnection experiment was conducted. This experiment involved disconnecting one of the edge devices from the central server manually.

The system incorporates a built-in mechanism to detect and respond to connection losses. This mechanism consists in the monitoring of the timestamp of the most recently received frame from each edge device. Should this timestamp exceed a predefined threshold (1 second), the system identifies the device as disconnected, temporarily removing it from the active devices list used in the 3D pose estimation pipeline. This enables the system to sustain its pose estimation capabilities by leveraging the remaining active devices, ensuring uninterrupted and precise performance.

Concurrently, the system initiates an automated reconnection protocol. The protocol's objective is to re-establish communication with the disconnected edge device at regular intervals, defined by a delta time parameter (1 second). This reconnection process persists until the device successfully rejoins the network. Upon reconnection, the edge device is seamlessly reintegrated into the active device list, resuming its contribution to the overall estimation process as though no disruption had occurred.

The experimental task demonstrated that the disconnection and reconnection procedures do not compromise the synchronization or overall performance of the 3D pose estimation process. The system has been demonstrated to successfully maintain temporal and spatial coherence during edge device failures, thus confirming its resilience and capacity for self-recovery in distributed environments. Figure 5.1 is a snapshot taken from the server logs.

5.3 System Performance

To assess the performance of the proposed distributed 3D pose estimation system, a comprehensive evaluation was conducted, both in isolation and through comparative analysis with state-of-the-art distributed frameworks in similar contexts. Table 5.5 presents the hardware used by the authors.

At the edge device level, the proposed system exhibits robust performance in 2D pose heatmap extraction. Iteration times for the edge devices are measured at 37.918 ms for the CMU Panoptic dataset, 23.616 ms for Campus, and 34.074 ms for Shelf. These durations correspond to effective frame rates ranging from approximately 26.3 FPS (CMU Panoptic) to 42.3 FPS (Campus) per edge device. In comparison, Boldo et al. [2] reported edge node frame rates of approximately 22–24 FPS per camera using Nvidia Jetson Xavier NX boards, while Hwang, Kim, and Kim [7] achieved 15–20 FPS on Jetson TX2 boards. Kim and Hwang [10] observed 19 FPS for Jetson TX2 and 27 FPS for Jetson AGX. These figures suggest that the proposed system's edge-level 2D heatmap generation is highly efficient and capable of matching or surpassing the raw frame rates of several contemporary distributed solutions. Furthermore, Bultmann and Behnke [3] achieved a 2D joint detection rate at 30 Hz (approximately 33.3 ms per frame) on Google Edge TPU Dev Boards, with an inference time of 0.024 s for a set of four images, indicating competitive single-image processing latency for the proposed system. Yang and Wang [14] notably accelerated their YOLO and Alphapose inference times from 57.9 ms to 13.52 ms and 21.6 ms to 7.05 ms respectively through TensorRT optimization, a benchmark that underscores the potential for further optimization within the proposed system's edge inference component.

Resource consumption at the edge devices for the proposed system, across the various datasets, reveals CPU utilization between 78% and 107%, RAM usage ranging from 2526 MiB to 2550 MiB, GPU utilization between 39% and 43%, and GPU memory consumption from 1306 MiB to 1350 MiB. Boldo et al. [2] reported lower CPU workloads (20.9–25.6%) but comparable GPU utilization (37.4%) and slightly higher memory usage (3010–3366 MB).

The central server component of the proposed system demonstrates robust performance in aggregating heatmaps and executing the FasterVoxelPose model for 3D human pose estimation. Server iteration times range from 12.912 ms (Campus dataset with 3 cameras) to 21.070 ms (CMU Panoptic dataset with 5 cameras). These times imply an effective 3D pose reconstruction throughput ranging from approximately 47.5 FPS to 77.4 FPS at the server. It is crucial to note that unlike the central server described by Boldo et al. [2], which primarily acted as a keypoint aggregator achieving very high raw server frame rates (153–233 FPS) without significant GPU utilization for 3D reconstruction (0.0% GPU workload), the proposed system's server actively leverages its GPU for the computationally intensive 3D pose calculation. This architectural distinction signifies a deliberate shifting of the primary computational burden for 3D reconstruction to the central server, leading to a different but optimized resource profile. The overall pipeline delay in distributed systems is a critical factor; Bultmann and Behnke [3] reported an average delay of 89 ms for a 4-camera setup and 200 ms for a 16-camera setup, while Kim and Hwang [10] observed a system delay of approximately 200 ms from scene to Graphical User Interface (GUI). The rapid iteration

times of both edge and server components in the proposed system suggest a competitive overall system delay.

A significant advantage of the proposed distributed architecture is its pronounced reduction in network bandwidth requirements. By transmitting compact 2D pose heatmaps instead of high-bandwidth raw camera imagery, the system substantially alleviates network load. This strategy is consistent with, and potentially builds upon, the bandwidth optimization techniques employed by several state-of-the-art systems. Bultmann and Behnke [3] achieved over 99% bandwidth reduction by transmitting only semantic skeletons, 15 kB/s per detected person versus 27 MB/s for raw VGA images. Hwang, Kim, and Kim [7] demonstrated reduced data traffic to approximately 64 Kbps by sending 2D pose results with timestamps, compared to 3.5 Mbps for raw RGB images. Boldo et al. [2] reported bandwidth usage between 108 Kbps and 1009 Kbps, while Kim and Hwang [10] also emphasized network traffic reduction to hundreds of kilobits per second by sending only 2D pose estimates to the server. Hwang and Kim [8] explicitly state that transmitting minimal data (2D poses plus 48 bits of clothing color) effectively solves bandwidth limitations. While specific bandwidth figures for heatmap transmission are not provided in the current results, the fundamental principle of sending compressed, high-level features like heatmaps inherently offers a substantial reduction in network strain, making the system highly suitable for environments with constrained or variable network capacities.

The scalability and real-time operational capabilities of the proposed system are evidenced by its effective performance with up to 5 cameras. This demonstrates its potential for deployment in multi-view scenarios, aligning with the trends observed in larger-scale systems. Bultmann and Behnke [3] scaled their system up to 16 sensor boards, while Yang and Wang [14] deployed a system with 12 industrial cameras, achieving an average frame rate of 40 FPS for multi-person pose estimation, supported by a custom hardware synchronizer resulting in a synchronization latency rate of only 10^{-8} seconds. Hwang, Kim, and Kim [7] utilized four Jetson TX2 boards to capture a static 20 FPS with a time interval (Δ t) of 20 ms. Hwang et al. [9] validated a system with six NVIDIA Jetson Xavier boards capable of real-time multi-person processing at 15 FPS (with 3 people). Kim and Hwang [10] implemented a demo platform with six edge devices, estimating 3D human poses at a constant 30 FPS in real-time. The consistent frame speed observed in the proposed system's server processing across varying camera counts and datasets suggests a robust and predictable pipeline. This characteristic is a key objective, explicitly addressed by Kim and Hwang [10] in their constant frame speed system. Furthermore, the framework's design inherently supports asynchronous multi-views, akin to the approach by Hwang, Kim, and Kim [7], allowing for more flexible deployment scenarios.

Regarding quantitative accuracy metrics, some presented in Table 5.6, the proposed system's 3D pose estimation performance is considered to be equivalent to that of the FasterVoxelPose model. This implies that the system benefits from the inherent accuracy characteristics of this state-of-the-art model. Contextualizing this against the established benchmarks in the literature: Bultmann and Behnke [3] reported significant MPJPE reductions (e.g., from 32.9 mm to 29.8 mm with semantic feedback) and improved 2D Joint Detection Rate. Hwang and Kim [8] demonstrated

accuracy improvements of approximately 11.5% on the Campus dataset and 8% on the Shelf dataset in simulated noisy environments by correcting camera calibration errors, and achieved PCP scores of 0.88 on both datasets for camera rotation of 3 degrees. Yang and Wang [14] achieved high average precision scores (99.8% AP for single-person, 87.3% AP for double-person, 72.6% AP for triple-person) through a highly synchronized hardware system and optimized algorithms. Liu et al. [12] focused on an automatic multi-camera system calibration method based on human joints, reporting lower average reprojection errors compared to traditional methods (4.78 pixels vs. 5.79 pixels for 4 cameras), which directly impacts 3D pose accuracy. It is important to note that the proposed system, like many other state-of-the-art solutions, relies on pre-calibrated camera setups, whereas Liu et al. [12] present a valuable method for automatic calibration, enhancing deployment flexibility. Boldo et al. [2] reported Mean Average Error (MAE) of 4.5–8.9 cm and Pearson correlation (PCC) of 97.5–98.8% with 2-3 cameras, demonstrating robustness to communication issues. Zhang and Xu [16]'s E3Pose system, while primarily focused on energy efficiency, also reported comparable accuracy with state-of-the-art solutions, outperforming various baselines by 2.8% to 13.1% in PCP on the Shelf dataset and 19.7% to 41.14% in AP on the CMU Panoptic dataset. By leveraging FasterVoxelPose, the proposed system is positioned to offer competitive accuracy in 3D pose estimation, directly comparable to results achieved by other systems that utilize robust 3D reconstruction techniques.

Beyond fundamental performance metrics, the proposed system's architectural design choice of offloading 2D pose heatmap extraction to edge devices and leveraging a central server for FasterVoxelPose-based 3D reconstruction contributes to its distinct characteristics. This contrasts with systems like Du et al. [6]'s Camera-Selecting Device-Edge Collaborative Inference (CDC-MCTPE), which focuses on jointly optimizing model split points and camera selection for minimal latency and energy under accuracy constraints. While not explicitly implementing adaptive camera selection, the proposed system's distributed design inherently supports a flexible number of contributing viewpoints. The emphasis on real-time processing and efficient data transmission aligns with the core objectives of all reviewed state-of-the-art systems.

In comparison to centralized state-of-the-art methods like FasterVoxelPose in its original design, our system introduces a crucial architectural distinction: the division of computation between the edge devices and the server. The traditional FasterVoxelPose pipeline is fully centralized, requiring all video feeds to be processed centrally, leading to higher latency, and centralized GPU load. While the original model demonstrates excellent accuracy and robustness, it scales poorly in large, distributed camera setups due to its dependency on full-resolution synchronized image input and high-end centralized compute resources. In contrast, our system preserves FasterVoxelPose's core 3D regression component but decentralizes the 2D inference stage. This significantly reduces bandwidth usage and distributes compute load, enabling real-time performance even with multiple cameras without sacrificing the geometric fidelity of the 3D output.

In summary, the proposed system offers a scalable, efficient, and real-time alternative to both centralized high-accuracy models and cutting-edge distributed platforms. It uniquely combines the robustness and accuracy of centralized models like FasterVoxelPose with the scalability and

responsiveness of distributed edge-compute architectures. By decoupling 2D and 3D processing, minimizing network load, and maintaining consistent inference times at the edge, the system proves well-suited for real-world deployments in motion capture and surveillance across dynamic environments.

Table 5.5: Distributed Systems Hardware

Paper	**Edge Device**	**Server**	**Cameras**
Boldo et al. [2]	Nvidia Jetson Xavier NX (384 CUDA cores, 8GB RAM)	Desktop PC (RTX 2070 Super, i5 7400, 8GB RAM)	StereoLabs Zed2 RGB-D (2K)
Bultmann and Behnke [3]	Google Edge TPU Dev Board (ARM Cortex-A53, 1GB RAM)	Central server with ROS	5MP RGB (up to 16 cameras)
Du et al. [6]	Jetson AGX Xavier	Lab server with A6000 GPU	Not specified
Hwang and Kim [8]	Multiple edge AI devices (OpenPose v1.7.0)	Single central server	Not specified
Hwang, Kim, and Kim [7]	4× Nvidia Jetson TX2	ASUS laptop (Intel i9, RTX3080)	4× Stereolabs Zed 2i (720p/60FPS)
Hwang et al. [9]	6× NVIDIA Jetson Xavier	Intel i9 CPU, RTX3080 GPU	Stereolabs ZED 2i RGB-D
Kim and Hwang [10]	3× Jetson TX2, 3× Jetson AGX	Intel i9 CPU, RTX3080 GPU	6× RGB cameras
Liu et al. [12]	Not applicable	Desktop PC (i7-9700K, 16GB, GTX 1660Ti)	2-4× RGB cameras (1280×720)
Yang and Wang [14]	Not specified	3 acquisition PCs + 1 control PC	12× Industrial cameras (2048×1536, 125 FPS))
Zhang and Xu [16]	2× Jetson Xavier NX, 3× Jetson TX2	Dell desktop (i7-8700K, 2× GTX 1080 Ti, 11GB)	5× Logitech C270 HD Webcam
Ours	Virtual Edge Device (similar with Jetson series)	Desktop PC (RTX 2080ti 11GB, 32GB RAM)	5x Nerian Ruby 3D Depth Camera

Table 5.6: Distributed Systems Metrics

Paper	**Accuracy Metrics**	**Bandwidth Efficiency**	**Scalability**	**Real-time**
Boldo et al. [2]	MAE: 4.5-8.9 cm, PCC: 97.5-98.8%	108-1009 kbps (very low)	Up to 10 cameras tested	22-24 FPS
Bultmann and Behnke [3]	MPJPE: 29.8 mm, JDR: 97.5%	99% reduction (15 kB/s vs 27 MB/s)	Up to 16 cameras	30 FPS
Du et al. [6]	Comparable to benchmarks	Optimized through camera selection	Multiple cameras	Real-time
Hwang and Kim [8]	Campus: 0.88 PCP, Shelf: 0.88 PCP	Minimal (2D poses + 48 bits)	Multiple cameras	-
Hwang, Kim, and Kim [7]	PDJ improvements with T-Sync	64 Kbps vs 3.5 Mbps (98% reduction)	4 cameras tested	20 FPS
Hwang et al. [9]	-	Edge processing reduces load	6 cameras	15 FPS (3 people)
Kim and Hwang [10]	-	Hundreds of kbps	6 cameras	30 FPS constant
Liu et al. [12]	ARE: 4.78 pixels	-	Up to 4 cameras	-
Yang and Wang [14]	99.8% AP (single), 72.6% AP (triple)	Distributed computing	12 cameras	40 FPS
Zhang and Xu [16]	Comparable to state-of-the-art	WiFi transmission optimized	5 cameras optimal	Real-time
Ours	Same as FasterVoxelPose	Optimized through heatmaps	5 cameras tested	30 FPS constant

Chapter 6

Conclusion & Future Work

This dissertation presented the development of a novel distributed 3D human pose estimation system, which effectively leverages edge computing for initial 2D pose heatmap extraction and a centralized server for subsequent 3D pose reconstruction utilizing the FasterVoxelPose model. The core objective of this research was to address two fundamental questions: firstly, how 3D pose estimation in a multi-camera environment can balance accuracy and efficiency in real-world scenarios (RQ1); and secondly, to explore the trade-offs between accuracy, processing time, and scalability in 3D pose estimation (RQ2). The architectural design was specifically conceived to address these objectives by strategically distributing computational load and by integrating a state-of-the-art 3D pose estimation model, thereby aiming to balance accuracy and efficiency while exploring the inherent trade-offs between accuracy, processing time, and scalability. The comprehensive performance analysis demonstrated the system's real-time capabilities, exhibiting competitive frame rates at both the edge devices and the central server, coupled with state-of-the-art accuracy.

Despite the demonstrated efficacy and promising performance characteristics, the current iteration of the system inherently possesses certain limitations that warrant further consideration. Firstly, the operational deployment necessitates an initial manual camera calibration. This requirement introduces a manual setup overhead, potentially limiting flexibility and ease of deployment in dynamic or new environments. Secondly, the evaluation of edge device performance was conducted utilizing emulated edge devices. While providing valuable insights into computational efficiency and resource consumption under controlled conditions, this approach may not fully capture the real-world performance of the physical hardware. Thirdly, the camera synchronization relies on an NTP server. While adequate for current demands, 30 FPS, and up to 60 FPS, this method may not provide the hyper-accurate synchronization required if future applications demand significantly higher frame rates, potentially introducing temporal misalignment issues.

Addressing these limitations forms the basis for compelling future research directions:

- **Automated Camera Calibration:** Integrate an automated camera calibration procedure at system startup. Such a feature would drastically reduce deployment complexity and enhance the system's adaptability to new environments.

- **Deployment on Physical Edge Devices:** The use of real edge devices will enable the full performance of the system, making it more suitable for real-world deployment.
- **Enhanced Camera Synchronization:** For applications demanding high frame rates, implement a more precise camera synchronization mechanism, potentially a hardware-based system, essential to minimize temporal discrepancies across views.

In conclusion, this dissertation presents a robust and efficient distributed framework for 3D human pose estimation, offering a viable solution for real-time applications such as surveillance systems, physical rehabilitation and sports assessments. While the system exhibits promising results, there is still room for improvement, particularly in the automation of the camera calibration process, the transition from emulated to physical edge devices, and the refinement of the camera synchronization method.

References

[1] Miniar Ben Gamra and Moulay A. Akhloufi. "A review of deep learning techniques for 2D and 3D human pose estimation". In: *Image and Vision Computing* 114 (Oct. 2021), p. 104282. ISSN: 0262-8856. DOI: 10.1016/j.imavis.2021.104282. URL: https://www.sciencedirect.com/science/article/pii/S0262885621001876 (visited on 10/10/2024).

[2] Michele Boldo et al. "Real-time multi-camera 3D human pose estimation at the edge for industrial applications". English. In: *Expert Systems with Applications* 252.PA (2024). ISSN: 09574174. DOI: 10.1016/j.eswa.2024.124089. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85192087637&doi=10.1016%2fj.eswa.2024.124089&partnerID=40&md5=dc9114c51503e3ce02a0c0aa0f1599f7.

[3] Simon Bultmann and Sven Behnke. "Real-Time Multi-View 3D Human Pose Estimation using Semantic Feedback to Smart Edge Sensors". English. In: *Robotics: Science and Systems*. MIT Press Journals, 2021. ISBN: 978-0-9923747-7-8. DOI: 10.15607/RSS.2021.XVII.040. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85117812900&doi=10.15607%2fRSS.2021.XVII.040&partnerID=40&md5=ed98153225875daf64799749b7ffdc17.

[4] Rohan Choudhury, Kris M. Kitani, and László A. Jeni. "TEMPO: Efficient Multi-View Pose Estimation, Tracking, and Forecasting". English. In: *Proceedings of the IEEE International Conference on Computer Vision*. Institute of Electrical and Electronics Engineers Inc., 2023, pp. 14704–14714. ISBN: 979-835030718-4. DOI: 10.1109/ICCV51070.2023.01355. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85180647443&doi=10.1109%2fICCV51070.2023.01355&partnerID=40&md5=8c85569fd4be0a5db53a14318f370910.

[5] Junting Dong et al. "Fast and Robust Multi-Person 3D Pose Estimation and Tracking From Multiple Views". English. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44.10 (2022), pp. 6981–6992. ISSN: 01628828. DOI: 10.1109/TPAMI.2021.3098052. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85111013140&doi=10.1109%2fTPAMI.2021.3098052&partnerID=40&md5=3368415b079a60caef49ccba29bf35aa.

[6] Zhuohang Du et al. "Camera-Selecting Device-Edge Co-Inference for Real-Time Multi-Camera 3D Pose Estimation". English. In: *IEEE Vehicular Technology Conference*. Institute of Electrical and Electronics Engineers Inc., 2023, pp. 1–6. ISBN: 979-835032928-5. DOI: 10.1109/VTC2023-Fall60731.2023.10333552. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85181176000&doi=10.1109%2fVTC2023-Fall60731.2023.10333552&partnerID=40&md5=3fb15cfbfd2c08e2ab92a7611bb102df.

[7] Taemin Hwang, Jieun Kim, and Minjoon Kim. "A Distributed Real-time 3D Pose Estimation Framework based on Asynchronous Multiviews". English. In: *KSII Transactions on Internet and Information Systems* 17.2 (2023), pp. 559–575. ISSN: 19767277. DOI: 10.3837/tiis.2023.02.015. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85151574717&doi=10.3837%2ftiis.2023.02.015&partnerID=40&md5=98d442b66fd9565178adeb7a8419e915.

[8] Taemin Hwang and Minjoon Kim. "Noise-Robust 3D Pose Estimation Using Appearance Similarity Based on the Distributed Multiple Views". English. In: *Sensors* 24.17 (2024). ISSN: 14248220. DOI: 10.3390/s24175645. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85203860631&doi=10.3390%2fs24175645&partnerID=40&md5=1f5122f3fbc9a8cc4a2eca07731ff18c.

[9] Taemin Hwang et al. "A Real-time Multi-Person 3D Pose Estimation System from Multiple RGB-D Views for Live Streaming of 3D Animation". English. In: *Companion Proceedings of the 28th International Conference on Intelligent User Interfaces*. IUI '23 Companion. New York, NY, USA: Association for Computing Machinery, 2023, pp. 105–107. ISBN: 9798400701078. DOI: 10.1145/3581754.3584144. URL: https://doi.org/10.1145/3581754.3584144.

[10] Minjoon Kim and Taemin Hwang. "Real-Time Multi-view 3D Pose Estimation System with Constant Frame Speed". English. In: *Communications in Computer and Information Science* 1832 CCIS (2023), pp. 250–255. ISSN: 18650929. DOI: 10.1007/978-3-031-35989-7_32. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85171326942&doi=10.1007%2f978-3-031-35989-7_32&partnerID=40&md5=42fb030898d67af7d1931dc7a580f501.

[11] Jiahao Lin and Gim Hee Lee. "Multi-view multi-person 3D pose estimation with plane sweep stereo". English. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE Computer Society, 2021, pp. 11881–11890. ISBN: 978-1-66544-509-2. DOI: 10.1109/CVPR46437.2021.01171. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85123194931&doi=10.1109%2fCVPR46437.2021.01171&partnerID=40&md5=b86f67cba4f0d54ba1486890de28e5

[12] Kang Liu et al. "Auto calibration of multi-camera system for human pose estimation". English. In: *IET Computer Vision* 16.7 (2022), pp. 607–618. ISSN: 17519632. DOI: 10.1049/cvi2.12130. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85135783311&doi=10.1049%2fcvi2.12130&partnerID=40&md5=17a3fedff2d03a309fb36b3e6a613d9b.

[13] Hanyue Tu, Chunyu Wang, and Wenjun Zeng. "VoxelPose: Towards Multi-camera 3D Human Pose Estimation in Wild Environment". English. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 12346 LNCS (2020), pp. 197–212. ISSN: 03029743. DOI: 10.1007/978-3-030-58452-8_12. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85097248916&doi=10.1007%2f978-3-030-58452-8_12&partnerID=40&md5=3c607c02cc7a9c9721e44ba31dc3a925.

[14] Yuan Yang and Yangang Wang. "Real-Time Multi-View Human Pose Estimation System". English. In: *Proceedings - 2024 39th Youth Academic Annual Conference of Chinese Association of Automation, YAC 2024*. Institute of Electrical and Electronics Engineers Inc., 2024, pp. 2181–2186. ISBN: 979-835037922-8. DOI: 10.1109/YAC63405.2024.10598814. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.

0-85200719686&doi=10.1109%2fYAC63405.2024.10598814&partnerID=40&md5=2262e97b81aa48223dcb61d9c2a1195b.

[15] Hang Ye et al. "Faster VoxelPose: Real-time 3D Human Pose Estimation by Orthographic Projection". English. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 13666 LNCS (2022), pp. 142–159. ISSN: 03029743. DOI: 10.1007/978-3-031-20068-7_9. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85144554637&doi=10.1007%2f978-3-031-20068-7_9&partnerID=40&md5=bd9d3e51d2915d696bc27aa2bc2e4f78.

[16] Letian Zhang and Jie Xu. "E3Pose: Energy-Efficient Edge-assisted Multi-camera System for Multi-human 3D Pose Estimation". English. In: *Proceedings of the 8th ACM/IEEE Conference on Internet of Things Design and Implementation*. IoTDI '23. New York, NY, USA: Association for Computing Machinery, 2023, pp. 52–65. ISBN: 9798400700378. DOI: 10.1145/3576842.3582370. URL: https://doi.org/10.1145/3576842.3582370.

[17] Yifu Zhang et al. "VoxelTrack: Multi-Person 3D Human Pose Estimation and Tracking in the Wild". English. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45.2 (2023), pp. 2613–2626. ISSN: 01628828. DOI: 10.1109/TPAMI.2022.3163709. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85128333359&doi=10.1109%2fTPAMI.2022.3163709&partnerID=40&md5=12d43bd473a5a7e1c1f34e5fc2a6f385.

[18] Mi Zhou et al. "ER-Net: Efficient Recalibration Network for Multi-view Multi-person 3D Pose Estimation". English. In: *International Conference on Virtual Rehabilitation, ICVR*. Vol. 2022-May. Institute of Electrical and Electronics Engineers Inc., 2022, pp. 298–305. ISBN: 978-1-66547-911-0. DOI: 10.1109/ICVR55215.2022.9847965. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85137156436&doi=10.1109%2fICVR55215.2022.9847965&partnerID=40&md5=32c909c2a88954dedc3e9a2de

[19] Zhiyi Zhu et al. "3D Associative Embedding: Multi-View 3D Human Pose Estimation in Crowded Scenes". English. In: *Proceedings of the 2023 4th International Conference on Computing, Networks and Internet of Things*. CNIOT '23. New York, NY, USA: Association for Computing Machinery, 2023, pp. 131–139. ISBN: 9798400700705. DOI: 10.1145/3603781.3603804. URL: https://doi.org/10.1145/3603781.3603804.

[20] Zonghuang Zhuang and Yue Zhou. "FasterVoxelPose+: Fast and Accurate Voxel-based 3D Human Pose Estimation by Depth-wise Projection Decay". English. In: *Proceedings of Machine Learning Research*. Vol. 222. ML Research Press, 2023, pp. 1763–1778. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85189627610&partnerID=40&md5=159b2ac1f0e5a5b11c23b439bb0410c2.