UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

DIÓGENES WALLIS DE FRANÇA SILVA

Unsupervised Multi-View Multi-Person 3D Pose Estimation

Recife

2023

DIÓGENES WALLIS DE FRANÇA SILVA

Unsupervised Multi-View Multi-Person 3D Pose Estimation

A M.Sc. Dissertation presented to the Center of Informatics of Federal University of Pernambuco in partial fulfillment of the requirements for the degree of Master of Science in Computer Science.

Concentration Area: Computational Intelligence

Advisor: Veronica Teichrieb

Co-advisor: João Paulo Silva do Monte Lima

Recife

2023

Catalogação na fonte
Bibliotecária Monick Raquel Silvestre da S. Portes, CRB4-1217

S586u Silva, Diógenes Wallis de França

Unsupervised multi-view multi-person 3D pose estimation / Diógenes Wallis de França Silva. – 2023.

74 f.: il., fig., tab.

Orientadora: Veronica Teichrieb.

Dissertação (Mestrado) – Universidade Federal de Pernambuco. CIn, Ciência da Computação, Recife, 2023.

Inclui referências.

1. Inteligência computacional. 2. Aprendizado profundo. I. Teichrieb, Veronica (orientador). II. Título.

006.31 CDD (23. ed.) UFPE - CCEN 2023-171

Diógenes Wallis de França Silva

“Unsupervised Multi-View Multi-Person 3D Pose Estimation”

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação. Área de Concentração: Inteligência Computacional.

Aprovado em: 28 de julho de 2023.

BANCA EXAMINADORA

____
Prof. Cleber Zanchettin
Centro de Informática / UFPE

____
Prof. Dr. Diego Gabriel Francis Thomas
Department of Advanced Information Technology / Kyushu University

____
Profa. Dra. Veronica Teichrieb
Centro de Informática / UFPE
(**Orientador**)

I dedicate this dissertation to my family and friends.

ACKNOWLEDGEMENTS

Thank God for all the energy and perseverance to keep my Artificial Intelligence journey. Since I was born, I have faced many challenges. Therefore, it is extraordinary for me to write this dissertation today. I remember doubting if I could study at a Federal University, and today I am an Engineer graduated from a Federal University. I am also grateful for finishing a master's degree in Computer Science.

I am also thankful for my amazing family. Even being a humble family, we always give relevance to studying and being the best we can be. Special thanks to my parents: Josélia and Dorgival. My mother Josélia (also called "olhinho") is always perfect in everything; I would be nothing without the support of my mother, even a 4000km distance; she is a fantastic mother. My dad Dorgival (also called "mané"), made it possible for me to focus on studying and always motivated me to achieve the most I could. Finally, I cannot forget my loved brothers, that are my best friends and also supported me, Diogo (also called "mutant") and Wanderson (also called "José").

I must mention that I had the best adviser in the world, Professor João Paulo (also called "Jonga"), that helped me to realize my dream of visiting Europe and participating in a congress in the UK. Jonga helped me a lot and made it possible for this dissertation to have two published papers. Always attentive and with great expertise, I could not imagine a better adviser than Jonga.

Thank you to Voxar Labs and VT for all their support and the fantastic people I could know there.

ABSTRACT

The problem of 3D pose estimation of multiple persons in a multi-view scenario has been an ongoing challenge in computer vision. Most current state-of-the-art methods for 3D pose estimation have relied on supervised techniques, which require a large amount of labelled data for training. However, generating accurate 3D annotations is costly, time-consuming, and prone to errors. Therefore, a novel approach that does not require labeled data for 3D pose estimation has been proposed. The proposed method, the Unsupervised Multi-View Multi-Person approach, uses a plane sweep method to generate 3D pose estimations. This approach defines one view as the target and the rest as reference views. First, the depth of each 2D skeleton in the target view is estimated to obtain the 3D poses. Then, instead of comparing the 3D poses with ground truth poses, the calculated 3D poses are projected onto the reference views. The 2D projections are then compared with the 2D poses obtained using an off-the-shelf method. Finally, the 2D poses of the same pedestrian obtained from the target and reference views are matched for comparison. The matching process is based on ground points to identify the corresponding 2D poses and compare them with the respective projections. To improve the accuracy of the proposed approach, a new reprojection loss based on the smooth L_1 norm has been introduced. This loss function considers the errors in the estimated 3D poses and the projections onto the reference views. It has been tested on the publicly available Campus dataset to evaluate the effectiveness of the proposed approach. The results show that the proposed approach achieves better accuracy than state-of-the-art unsupervised methods, with a 0.5% points improvement over the best geometric system. Furthermore, the proposed method outperforms some state-of-the-art supervised methods and achieves comparable results with the best-managed approach, with only a 0.2% points difference. In conclusion, the Unsupervised Multi-View Multi-Person approach is a promising method for 3D pose estimation in multi-view scenarios. Its ability to generate accurate 3D pose estimations without relying on labeled data makes it valuable to computer vision. The evaluation results demonstrate the proposed approach's effectiveness and potential for future research in this area.

Keywords: 3D human pose estimation; unsupervised learning; deep learning; reprojection error.

RESUMO

O problema da estimativa de pose 3D de múltiplas pessoas em cenários de múltiplas visualizações tem sido um desafio contínuo em visão computacional. A maioria dos métodos de estado da arte para estimativa de pose 3D atualmente depende de técnicas supervisionadas, que exigem uma grande quantidade de dados rotulados para o treinamento. No entanto, gerar anotações 3D precisas é caro, consome tempo e está sujeito a erros. Portanto, foi proposta uma abordagem nova que não requer dados rotulados para estimativa de pose 3D. A abordagem proposta não supervisionada que trata de múltiplas visualizações e múltiplas pessoas, utiliza um método de varredura de planos para gerar estimativas de pose 3D. Essa abordagem define uma visualização como alvo e as demais como visualizações de referência. Primeiramente, a profundidade de cada esqueleto 2D na visualização alvo é estimada para obter as poses 3D. Em seguida, em vez de comparar as poses 3D com as poses verdadeiras, as poses 3D calculadas são projetadas nas visualizações de referência. As projeções 2D são, então, comparadas com as poses 2D obtidas usando um método pronto para uso. Por fim, as poses 2D do mesmo pedestre obtidas a partir das visualizações alvo e de referência são comparadas para avaliação. O processo de comparação é baseado em pontos de referência para identificar as poses 2D correspondentes e compará-las com as respectivas projeções. Para melhorar a precisão da abordagem proposta, foi introduzida uma nova perda de reprojeção baseada na norma L_1 suave. Essa função de perda considera os erros nas poses 3D estimadas e nas projeções nas visualizações de referência. Ela foi testada no conjunto de dados público Campus para avaliar a eficácia da abordagem proposta. Os resultados mostram que a abordagem proposta alcança maior precisão do que os métodos não supervisionados de estado da arte, com uma melhoria de 0,5 ponto percentual em relação ao melhor sistema geométrico. Além disso, o método proposto supera alguns métodos supervisionados de estado da arte e alcança resultados comparáveis com a melhor abordagem supervisionada, com apenas uma diferença de 0,2 ponto percentual. Em conclusão, a proposta abordagem não supervisionada em um cenário com múltiplas vistas e múltiplas pessoas é um método promissor para a estimativa de pose 3D. Sua capacidade de gerar estimativas de pose 3D precisas sem depender de dados rotulados a torna valiosa para a visão computacional.

Palavras-chaves: estimação de poses humanas em 3D; aprendizado não supervisionado; aprendizado profundo; erro de reprojeção.

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

3DPS	3D Pictorial Structure
BERT	Bidirectional Encoder Representations from Transformers
CNN	Convolutional Neural Network
CPN	Cuboid Proposal Network
CRF	Conditional Random Field
GAN	Generative Adversarial Network
GPT	Generative Pre-Trained Transformer
HR-Net	High-Resolution Net
ICANN	International Conference on Artificial Neural Networks
KL	Kullback-Leibler
KTH	Kungliga Tekniska högskolan - Royal Institute of Technology
LSP	Leeds Sports Pose
LSTM	Long Short Term Memory
MPJPE	(Mean Per Joint Position Error
MRPE	Mean of the Root Position Error
MS-COCO	Microsoft Common Objects in Context
MSE	Mean squared error
PCK	Percentage of Correct Keypoints
PCP	Percentage of Correctly estimated Parts
PRN	Pose Regression Network
ReLU	Rectified linear unit
RGB	Red Green Blue
RNN	Recurrent Neural Network
ssvm	Structured Support Vector Machine
svm	Support Vector Machine

VISAPP	International Conference on Computer Vision Theory and Applications

LIST OF SYMBOLS

Σ	Summation
$\in$	Membership
arg min	Set of values where a function attains minimum
σ	Standard deviation
λ	Integer multiplier
$\mathcal{N}$	Gaussian distribution
δ	Offset
$\lvert x\rvert$	Absolute value

CONTENTS

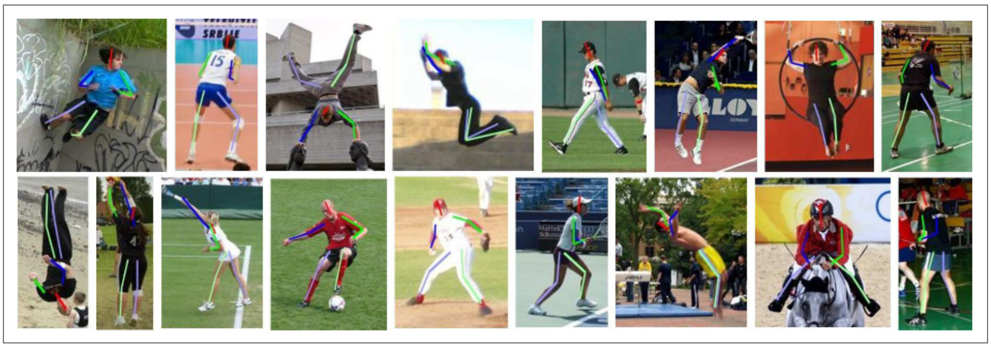
1 INTRODUCTION

3D human pose estimation is an active research area on Computer Vision. It has the goal of estimating the 3D position of articulated human joints. Depending on the scenario, these joints are obtained from RGB images acquired by single or multiple cameras. Furthermore, 3D human pose estimation has several applications: video surveillance, autonomous driving, biomechanics and medicine, sports performance analysis and education, autonomous driving, human-computer interaction, psychology, try-on, and fashion (WANG et al., 2021). Therefore, it is a relevant and large field to be explored and an excellent opportunity to generate impacting contributions to society.

Some application areas are selected, and it is described the relevance of the 3D pose estimation on these real-world applications (WANG et al., 2021):

- *Sports performance and instruction*. People playing sports need to execute physical movements following specific rules. The manner the person moves or executes the activities can be improved using 3D pose estimation (HWANG; PARK; KWAK, 2017). For example, if a football player kicks a ball in a penalty kick, the leg's angle and the foot's position can wrongly lead the ball far away from the goal. Estimating its 3D position makes it possible to measure and suggest adaptations and corrections to the football player. In this manner, this player can improve. This context can be extended to other sports like swimming, skateboarding, marathons, and gym training. In this case, it can suggest the best position to exercise a specific muscle. Figure 1 shows some pose estimations in a sports dataset;

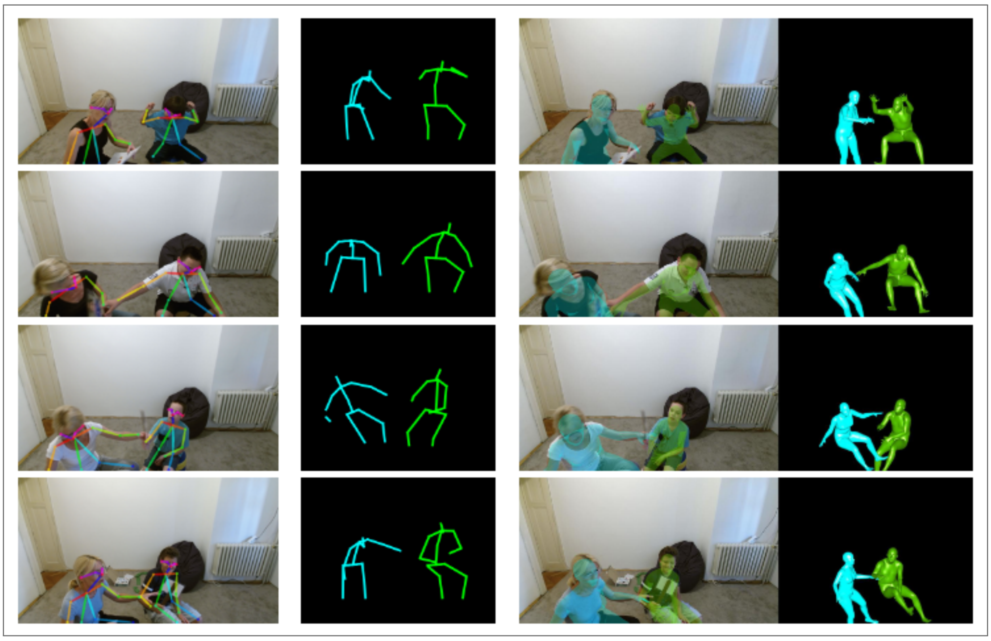
Figure 1 – Qualitative results from (HWANG; PARK; KWAK, 2017) on LSP dataset (JOHNSON; EVERINGHAM, 2010).



Source: HWANG; PARK; KWAK (2017)

- *Psychology*. Mental states or emotions also can be recognized from 3D poses. As in (MARINOIU et al., 2018), it is possible to use the 3D pose estimations in therapy for helping people with mental diseases, in this case, children with autism. The Figure 2 illustrates human interactions and the respective 2D and 3D pose estimations;

Figure 2 – 2D and 3D pose reconstructions shown on annotated dataset from (MARINOIU et al., 2018).



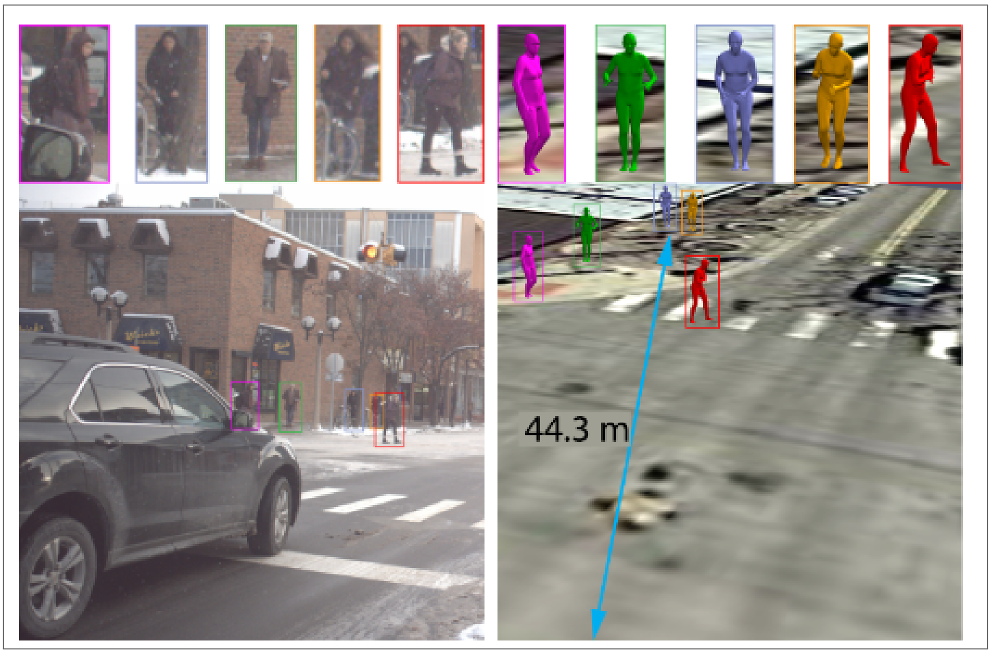
Source: MARINOIU et al. (2018)

- *Autonomous Driving*. An autonomous car can deal with undesired situations involving pedestrians, such as collisions. 3D pose estimation can be used to avoid these collisions utilizing the information about the positioning and movement intention of the pedestrian (KIM et al., 2019). Figure 3 shows pedestrian estimated poses.

This work deals with multi-view multi-person 3D pose estimation. Estimating poses from multiple people is important in several applications, such as surveillance, human-computer interaction, and augmented reality. (CORMIER et al., 2022) shows pose estimation for surveillance context as illustrated on Figure 4. (CIMEN et al., 2018) is an example of human pose estimation for augmented reality, as shown in Figure 5.

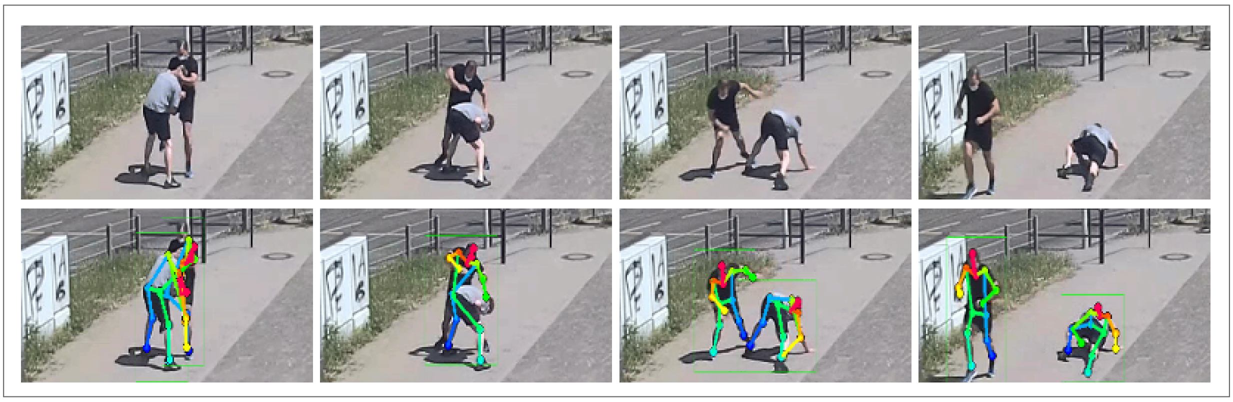
It uses multiple cameras to estimate people's positions. The cameras need to be calibrated (once we need to know the camera's extrinsic and intrinsic parameters) and synchronized. Furthermore, unlike monocular solutions, multiple views have the advantage of obtaining depth information. In this manner, it is possible to solve occlusion problems. Also, multi-view allows

Figure 3 – Left: Image with bounding boxes around the pedestrians. Right: A rendered image with the 3D human mesh models.



Source: KIM et al. (2019)

Figure 4 – Street fight containing multiple human interactions for pose estimation.

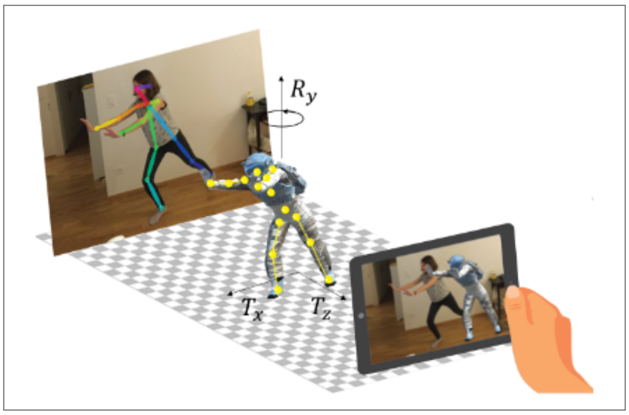


Source: CORMIER et al. (2022)

various cameras to be available in several environments, increasing the coverage. The visibility is also improved using more than one singular camera, as shown in Figure 6.

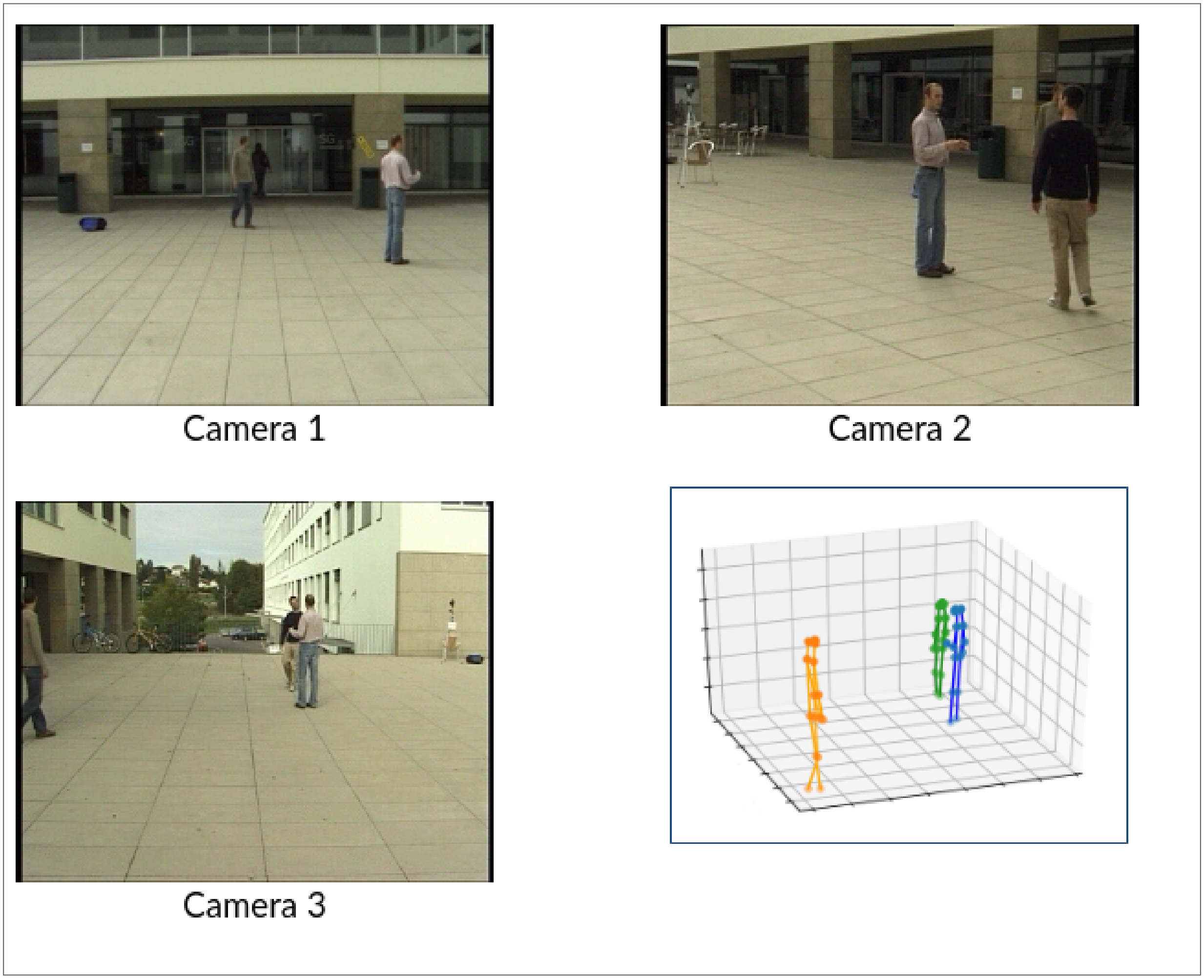
With the RGB images obtained by each camera, 2D pose estimations are generated. These estimations are the input to the method presented in this dissertation to estimate the 3D skeleton of each person. Beyond the proposed work (that will be detailed) (BELAGIANNIS et al., 2014; BELAGIANNIS et al., 2014; BELAGIANNIS et al., 2015; DONG et al., 2019; ERSHADI-NASAB et al., 2018; HUANG et al., 2020; TU; WANG; ZENG, 2020; LIN; LEE, 2021) also deal with the 3D pose estimation in a multi-view multi-person scenario. The first works were based on geometric

Figure 5 – Automatically augmenting mobile pictures with digital avatars imitating poses.



Source: CIMEN et al. (2018)

Figure 6 – Multi-view example in an outdoor environment from (BERCLAZ et al., 2011). The same scene is viewed by different cameras, bringing more coverage, visibility, redundancy and depth information.



Source: The author (2023)

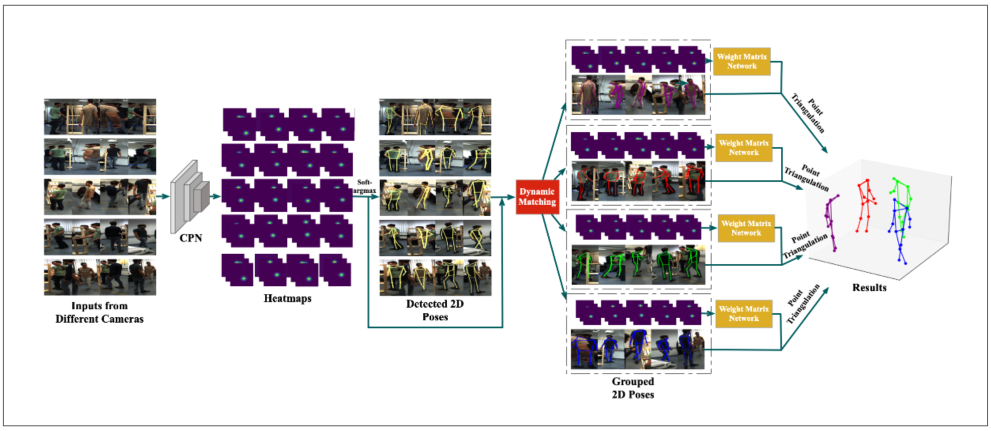
approaches, and the most recent are developed using neural networks. The proposed approach provides a solution using neural networks and geometric concepts, enabling 3D pose estimation in an unsupervised manner.

An unsupervised 3D pose estimation method has the advantage of not using annotated 3D ground truth. Generating 3D pose labels is a high-cost process and can have wrong annotations and other joint problems related to the label-generating process. The Campus dataset (BERCLAZ et al., 2011), for example, was generated by manual joint annotation (BELAGIANNIS et al., 2014). In this method, the reference values compared with the estimations (output of the neural network) are obtained along with the 3D pose estimation. Therefore, only the camera parameters and the synchronization among the cameras are necessary. In that case, it performs neural network training more efficiently than traditional neural network methods. Consequently, it can create a 3D pose estimation model without needing people to generate 3D annotated ground truth.

1.1 PROBLEM

3D human pose estimation is a challenging task. The challenge is even more significant when dealing with multiple people. However, some scenarios facilitate the creation of robust solutions, such as using various cameras. Furthermore, multiple cameras can provide depth information different from a single camera. This dissertation proposes to estimate 2D poses from different views and combine them using a neural network approach to generate 3D poses. Figure 7 briefly shows a structure for 3D pose estimation using Neural Networks.

Figure 7 – Schema of Neural Network approach for multi-view multi-person 3D pose estimation.



Source: HUANG et al. (2020)

The most relevant works on 3D pose estimation are based on geometric methods or neural networks. The neural network approach achieves the best results. However, it has the limitation and costs of obtaining labeled datasets. These datasets are obtained with human annotations. Therefore, they can have errors and also a high price. Furthermore, the need for 3D labeled data makes the methods less general since they can work only when 3D ground truth is available.

This work aims to perform 3D pose estimation using state-of-the-art approaches like neural networks. However, it also has the goal of not using 3D annotated data. In this manner, it can perform training without using these labeled data.

1.1.1 Hypothesis

In the rest of this dissertation, we will analyze the hypothesis statements presented below:

- H1: The reprojection error of 2D poses can be utilized as a loss function for training models in 3D pose estimation. By doing so, there is no need to employ losses that directly compare the estimations with 3D labels. This hypothesis is tested by incorporating the reprojection error as a loss in a neural network designed to estimate 3D poses.
- H2: The reprojection error mentioned earlier can be calculated using the smooth L1 loss. This hypothesis is tested by employing the mentioned loss for comparing the 2D poses.
- H3: In a multi-person scenario, calculating reprojection error requires performing person matching between views. Ground point matching and back-projection approaches can be used for this purpose. The matching directly impacts the comparison of 2D poses; therefore, a robust method is necessary to achieve accurate results.

1.2 GOALS

The main goal of this work is to perform 3D pose estimation in an unsupervised manner combining neural networks and geometric concepts. This dissertation has the following specific purposes:

- Study multi-view multi-person 3D pose estimation methods, focusing on the state-of-the-art. The goal is to identify the key contributions of each paper, aiming to develop the presented method;
- Group techniques based on their approach: neural networks and geometric approaches;

- Study geometric and neural network-based techniques that can be able to generate 3D poses in an unsupervised way;

- Estimate 3D poses of multiple people in a multiple-view scenario using geometric and neural network-based techniques in an unsupervised manner.

- Evaluate the proposed approach using PCP (percentage of correctly estimated parts) metric (WANG et al., 2021). Considering the complexity of the dataset and the precision of 2D part detectors, the PCP score provides more informative results compared to those based on the Euclidean distance.

1.3 CONTRIBUTIONS

The following contributions can be pointed out:

- A review of methods for 3D pose estimation of multiple people in a multi-view scenario, classifying the techniques into two groups: neural networks and geometric;

- An innovative approach to performing person matching is presented. The method utilizes ground points to represent each person, instead of comparing 2D poses. The matching process is based on measuring the distance between single points attached to each person. This approach avoids the use of more complex techniques like person re-identification or epipolar distance, which would also require higher computational costs;

- An unsupervised manner of training a model able to perform 3D pose estimation in a multi-view multi-person scenario, that is, training a model without using 3D labeled data;

- Examination and comparison of the acquired findings with pertinent studies, encompassing both geometric and neural networks approaches.

- Publication at International Conference on Artificial Neural Networks (ICANN) 2022: Unsupervised Multi-view Multi-person 3D Pose Estimation Using Reprojection Error.

- Publication at International Conference on Computer Vision Theory and Applications (VISAPP) 2023: UMVpose++: Unsupervised Multi-View Multi-Person 3D Pose Estimation Using Ground Point Matching.

1.4 WORK STRUCTURE

The chapters are organized as follows. Chapter 2 briefly shows and explains essential basic concepts to understand the core topics of 3D pose estimation. Chapter 3 discusses the related works, classified as neural networks or geometric methods. Chapter 4 describes the proposed process, showing the core concepts for developing this unsupervised 3D pose estimation approach. Chapter 5 analyzes and discusses the results and experiments, comparing the presented method results with the state of the art. Finally, Chapter 6 presents the conclusion and suggestions for future works.

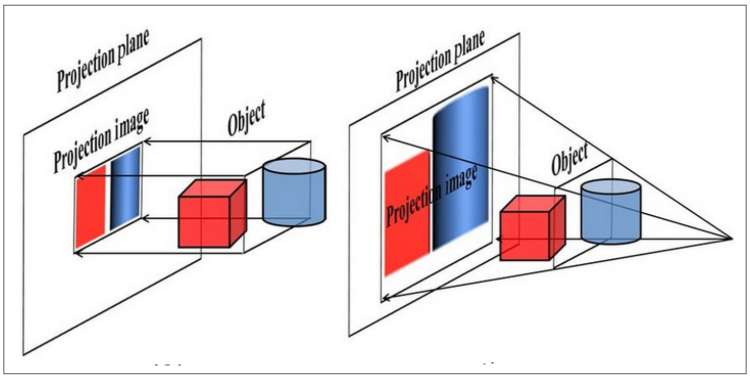
2 THEORETICAL BACKGROUND

This chapter discusses the key concepts to define a theoretical base related to this work. These concepts cover the necessary knowledge to develop the solution and research proposed. The foundations of this work include the Computer Vision and Artificial Intelligence areas. Section 2.1 describes how to project points from 3D to 2D using camera parameters. Section 2.2 shows how the reprojection error works. Section 2.3 describes key machine learning topics, such as learning methods (supervised, unsupervised, and semi-supervised). Section 2.4 explains relevant deep learning components such as Convolutional Neural Network (CNN) and the general view of loss functions—finally, section 2.5 presents evaluation metrics for comparing 3D skeletons.

2.1 CAMERA MULTIVIEW GEOMETRY

A camera is an instrument made by sensors responsible for taking objects (or points) in the world (3D points) and projecting them onto 2D images. This process involves a well-defined pipeline, converting world points to camera points and image points. This transformation of 3D (real world) to 2D (image) is also called projection. The projection maps 3D objects to 2D images in a defined projection plane (HARTLEY; ZISSERMAN, 2003). Furthermore, there are different methods of projection, as shown in Figure 8. However, this work is interested in perspective projection. Therefore, the following paragraphs detail the process and explain the camera parameters.

Figure 8 – As shown above, two 3D objects are mapped to 2D using different techniques. The first is an orthographic projection, and the second is a perspective projection.



Source: JIA et al. (2014)

The transformation from 3D to 2D is made using projective geometry. This 3D to 2D mapping is created by matrix multiplications, considering points as vectors. These matrices must contain information related to the intrinsic and extrinsic camera parameters. Modeling the camera as the projection matrix **P**, we can convert world points to image points (HARTLEY; ZISSERMAN, 2003). The projection of 3D onto 2D is described with the following equation:

$$\mathbf{x} = \mathbf{PX}, \tag{2.1}$$

where **x** are the 2D coordinates (image point), **P** is the projection matrix, and **X** are the world 3D points. This equation maps 3D to 2D, multiplying the camera matrix by the world coordinates. The camera matrix contains parameters related to internal and external camera factors. In this manner, the parameters must correctly represent the camera, obligating the camera to be calibrated. Equation 2.1 can also be written as:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} p_1 & p_2 & p_3 & p_4 \\ p_5 & p_6 & p_7 & p_8 \\ p_9 & p_{10} & p_{11} & p_{12} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}. \tag{2.2}$$

The **x** equation is related to homogeneous coordinates. Initially, the image has 2D points, and this third coordinate is added to represent scale and translation, making more effortless transformations such as scaling, rotation, and translation. In addition, homogeneous coordinates facilitate lines and shape representations in the space and projection and back-projection operations (HARTLEY; ZISSERMAN, 2003). The same applies to **X**: in this case, there are four coordinates, considering the additional homogeneous coordinate for the world 3D point. Note that to convert **x** from homogeneous coordinates to the corresponding 2D vector, remove the third coordinate and divide the x and y terms by z.

The camera matrix contains information related to the camera position and internal elements from a camera that impact image projection. In this manner, the camera matrix is compound by the intrinsic and extrinsic (translation and rotation) matrices (KITANI, 2017) as shown in the equation below:

$$P = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} intrinsic \begin{bmatrix} r_1 & r_2 & r_3 & t_1 \\ r_4 & r_5 & r_6 & t_2 \\ r_7 & r_8 & r_9 & t_3 \end{bmatrix} extrinsic. \quad (2.3)$$

Intrinsic parameters are related to how the camera captures the images. The intrinsic matrix contains focal length (f_x and f_y) and translation operators (c_x and c_y). The goal of the intrinsic matrix is to convert from the camera coordinate system to the pixel coordinate system. The extrinsic matrix parameters are related to camera location and orientation (translation and rotation matrices). Therefore, the extrinsic matrix must convert from world coordinates to camera coordinates (FORSYTH; PONCE, 2002). It obtains the projection matrix **P** by multiplying these matrices.

2.2 REPROJECTION ERROR

As mentioned in Section 2.1, it is possible to project points from 3D to 2D or back-project from 2D to 3D, given the camera parameters. There are measures to check if the parameters are correct and if the camera is calibrated. One manner to verify this is the reprojection error. The reprojection error involves the distance measured between a projected point and the actual point position in that image (HARTLEY; ZISSERMAN, 2003).

Beyond that, it is also possible to check a 3D position of a point using reprojection error. Given a multi-view scenario, a fact viewed in all camera images can be estimated in world coordinates. In case this 3D estimation is correct, the projection onto the camera images must be at the same position as the original point location in the camera images.

Considering the actual point position as **x** and **x̂** as the projected point, the reprojection error is simple to calculate, as seen in the following equation:

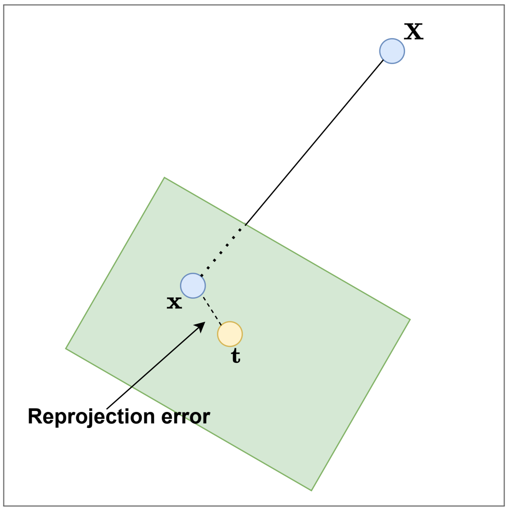
$$d(\mathbf{x}, \hat{\mathbf{x}}) = \sqrt{\sum_{i=1}^{n} |x_i - \hat{x}_i|^2}. \quad (2.4)$$

The $d(\mathbf{x}, \hat{\mathbf{x}})$ term is related to Euclidean distance between the true position **x** and the projected point **x̂**. This projected point is obtained by multiplying the projection matrix **P** by the corresponding 3D point $\hat{\mathbf{X}}$ that we wish to project:

$$\hat{\mathbf{x}} = \mathbf{P}\hat{\mathbf{X}}. \quad (2.5)$$

Figure 9 shows an example of reprojection error.

Figure 9 – The 3D point **X** is projected onto an image (green plane), and this projection **x** is used for comparison. The reprojection error is the Euclidian distance between the projected point **x** and the actual point position **t**.



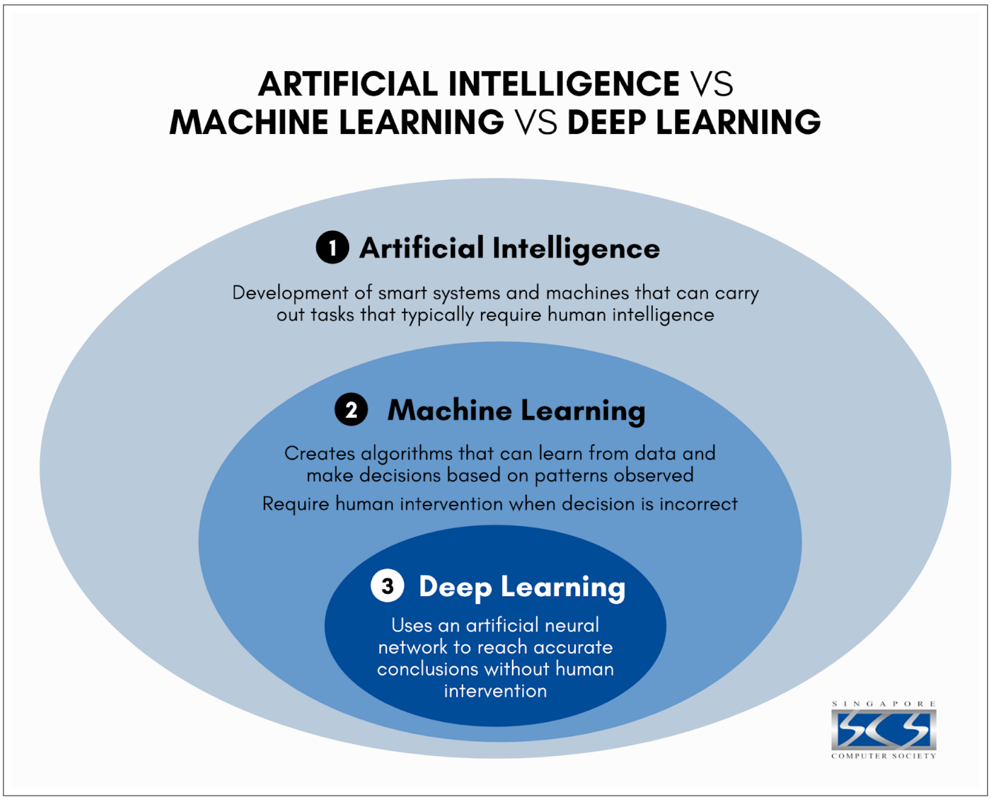
Source: The author (2023)

2.3 MACHINE LEARNING

According to Machine Learning pioneer Tom Mitchell, "Machine Learning is the study of computer algorithms that allow computer programs to improve through experience automatically" (MITCHELL et al., 2007). This definition shows the core idea behind Machine Learning: algorithms are the methods used for the computer learning process, and the experience is directly related to data. As a branch of Artificial Intelligence, Machine Learning has the goal of computer learning through training models and data. Beyond that, machine learning methods can be used in applications such as recommendation systems, object detections, clustering, 2D or 3D pose estimation, and image segmentation. Furthermore, Machine Learning is a topic encompassed by Artificial Intelligence, as shown in Figure 10.

Machine Learning involves many relevant topics related to its structure: learning algorithms according to the problem context, evaluation metrics, optimization algorithms, data analysis, and feature engineering in some cases. Beyond that, Machine Learning has several methods according to how the data is used. Some of these methods are supervised learning, unsupervised learning, and semi-supervised learning.

Figure 10 – Artificial Intelligence encompasses Machine Learning and Deep learning, and Machine Learning encompasses Deep Learning. Artificial Intelligence is more general, considering any intelligent system able to execute tasks that need human Intelligence. However, Machine Learning is made by a learning algorithm that depends on data. The same applies to Deep Learning, but the learning algorithms are more complex using neural networks.


ARTIFICIAL INTELLIGENCE VS
MACHINE LEARNING VS DEEP LEARNING
1 Artificial Intelligence
Development of smart systems and machines that can carry out tasks that typically require human intelligence
2 Machine Learning
Creates algorithms that can learn from data and make decisions based on patterns observed
Require human intervention when decision is incorrect
3 Deep Learning
Uses an artificial neural network to reach accurate conclusions without human intervention
SINGAPORE SCS COMPUTER SOCIETY

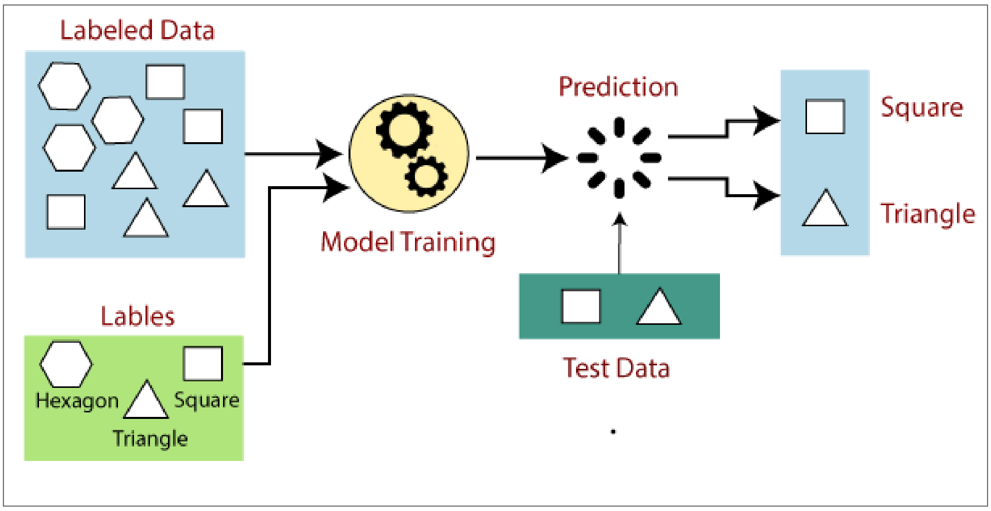
Source: SOCIETY (2021)

2.3.1 Supervised learning

Considering a machine learning model, how the model learns from data determines supervised learning. A managed model learns from labels attached to each sample. Considering data for training a model, this data can be divided into **X** and **y**, where **X** is the data features, and **y** are the labels. In the case of a credit analysis problem, **X** is the client's characteristics, such as age, credit history, and salary. Then **y** is the client's situation as a good or lousy payer. Based on client features, this model aims to learn to determine if that person will repay the loan. It is essential to mention that, along the training process, the model output is compared with the labels (ground truth). In this manner, the model parameters are adjusted to minimize the error (LIU; LIU, 2011).

Most Machine Learning models are supervised. Furthermore, there is the need to create labels. If the goal is to train a model to detect a face, a dataset must contain bounding boxes manually attached to each front. The model uses this information to tune the parameters and

Figure 11 – They supervised learning schema. Data is used as input along with the respective labels. The model uses both information (data features and labels) to learn. The next step after training the model is to make predictions using test data; in this manner, it is possible to check the model behaviour on external data and verify if the model solves the classification problem.



Source: RAGUPATHI (2022)

minimize the errors in the training dataset (LIU; LIU, 2011).

Labelling samples takes work. In some contexts, such as estimating 3D poses, obtaining ground truth data is challenging. Some online services provide the possibility of hiring people to perform labelling. Nevertheless, manual labels can be wrong, affecting the model training (TU; MENZIES, 2022).

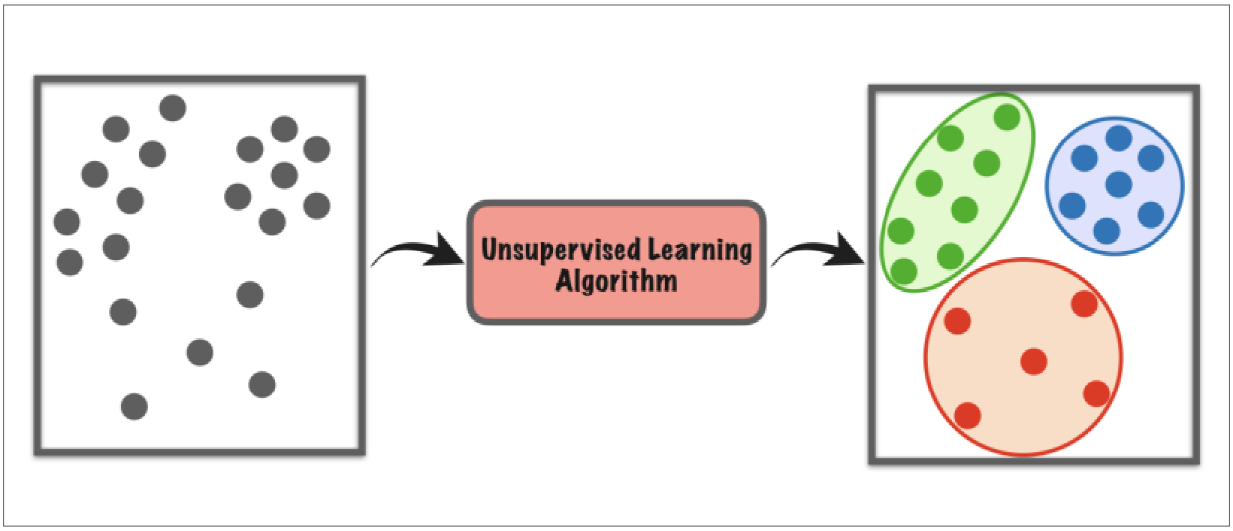
Beyond that, the model can suffer from overfitting, in the case the performance on training data is high, but the model needs to better generalize to external data. Therefore, before training the model, it is essential to make an exploratory data analysis to verify if the data is generalizing the context.

Therefore, the supervised methods need to be labelled data to compare with the model's output, and so learn how to apply this data knowledge. The labels are the core difference compared with the other machine learning approaches. Figure 11 illustrates the supervised learning process.

2.3.2 Unsupervised learning

Different from supervised methods, unsupervised learning does not need labelled data as shown in Figure 12. Furthermore, the context is different, as will be described. First, it briefly details the meaning of supervised learning and unsupervised, so it is easier to see the difference between them. Usually, supervised learning is present in classification and regression problems. A model is trained to aim to learn to predict a class correctly or estimate a value according to the respective received input. However, unsupervised learning refers to discovering underlying patterns in data without explicit training on labelled data. An unsupervised approach can detect clusters, for example, the K-means method. K-means is an unsupervised model that identifies groups based on data features. By taking centroids at a start point and updating each algorithm iteration, these centroids will be the centre of the created clusters. The goal is to create homogeneous groups considering the elements in each cluster and heterogeneous compared with the other clusters. The learning process occurs according to characteristics related to input data (DAYAN; SAHANI; DEBACK, 1999).

Figure 12 – Unsupervised learning schema. Just for illustration, this example considers data represented on a cartesian plane; this method extends to N-dimensional data. In this example, the data is described in a Cartesian plane, and the goal is to identify homogeneous groups. The cluster creation is made using an unsupervised learning algorithm. In this figure, there are 3 data clusters. The algorithm goal is that each group is homogeneous among its elements and heterogeneous related to other groups.



Source: JEFFARES (2018)

These created groups can serve as a manner to identify similar elements. Beyond that, measuring performance is directly related to the context and solution purpose, different from supervised methods that compare the prediction to ground truth and can obtain the model's performance.

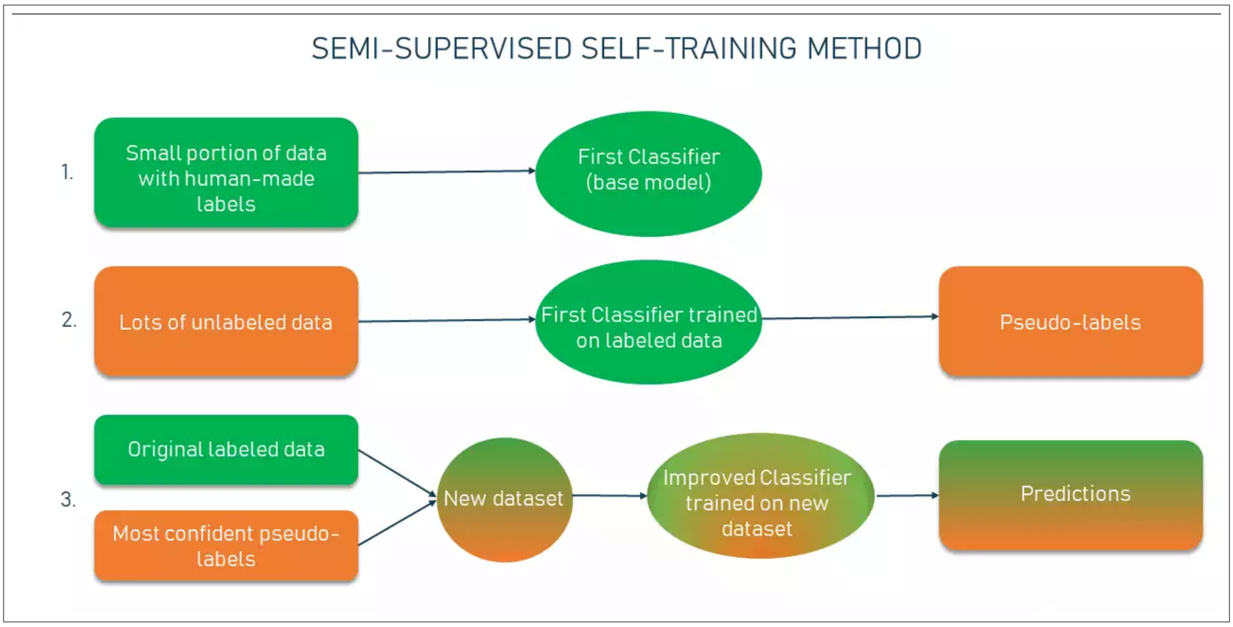
Beyond clustering, unsupervised learning is also used for association and dimensionality reduction. The association approach consists in obtaining relationships between variables, commonly used to develop recommendations in the retail market as Amazon's "Customers Who Bought This Item Also Bought". Finally, dimensionality reduction can be used to decrease the number of features in a dataset, which makes the training process faster or helps visualize the data, considering that fewer features are easier to understand (GHAHRAMANI, 2004).

2.3.3 Semi-supervised learning

The semi-supervised approach deals with unlabeled and labelled data. This manner is considered a hybrid method. Usually, a semi-supervised method is used to train a supervised model in a massive dataset with unlabeled data. The semi-supervised method can be used to generate these labels. Manually labelling data is costly and takes a considerable amount of time. Generating labels automatically using semi-supervised learning can help improve model performance once more data are able to perform supervised learning (ZHU, 2005).

Self-training is an example of semi-supervised learning to generate labels for this huge unlabeled dataset. The self-training procedure can be described in a well-defined pipeline. The first step is taking a small portion of manually labelled data, which is used to train a supervised model. This first classifier is used in the unlabeled data so that self-training can generate pseudo-labels. Each generated label has a level of confidence obtained as a prediction score. The generated labels with the highest confidence values complement the manually labelled data. This new dataset made by combining auto and manually-generated labels is used to train the supervised model, improving the model. This process can be iterative, and after retraining using the hybrid dataset, the process of generating labels can run again until the model performance is satisfactory (HADY; SCHWENKER, 2013). Figure 13 illustrates this pipeline.

Figure 13 – The self-training method is a semi-supervised method used in large unlabeled datasets. This figure illustrates the self-training pipeline. After step 3, it is possible to return to step 2 and iteratively apply the model.


SEMI-SUPERVISED SELF-TRAINING METHOD
1.
Small portion of data with human-made labels
First Classifier (base model)
2.
Lots of unlabeled data
First Classifier trained on labeled data
Pseudo-labels
3.
Original labeled data
Most confident pseudo-labels
New dataset
Improved Classifier trained on new dataset
Predictions

Source: ALTEXSOFT (2022)

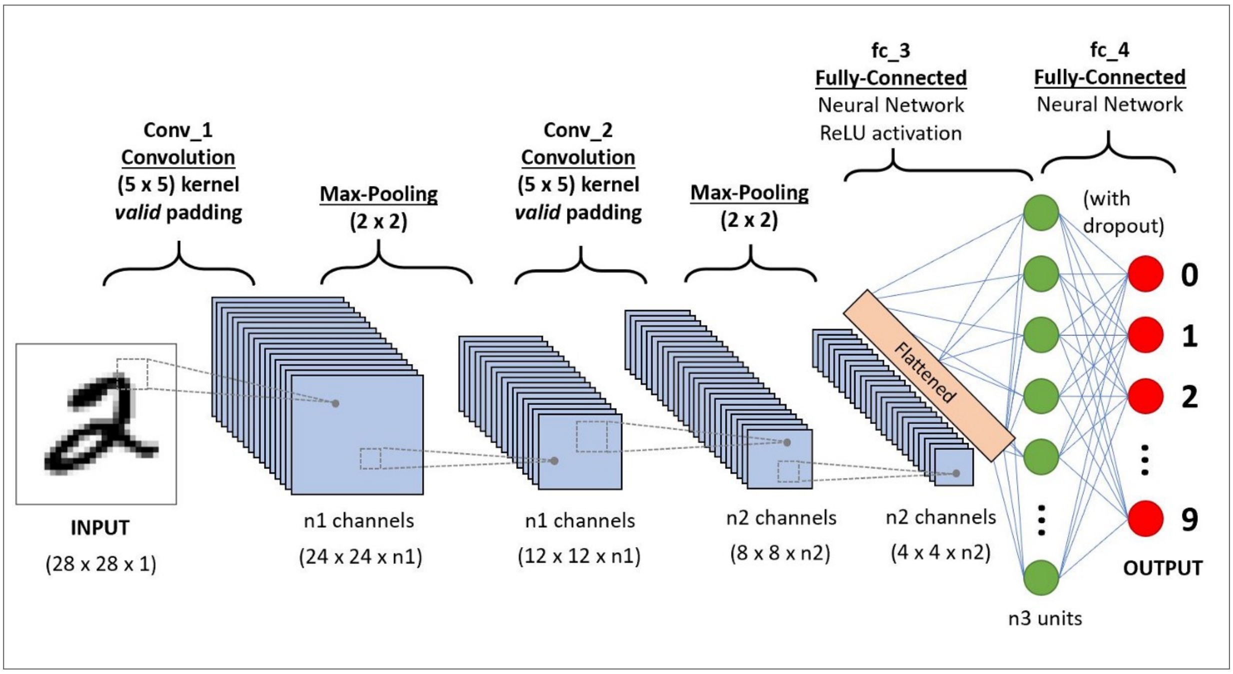
2.4 DEEP LEARNING

As shown in Figure 10, the Deep Learning approach is a subset of Machine Learning. Deep Learning consists of training learning models using artificial neural networks. Neural networks can have several layers and learn the relevant features from input data. There are a lot of deep learning structures such as CNN (convolutional neural networks) (O'SHEA; NASH, 2015), Recurrent Neural Network (RNN) (SHERSTINSKY, 2020), Transformers (VASWANI et al., 2017), Generative Adversarial Network (GAN) (GOODFELLOW et al., 2020), and Long Short Term Memory (LSTM) (HOCHREITER; SCHMIDHUBER, 1997). The context and purpose of the application determine the approach to be used. For example, the recent text generator ChatGPT (based on Generative Pre-Trained Transformer (GPT)-3.5 architecture (BROWN et al., 2020)) is made with Transformers. Text models such as Bidirectional Encoder Representations from Transformers (BERT) (DEVLIN et al., 2018) also use transformers. Generating faces or neural styles can be made with GANs, such as the online tool Deep Dream (MORDVINTSEV; OLAH; TYKA, 2015). For 2D pose estimation or 3D pose estimation, it is possible to obtain excellent results with CNNs; however, transformers also can be used in some cases, as in (EINFALT; LUDWIG; LIENHART, 2023; ZHENG et al., 2021).

2.4.1 Convolutional Neural Network

Convolutional neural networks are high-performance neural networks for solving problems in image, audio, and speech data contexts. They comprise layers such as the convolutional layer, pooling layer, and fully-connected layer. The convolutional layer is the first layer in the architecture of CNNs, followed by other convolutional layers or pooling layers, and the fully-connected layer is the last (O'SHEA; NASH, 2015). For example, figure 14 illustrates a CNN.

Figure 14 – The CNN is compound basically by convolutional, pooling, and fully-connected layers.



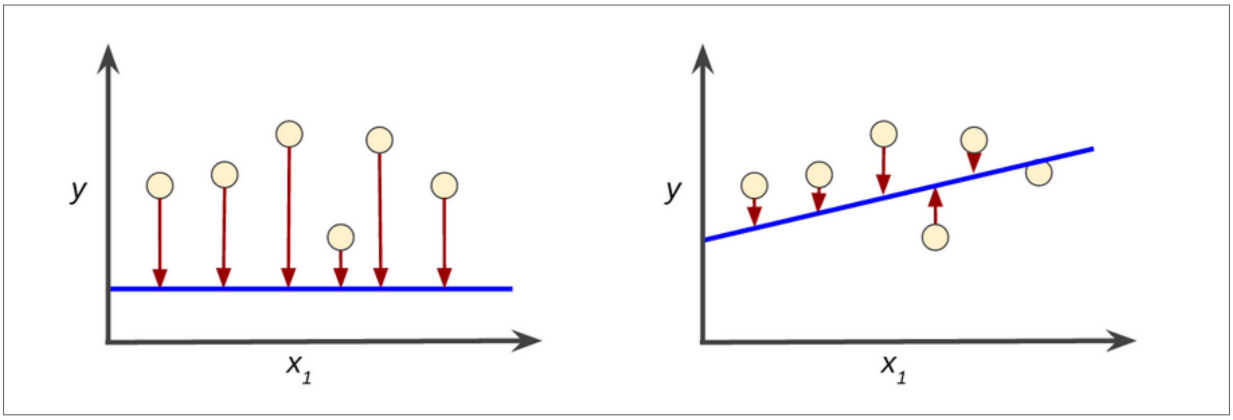
Source: VARSHINI et al. (2020)

The convolutional layer can be considered the CNN core. It is responsible for the majority of computation. The essential components are the input data, a filter, and a feature map. The convolutional layer will check the features of the respective input using a filter by the convolution process. After the convolution, transformation such as Rectified linear unit (ReLU) is applied, for example. The pooling layers are responsible for dimensionality reduction and downsampling. Pooling layers also use a filter for the input; instead of a convolution, they apply a filter that runs an aggregation function. The main pooling processes are max pooling and average pooling. They help to improve efficiency and limit overfitting. Finally, the fully-connected layer is responsible for the classification or regression process based on the features generated by the previous layers. Fully-connected layers use an activation function and are the last layer in the CNN architecture (O'SHEA; NASH, 2015).

2.4.2 Loss

A training model aims to obtain the minimum loss value. The loss is a penalty for wrong predictions, indicating how good or bad a model prediction is. A correct prediction, equal to ground truth, generates a zero value as a loss. However, a lousy prediction will have a high loss value. In this manner, the weights from a neural network are updated to minimize the loss. Figure 15 illustrates two models and their failures.

Figure 15 – The arrow's length is related to loss considering this respective single point; high lengths mean high loss. This way, the fitted model represented by the blue line has a higher loss value than all the aggregated arrows in the right image. It means the model at the right is better since it can obtain a minimum loss value.



Source: **GOOGLE** (2022)

There are several loss functions such as Mean squared error (MSE) (L_2 norm) for regression problems and cross-entropy for binary and multi-class classification. As an example, the MSE loss is given by

$$MSE = \frac{1}{N} \sum_{(x,y) \in D} (y - pred(x))^2, \tag{2.6}$$

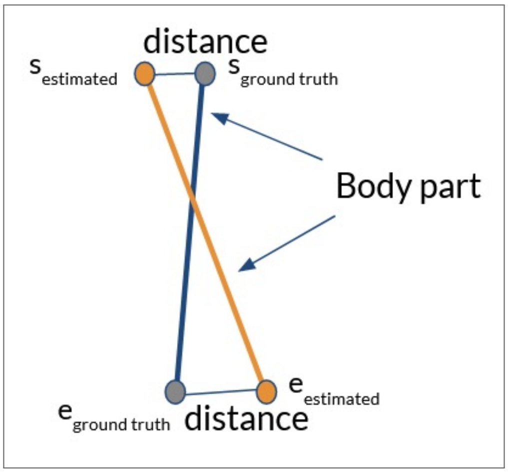
where x is the input data, $pred()$ is the function model obtained along training process, $pred(x)$ is the model's output, y is the ground truth, D is the dataset containing labels and input features, and N is the cardinality of the sample set. Note that our goal is to obtain a $pred()$ function able to have the minimum loss and also able to generalize for data outside the training set (ALZUBAIDI et al., 2021).

2.5 EVALUATION METRICS FOR 3D POSE ESTIMATION

3D pose estimation in this work consists of estimating 3D positions from points related to 2D human body poses obtained from Red Green Blue (RGB) images. Once received the 3D skeleton, it is necessary to compare it with the ground truth to check the performance of the 3D pose estimator. The most frequent evaluation metrics for 3D pose estimation are (Mean Per Joint Position Error (MPJPE) (WANG et al., 2021), Percentage of Correctly estimated Parts (PCP) (WANG et al., 2021), Percentage of Correct Keypoints (PCK) (WANG et al., 2021), Bone Error, Bone Std, Illegal Angle, and Mean of the Root Position Error (MRPE) (WANG et al., 2021).

The metric is chosen depending on the problem context and dataset. Some datasets, such as Human3.6M (IONESCU et al., 2013), have well-defined protocols such as P1, P2, and P3 combined with MPJPE to evaluate the estimator performance. In Campus (BERCLAZ et al., 2011) or Shelf-dataset (BELAGIANNIS et al., 2014), it is common to use the PCP metric. Figure 16 illustrates PCP.

Figure 16 – In this image, there are two body parts, the grey one is the ground truth, and the orange one is the estimation. PCP is obtained by comparing these body parts' start and endpoints. The analysis is correct if it is less or equal to a defined threshold.



Source: The author (2023)

PCP works by comparing skeleton parts, where each piece is composed of two joint points. PCP is calculated by comparing these collaborative pairs' positions, i.e. the estimated couple $(\hat{s}_n, \hat{e}_n)$ and the ground-truth pair (s_n, e_n). Considering that s is the start joint point and e is the endpoint, they are compared, and in case the distance is below a threshold, that part is considered correct according to the following equation:

$$\frac{||s_n - \hat{s}_n|| + ||e_n - \hat{e}_n||}{2} \leq \alpha ||s_n - e_n||. \tag{2.7}$$

Note that the larger the body part, the larger the threshold, so the sensitivity is more significant in small amounts with less tolerance. Furthermore, α is a threshold parameter, working as a factor to control the limit values to consider a body part as correct.

3 RELATED WORKS

This chapter describes approaches for multi-view multi-person 3D pose estimation. It is divided in two groups of methods: geometric, and deep learning based approaches.

3.1 MULTI-VIEW MULTI-PERSON GEOMETRIC METHODS

There are 3D pose estimation works in the multi-view multi-person context developed entirely using projective geometry concepts. This section details how these methods work.

The (BELAGIANNIS et al., 2014) is a work that addresses 3D pose estimation for multiple persons using images obtained from the multi-view scenario. As mentioned in (BELAGIANNIS et al., 2014), multi-person 3D pose estimation is a more challenging problem than the single-person context. There are issues such as occlusion, identifying the same person in different views, and a larger state space. To deal with these problems they create a reduced state space by triangulation of the skeleton joints from the camera views. Furthermore, they also introduce a novel 3D Pictorial Structure (3DPS). The 3DPS model from (BELAGIANNIS et al., 2014) infers a 3D pose from their reduced state space.

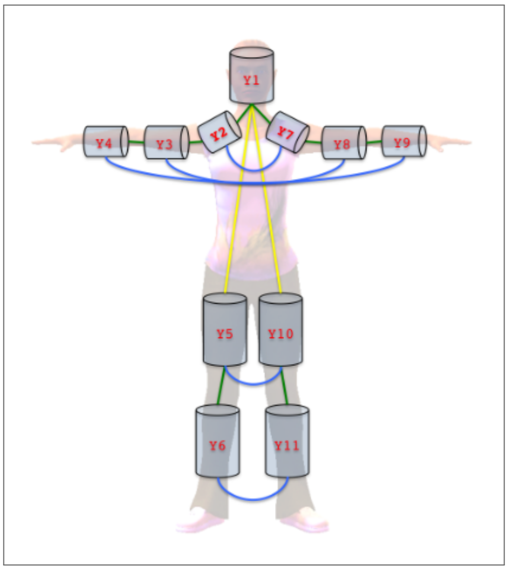
Beyond 3DPS model creation, they introduce the Shelf and Campus datasets. Both of them are for the multi-view multi-person scenario. However, Shelf comprises an indoor environment, while Campus in an outdoor dataset. To compare their approach, they performed 3D pose estimation on images containing only a single person as in KTH Multiview Football II dataset (BURENIUS; SULLIVAN; CARLSSON, 2013).

Using a novel 3DPS approach, they create a graphical model of the human body. The human body is represented by 11 variables as shown in Figure 17. The 3D pose is represented by the configuration of these variables.

They were able to achieve good results at Human-Eva I, being superior at (SIGAL et al., 2012) and got near results compared to (AMIN et al., 2013). They also obtained competitive results compared to (BURENIUS; SULLIVAN; CARLSSON, 2013) in the KTH dataset.

The authors in (BELAGIANNIS et al., 2014) were also responsible for some of the most famous datasets for multi-view multi-person 3D pose estimation, the Shelf and Campus datasets. In this manner, they started as the reference of state of the art. This work can be considered a starting point for multi-view multi-person 3D pose estimation methods. Some results are

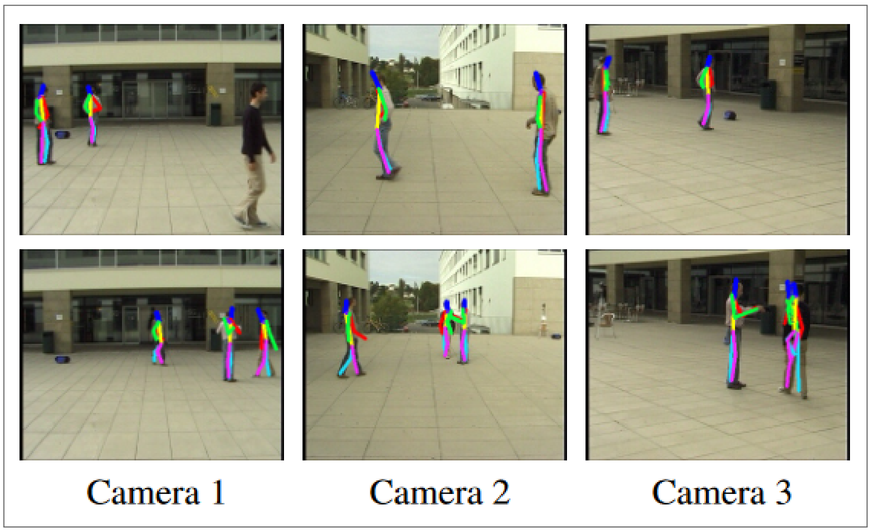
Figure 17 – Graphical representation of the human body involves 11 variables to depict different body parts. Kinematic constraints are denoted by green edges (for rotation) and yellow edges (for translation), whereas collision constraints are depicted with blue edges.



Source: BELAGIANNIS et al. (2014)

shown in Figures 18 and 19.

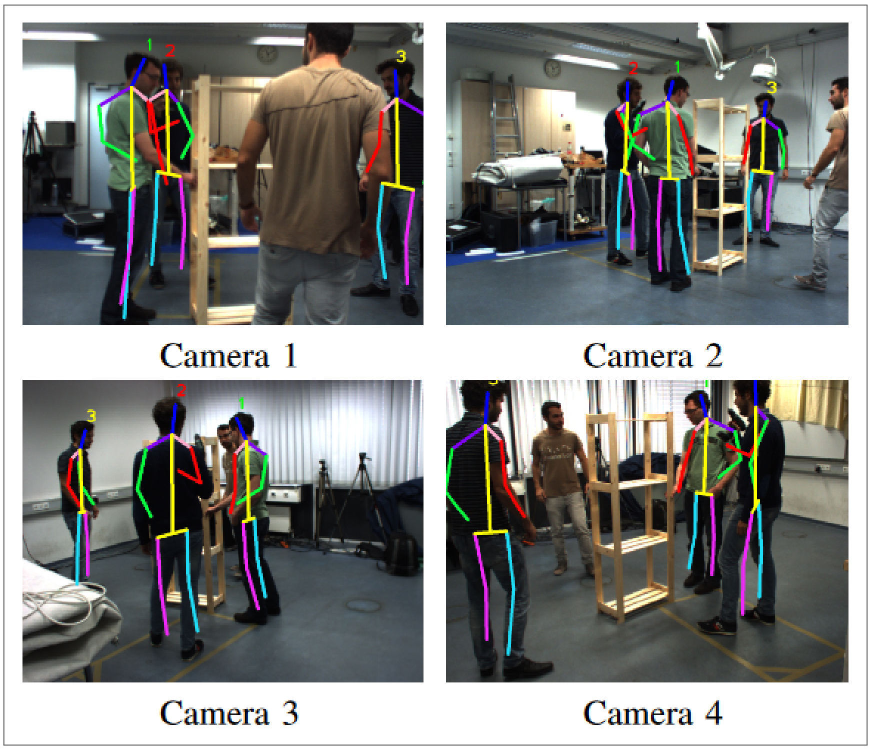
Figure 18 – 2D projections from 3D pose estimations in (BELAGIANNIS et al., 2014) related to Campus dataset.



Source: BELAGIANNIS et al. (2014)

The same authors released this improved version (BELAGIANNIS et al., 2015) of the (BELAGIANNIS et al., 2014). They also use a 3DPS model, however, now they use a Structured Support Vector Machine (svm) (SSVM) to improve the 3DPS model. They use multiple potential functions that must be weighted correctly. In this manner, they used Structured Support Vector Machine (ssvm) to learn the model parameters. They also increased the number of variables in the graphical model to represent the human body. Now they use 14 variables, instead of 11 as in the previous work.

Figure 19 – 2D pose projections from estimated 3D skeletons in 4 of 5 views from Shelf dataset.



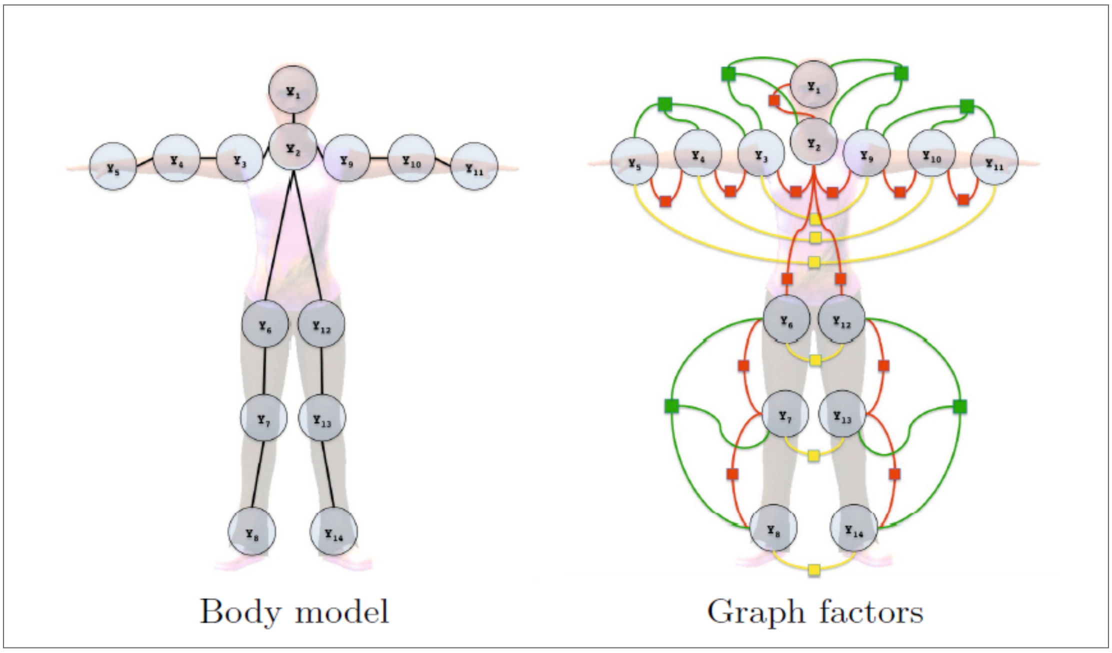
Source: BELAGIANNIS et al. (2015)

With this new approach they were able to achieve significant improvements compared with the previous work (BELAGIANNIS et al., 2014). They achieved the state-of-the-art at Kungliga Tekniska högskolan - Royal Institute of Technology (KTH) Multiview Football II (BURENIUS; SULLIVAN; CARLSSON, 2013) and also outperformed other methods in the Campus and Shelf datasets.

This work needs to match the 2D poses to perform 3D pose estimation. Furthermore, this 2D pose moves during time coherently. To estimate the 3D poses considering the consistency over time as mentioned before, (BELAGIANNIS et al., 2015) use a 3DPS (3D Pictorial Structure) model as shown in Figure 20. The temporal consistency improves the 3D pose estimation performance, once (BELAGIANNIS et al., 2015) identified the position before inference reducing the state space size.

They build a temporally consistent 3DPS using a Conditional Random Field (CRF). This field is composed of unary, pairwise, and ternary potential functions. These unary functions are responsible for describing the relationship between the random variables and the state space. The random variables are the joints from the body pose, and they take their values

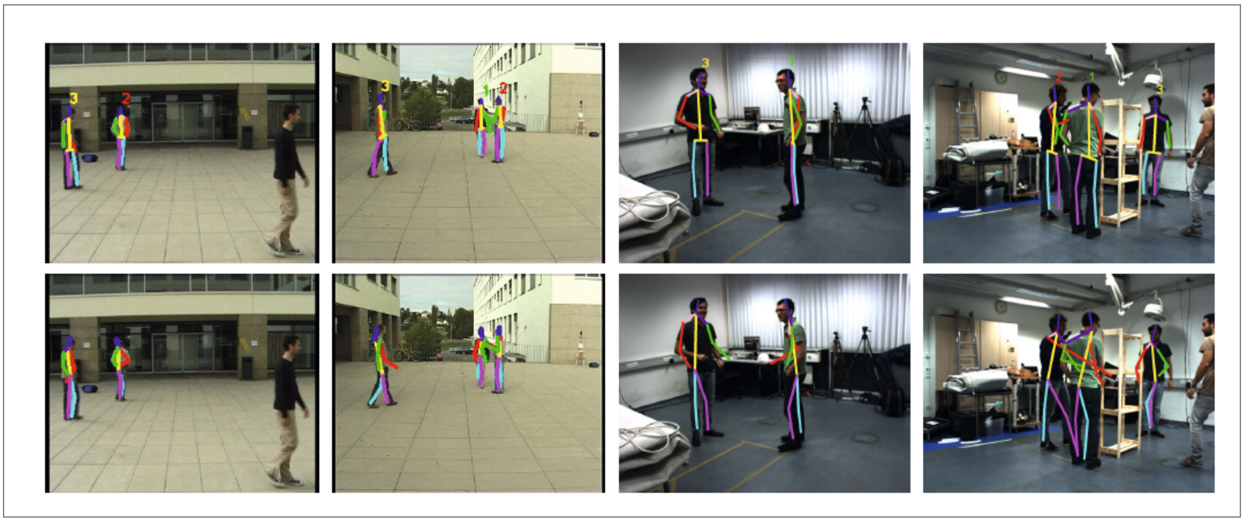
Figure 20 – On the left, human body graphical model with 14 variables used by (BELAGIANNIS et al., 2015) in to represent the body joints. On the right, graph factors are illustrated. The constrains are translation (red) and rotation (green) factors (edges). The yellow edges are the collision constrains. Source: (BELAGIANNIS et al., 2015).



Source: BELAGIANNIS et al. (2015)

from a defined state space. Furthermore, the pairwise and ternary potentials are responsible for modeling the interactions between the random variables (BELAGIANNIS et al., 2015).

Figure 21 – Qualitative comparison between (BELAGIANNIS et al., 2015) and (BELAGIANNIS et al., 2014). The top row are the (BELAGIANNIS et al., 2015) results and the bottom are related to (BELAGIANNIS et al., 2014). In all comparisons, the estimated poses from (BELAGIANNIS et al., 2015) are more precise due to reduced state space and temporal potential function regularisation.



Source: BELAGIANNIS et al. (2015)

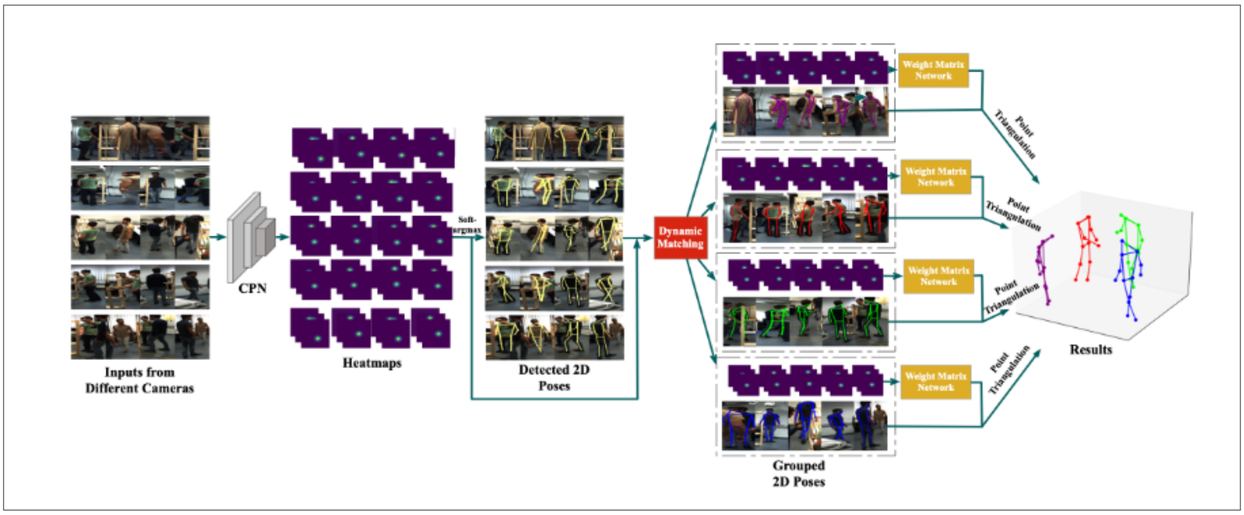
By using 3DPS and considering temporal consistency, they were able to outperform state-of-the-art methods in the Campus and Shelf datasets. The metric used for evaluation and comparison was the PCP metric. Some results are shown in Figure 21.

3.2 MULTI-VIEW MULTI-PERSON DEEP LEARNING BASED METHODS

This section details how multi-view multi-person supervised methods work. Differently from the geometric approaches, these methods need to use labeled data to learn the 3D pose estimation task.

The methods mentioned before use 3DPS to perform 3D pose estimation. The 3DPS approach has high computation costs and low accuracy in the task of joint detection. With the popularity of Deep Neural Networks, a new manner was established, considering three well-defined steps: 2D pose detection in each view, creation of a 2D pose cluster related to each person through a matching process, and finally 3D pose estimation.

Figure 22 – The framework of (HUANG et al., 2020). The images are the input into to the 2d poses estimator (HUANG et al., 2020) to get the heatmaps. Next, they apply soft-argmax on heatmaps to get the corresponding 2d poses. Then, they feed both heatmaps and 2D estimated poses into matching module. They then sent the heatmaps into a network to get weight matrices. Finally, each cluster is sent to a weight-sharing 3d pose estimator to get the 3D pose.

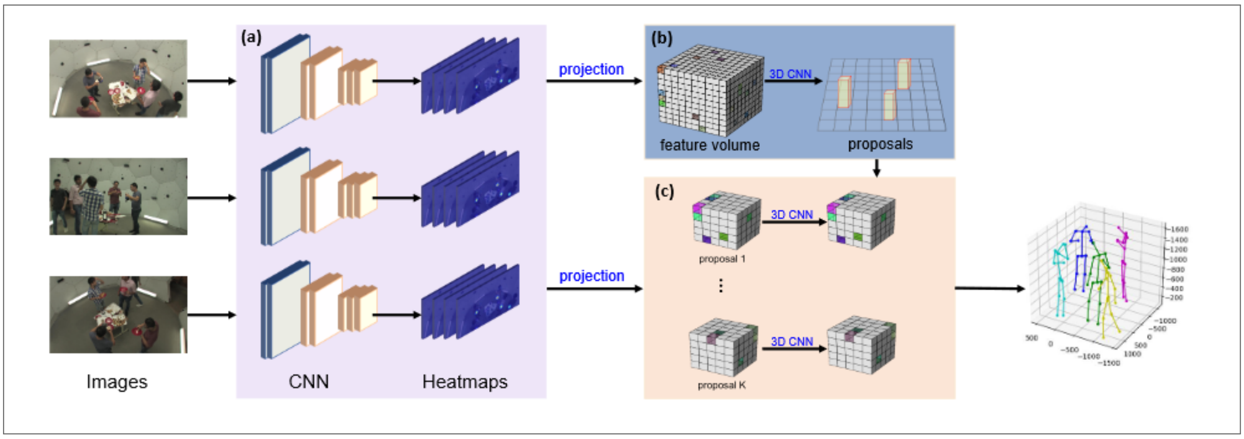


Source: HUANG et al. (2020)

Several methods work by running this process in a cascade manner through each one of these steps. This way, the 2D images and camera parameters feed the steps. One point to consider is that the steps can be correlated, so a change in one can affect the others. To avoid these problems, (HUANG et al., 2020) propose an end-to-end approach that joins the steps in a single model as shown in Figure 22. The step of the matching process disjoints the pipeline. To solve this, (HUANG et al., 2020) inspires Capsule Networks to create a dynamic matching process working as path gradient flow select, directing the paths from steps 1 to 3. Beyond that, they propose a novel matching algorithm to deal with a large number of cameras.

The (TU; WANG; ZENG, 2020) method is also known as VoxelPose and is shown in Figure 23. Previous works needed to obtain cross-view correspondences based on 2D poses from challenging environments. This way, these poses sometimes are incomplete or noisy. VoxelPose has a different approach that deals with incomplete and noisy 2D poses, directly working in the 3D space, thus avoiding to handle wrong 2D poses from the camera views.

Figure 23 – Overview of (TU; WANG; ZENG, 2020) approach. There are three well defined steps: (a) first, they estimated 2D pose heatmaps; (b) second, they build a feature volume from a 3D space warping the heatmaps, so they fed a Cuboid Proposal Network to find people instances; c) Finally, they build a finer-grained feature volume and obtain the 3D human pose.



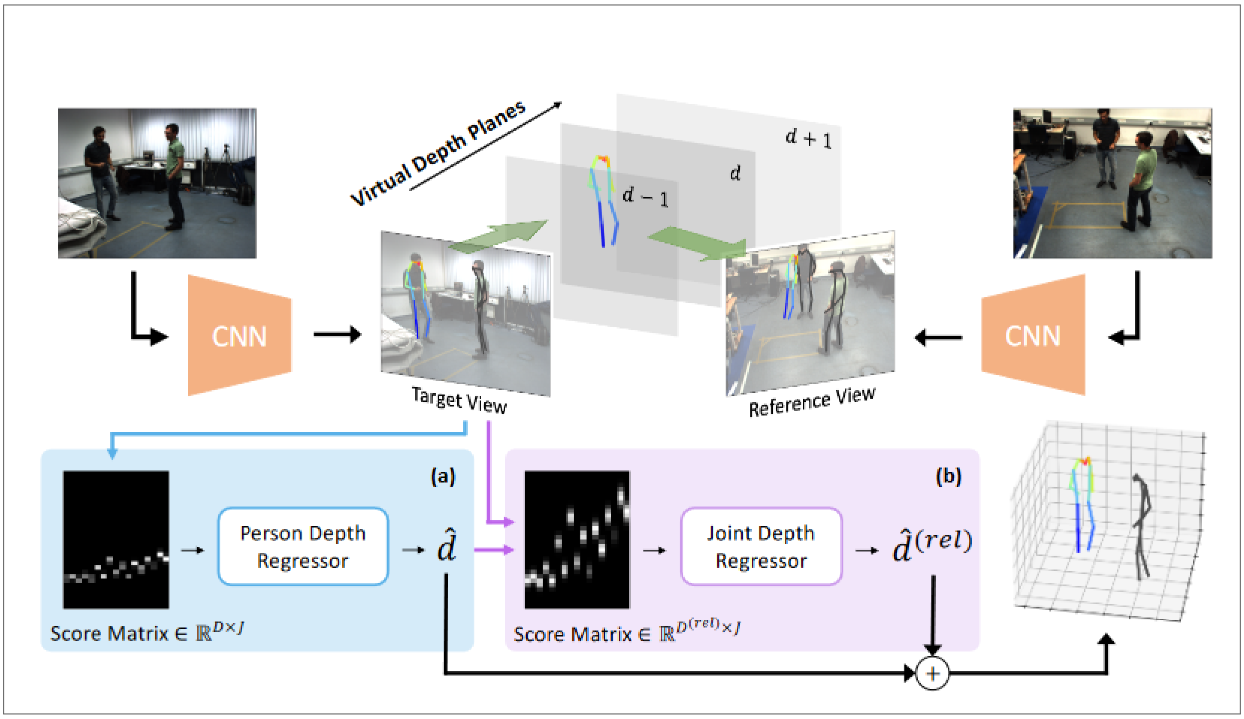
Source: TU; WANG; ZENG (2020)

VoxelPose works directly in 3D through a method in which the features from the camera views must be aggregated in the 3D voxel space and serve as input to a Cuboid Proposal Network (CPN) to locate the people. After that, they then propose a Pose Regression Network (PRN) to obtain the detailed 3D poses. VoxelPose is robust to occlusion, which is an important aspect considering the practical scenarios.

3D pose estimation methods usually obtain cross-view correspondences to obtain clusters of 2D poses of people present in the multi-view scenario. From these clusters, they can estimate the 3D body pose. Algorithms for computing cross-view correspondences in multi-view scenarios may deal with challenging environments that give rise to incorrect correspondences. This wrong matching can impact 3D pose estimation.

The work of (LIN; LEE, 2021) has an approach based on plane sweep stereo, which performs cross-view matching and 3D pose estimation in a single task, different from methods based on multi-stage solutions.

Figure 24 – Overview of (LIN; LEE, 2021). First is estimated a 2D pose estimation for each view. Second they use the plane sweep algorithm to obtain cross-view consistency for the highlighted target person. The person-level depth is obtained in (a), and the joint-level is obtained in (b). Combining person-level and joint-level (LIN; LEE, 2021) is able to estimate the 3D pose.



Source: LIN; LEE (2021)

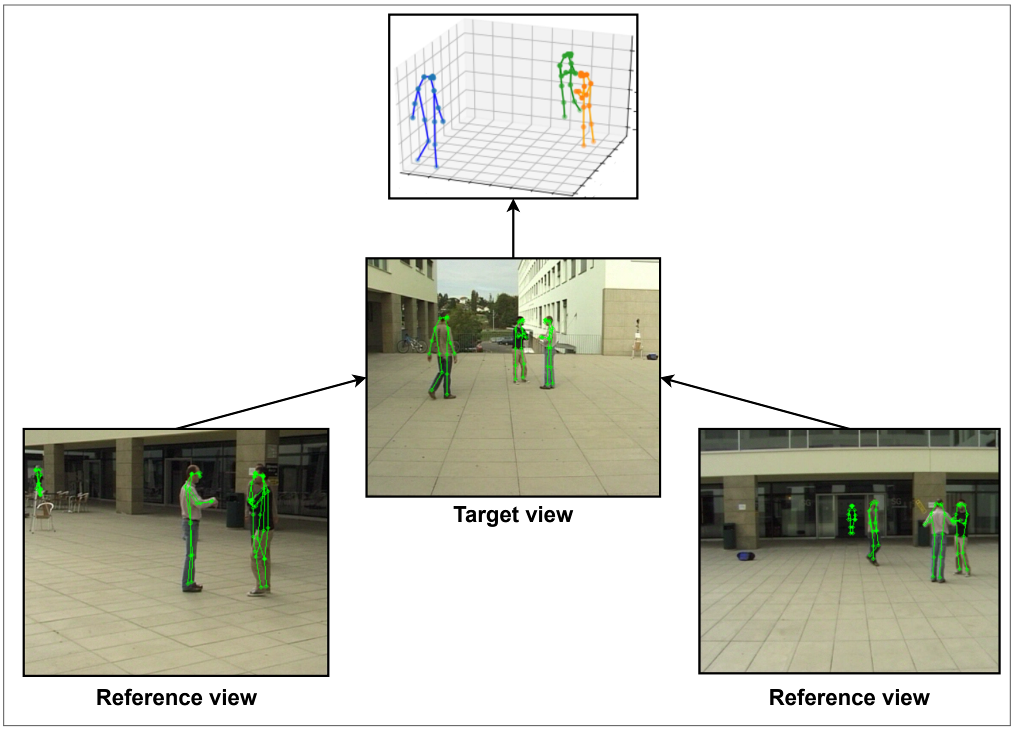
They propose a method in which a target view is defined among the available views, and the others are considered reference views. They obtain the depth of each joint from the 2D body poses of the target view. Using a back-projection method, they use the multiple reference views to enforce consistency across views.

This method is considered a coarse-to-fine scheme. First, they estimate the person-level depth, and so the joint-level depth as illustrated in Figure 24. Combining them they are able to perform the 3D pose.

4 UNSUPERVISED MULTI-VIEW MULTI-PERSON 3D POSE ESTIMATION

This chapter proposes a unsupervised 3D pose estimation of multiple persons in a multi-view scenario. This work develops two approaches and compare each other. The first is using backprojecting as matching process, and the next is using ground point matching. Beyond that, the backprojecting approach has a MSE loss, instead of a L_1 smooth loss as the second approach. Both methods aim to obtain 3D body skeletons using the plane sweep stereo work (LIN; LEE, 2021) as shown in Figure 25. With (LIN; LEE, 2021) technique, the 3D poses are generated. The obtained poses are then projected onto each one of the reference views, so these 2D projections are compared with the respective matched 2D body skeletons. The back-projection method is illustrated in Figure 26 and ground point matching approach is detailed in Figure 27.

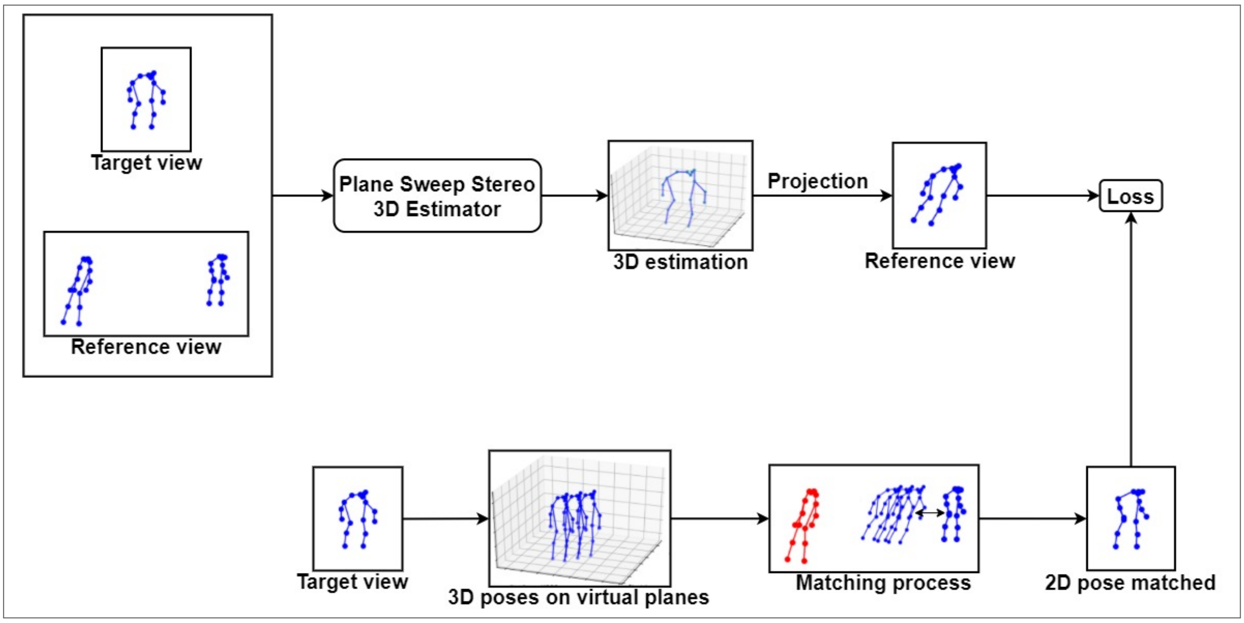
Figure 25 – First, we define a target view and the reference views. Then we estimate the 3D joints using these defined views as our input. Each predicted depth is related to the 2D pose estimations from the target view, so our predictions are based on the target view skeletons.



Source: The author (2023)

One of the key points in this work is the matching process between target and reference views. The matching is obtained using two methods: backprojecting the estimated 2D pose (LIN; LEE, 2021), and ground points attached to people for which we wish to estimate the 3D pose (LIMA et al., 2021).

Figure 26 – Considering the multiple views available, as in (LIN; LEE, 2021), we define a target view, and all the others are determined as reference views. Our overview solution image has two views: the target view and only one reference view. Our approach follows two sequential steps: the first is to estimate the 3D pose using (LIN; LEE, 2021), go along with the projection of this 3D pose onto the reference view. The second is to utilize back-projection to make the matching process, so we establish the 2D pose compared with the 2D projection obtained from 3D estimation. Comparing these poses, we have a reprojection error loss.



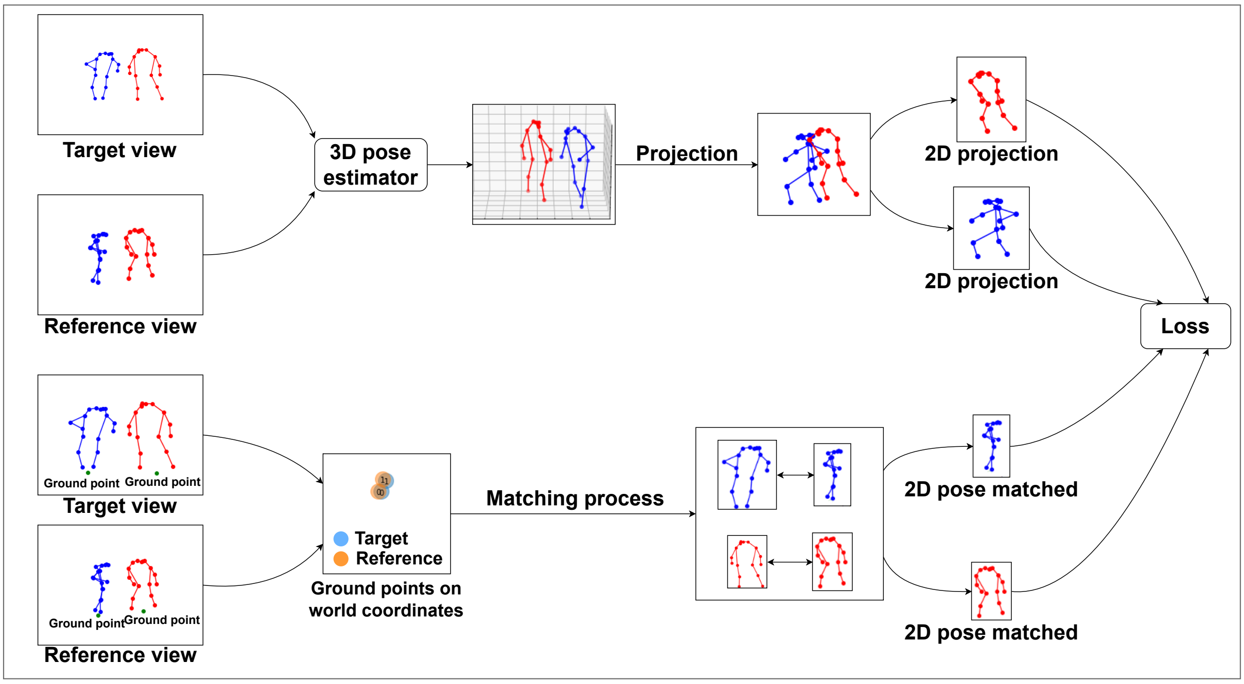
Source: The author (2023)

Related to backprojecting, the 2D pose from target view is projected in sucessive virtual depth planes, so these 3D skeletons are projected in the respective reference views. We compare the skeletons, and the nearest 2D pose in the reference view is our matched pose. In the ground point case, it is estimated the 2D ground points attached to each 2D body skeleton, then these points are projected onto world points using a homography matrix, and finally the Hungarian algorithm is used for matching the poses using the Euclidean distances among the calculated ground points.

Beyond the matching process, other key points of this work are the reprojection error and the loss. The backprojecting approach uses a MSE loss, and the ground point a smooth L_1 loss to compare the projected and the matched 2D poses, and so optimize the neural network. Usually the MSE loss is the typical approach to compare 2D poses, as shown in (LI et al., 2021). However, using smooth L_1 loss achieves higher performance. Another works also use smooth L_1 loss, such as (BRYNTE; KAHL, 2020) for calculating the reprojection error. The plane sweep approach also uses smooth L_1 loss to compare the pose estimations (LIN; LEE, 2021). This work improved significantly the results by using the smooth L_1 loss.

It is important to mention that the matching process is used only during the training

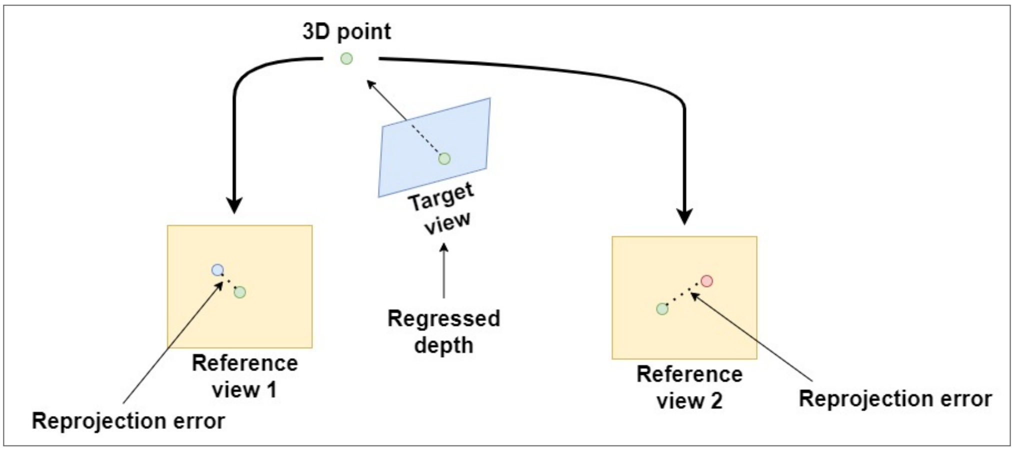
Figure 27 – We have well-defined target and reference views. Using the 2D poses estimated in each view, we perform 3D pose estimation as in (Lin and Lee, 2021). Each 3D skeleton is projected onto the reference view, and we compare them with the matched 2D poses. These matching 2D poses are obtained using ground points. For each 2D pose, we have ground points associated and, utilizing a homography matrix as in (Lima et al., 2021), we project these points onto world coordinates. Taking the Euclidean distance, we build a cost matrix used on the Hungarian Algorithm to perform the matching. With the 2D poses matched, we compare them with a smooth L1 loss (Girshick, 2015).



Source: The author (2023)

process. Once the model is trained, the 3D poses can be inferred using the neural network structure provided by (LIN; LEE, 2021).

Figure 28 – Overview of the reprojection process. Once we have a point in a target view, we estimate its depth using [12] and project this point onto both the reference views. The squared Euclidean distance between the points is the reprojection error [7]. The blue and orange points represent the real position of the target point in the respective reference view. We compare the projected point (green point) with the estimated position. We are not using ground truth to verify if the 3D point is in the correct position; instead, we use the reprojection error.



Source: The author (2023)

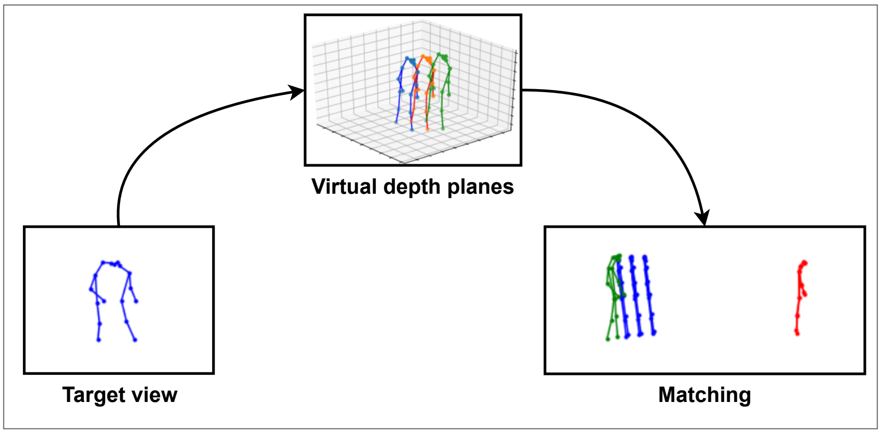
4.1 REPROJECTION ERROR

At this section, it is brief detailed the reprojection error (HARTLEY; ZISSERMAN, 2003). This key concept is used for computing the loss. In this case, the reprojection error is the manner of how to compare the 2D poses during the training process. This way, it is possible to train the neural network model without needing to use 3D pose labeled ground-truth data.

A 3D pose is estimated for each 2D pose on the target view using (LIN; LEE, 2021). The generated 3D poses are projected onto the respective reference views, so that pose goes from world points (3D pose) to image points (2D pose). This way, the projected poses are compared with the matched 2D poses in the reference views. The comparison is made using MSE loss and smooth L_1 norm, as it is described in the next sections.

Furthermore, it can be established that the reprojection error is the process of comparing the projection of a 3D point onto a 2D point (image point) with the original 2D position of that point in a given image as illustrated in Figure 28.

Figure 29 – The estimated 2D pose in the target view is back-projected onto successive virtual depths planes, and each 3D skeleton is projected onto the reference view. We compare the skeletons, and the nearest 2D pose in the reference view is our matched pose. This matched pose is compared with the 2D projection of the estimated 3D pose.



Source: The author (2023)

4.2 UNSUPERVISED MULTI-VIEW MULTI-PERSON 3D POSE ESTIMATION WITH BACK-PROJECTION MATCHING

4.2.1 Matching process

Based on (LIN; LEE, 2021), we estimate 2D poses from all the views using an off-the-shelf method (SUN et al., 2019), and after that, we perform a back-projection for each 2D pose of the target image using virtual depth planes as shown in Figure 29. Finally, each 3D pose in these depth planes is projected onto the reference views, and we measure the distance between this projected 3D pose and the estimated 2D poses in the reference view:

$$m = \arg\min_{r} \sum_{i=1}^{J} d(r_i, p_i), \tag{4.1}$$

where J is the number of joints, $d(x, y)$ is the distance between the joints x and y, $\{r\}$ is the set of the 2D poses from the reference view, p is the projected pose in reference view, and m is the nearest 2D pose in the $\{r\}$ set.

4.2.2 Loss function

In (LIN; LEE, 2021) they compute two losses, one for person position (center hip joint) and another for joints positions. With the regressed depth, they obtain the 3D hip point and joints. Then, they compare the estimates with the 3D ground truth. UMVpose uses the regressed depth to generate a 3D point (using the center hip) related to the person's position and a 3D pose with all the joints. We project the person's 3D location and the estimated 3D pose onto each reference view. Using the concept of reprojection error, we compute a loss comparing the target 3D estimate projected onto the reference view with the matched 2D pose. As in (LIN; LEE, 2021), we use two losses, a position loss (related to hip point) and a joint loss. Both losses are computed using MSE.

The position loss is given by

$$\mathcal{L}_{pose} = \sum_{r=0}^{R} \frac{1}{P} \sum_{i=1}^{P} (position_r(i)_{proj} - position_r(i)_{ref})^2, \tag{4.2}$$

where P is the number of persons in the target view, $position_r(i)_{proj}$ is the projected pose and $position_r(i)_{ref}$ is the matched pose in the reference view, and R is the number of reference views.

The joint loss is obtained by

$$\mathcal{L}_{joint} = \sum_{r=0}^{R} \frac{1}{P} \sum_{i=1}^{P} \sum_{j=1}^{J} (joint_{r,j}(i)_{proj} - joint_{r,j}(i)_{ref})^2, \tag{4.3}$$

where P is the number of persons in the target view, $joint_{r,j}(i)_{proj}$ are the 17 joints projected onto the reference view and $joint_{r,j}(i)_{ref}$ are the joints from the matched skeleton in the reference view.

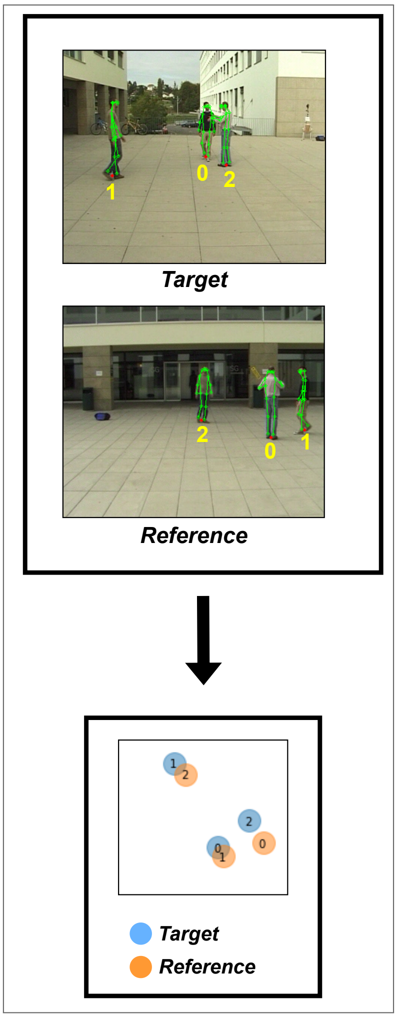
4.2.2.1 Regularizer term

We also use a regularizer term, more precisely the Kullback-Leibler (KL) one (ERVEN; HARREMOS, 2014), inspired by (NIBALI et al., 2018). We get the keypoints positions, and we multiply each coordinate by a Gaussian distribution $\mathcal{N}(0, \sigma)$, so we apply this to projected and matched 2D poses. We apply KL divergence to these poses multiplied by λ. We also make this with the center hip point in $\mathcal{L}_{position}$. We use the parameter values $\sigma = 1$ and $\lambda = 1$, since they provide the best results in (NIBALI et al., 2018).

4.2.2.2 Optimizer

3D pose estimation learning methods commonly use the Adam optimizer (KINGMA; BA, 2014). Unfortunately, Adam takes a long time to converge, so we decided to use AdaBelief (ZHUANG et al., 2020), which has three key features: fast convergence, good generalization, and training stability. AdaBelief was faster in convergence than Adam. Therefore, we could see progress early. Furthermore, we could make the analysis faster than the Adam optimizer when we performed different tests.

Figure 30 – The matching process occurs using ground points. Each person has a ground point, and it is projected onto world coordinates. Next, we measure the distance between these points to obtain a cost matrix. Finally, we use the Hungarian algorithm to perform matching between target and reference views based on our cost matrix.



Source: The author (2023)

4.3 UNSUPERVISED MULTI-VIEW MULTI-PERSON 3D POSE ESTIMATION WITH GROUND POINT MATCHING

4.3.1 Matching process

In this section it is detailed how to execute the matching process using an approach based on ground points as shown in Figure 30. It is important to mention that the matching process is used only during the training process. When performing inference, it is not needed to use the ground point matching technique.

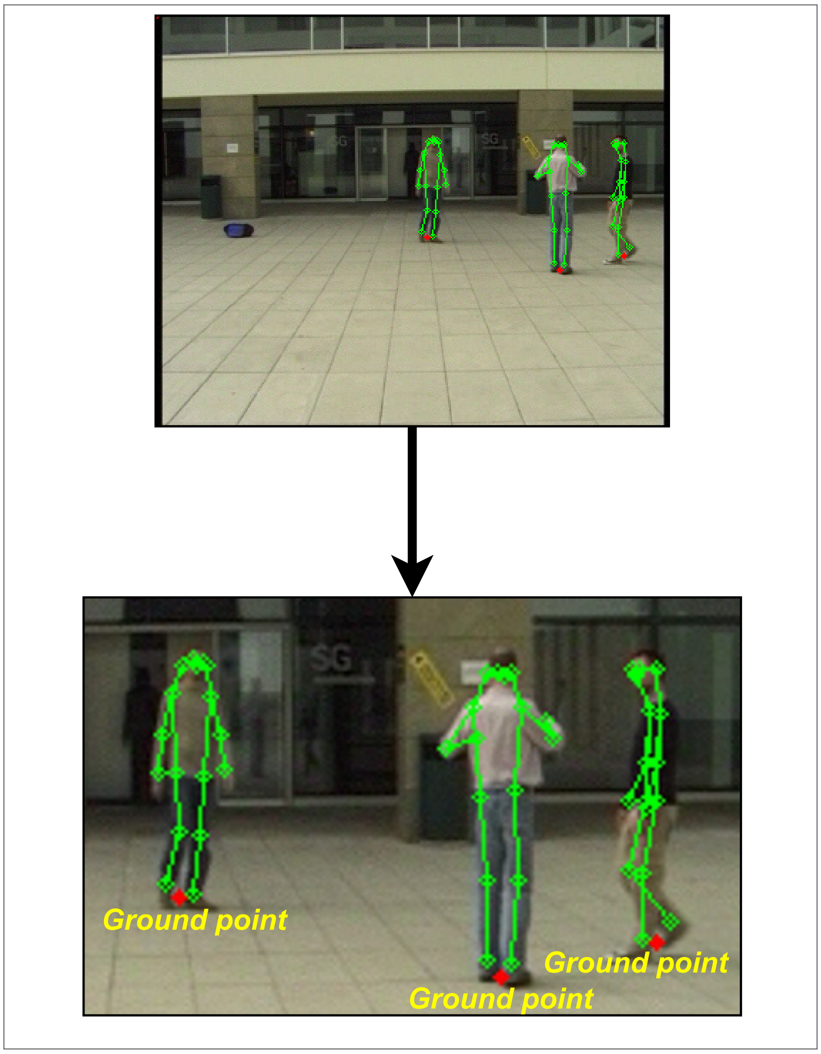
It is performed unsupervised 3D body pose estimation using the reprojection error. Furthermore, it is made a comparison betwen poses using ground point matching. The matching process aims to identify the corresponding 2D pose in a reference view related to a projected 2D pose from respective 3D body skeleton. This way, it is clear the relevance of the matching process, since it is necessary to compare the corresponding 2D poses in order to obtain a coherent training process for generating the 3D poses.

The matching process is build up using ground points approach as in (LIMA et al., 2021), as illustrated in Figure 31. First, the ground points are obtained from 2D poses. They are estimated considering a line between the right and left ankle joints, so we get the middle point of this line, and, taking an offset δ in the ground direction, it generates the ground point. The offset is calculated using the own 2D pose. Considering the 2D body skeleton, we take the highest and lowest value from x and y coordinates. This way, we have the bounding box related to that 2D pose as shown in (XIU et al., 2018). Once the bounding box is estimated, we get the maximum y value of the bounding box ($bb_{y_{max}}$) and the highest y value between the right (ra_y) and left (la_y) ankle joints, and then calculate δ as:

$$\delta = bb_{y_{max}} - max(la_y, ra_y). \tag{4.4}$$

The 3D ground points are located in the ground plane. Therefore, their Z component in world coordinates must be zero. This way, the ground points in image coordinates are projected onto the world coordinate system using a homography matrix **H** as in (LIMA et al., 2021):

Figure 31 – We assign a ground point to each 2D pose. We obtain this ground point as described in (LIMA et al., 2021). We estimate a bounding box for each 2D pose and build a line between the ankles. So we take the middle point and apply an offset in the direction of the ground. Our goal is to represent each person by this point. Therefore the ground points are our reference to match the 2D skeletons of the target and reference views.



Source: The author (2023)

$$
\begin{aligned}
s\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} &= \mathbf{K}[\mathbf{R}^1\mathbf{R}^2\mathbf{R}^3\mathbf{t}]\begin{pmatrix} X \\ Y \\ 0 \\ 1 \end{pmatrix} \\
&= \mathbf{K}[\mathbf{R}^1\mathbf{R}^2\mathbf{t}]\begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} \qquad (4.5)
\end{aligned}
$$

where **K** represents the intrinsic parameters matrix, while [**R**|**t**] is the extrinsic parameters matrix. Note that the world ground points are represented by $(X, Y, 0)$ coordinates, and (x, y) are the image points. Each $\mathbf{R}^i$ corresponds to the i-th column of **R**.

As an example, in Figure 30 we take a pair of views (target and reference) and project the ground points from 2D image coordinates of each view onto world coordinates. Then we take the Euclidean distance between the projected ground points in world coordinates, so it is created a cost matrix. The cost matrix rows are the corresponding distances between target view ground points and reference view ground points. This way, each row is the Euclidean distance between the world ground point of a person in the respective target view and all the ground points of other persons on a given reference view. The elements of the cost matrix can be described as $d_{\left(target_{person_i}, reference_{person_j}\right)}$, where d is the Euclidean distance between the world ground points correspondent to two persons in the respective views. The matching process is performed using the Hungarian algorithm (KUHN, 1955). Performing matching using ground points is a robust approach, because it is compared only one sigle point (ground point) instead of all the joints present in a 2D skeleton. Furthermore, it is not necessary to perform several 3D projections of all 2D points as seen in the back-projection method presented in (LIN; LEE, 2021). In this case, the matching using ground points is more simple, once it only needs to estimate the ground point from the 2D pose and project it onto world coordinates. Differently from the back-projection method, ground point matching is more robust, since back-projection can generate false positive matchings when the person in target view is not in the reference views. The cost matrix is as follows:

$$Cost\ matrix = \begin{bmatrix} d_{11} & d_{12} & ... & d_{1n} \\ d_{21} & d_{22} & ... & d_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ d_{n1} & d_{n2} & \dots & d_{nn} \end{bmatrix}. \tag{4.6}$$

4.3.2 Loss Function

Based on the neural network structure presented in (LIN; LEE, 2021), this work suggests a new loss while retaining the existing structure. The goal is to eliminate the need for 3D labeled data by utilizing reprojection errors. By matching poses and using the 2D projections obtained from the estimated 3D pose, it is possible to create a loss function that does not require ground-truth information. Unlike works that compare 2D poses using MSE, our approach recommends using the smooth L_1 loss instead of the MSE loss.

Loss functions that involve the comparison of 2D poses typically utilize the MSE loss, as demonstrated in (LI et al., 2021). Nonetheless, some studies, such as (BRYNTE; KAHL, 2020), employ the smooth L_1 loss to compute the reprojection error. Additionally, the neural network in (LIN; LEE, 2021) also uses the smooth L_1 loss.

The smooth L_1 loss is defined as

$$\text{smooth}_{L_1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise.} \end{cases}$$

In this work context, the adoption of the smooth L_1 loss resembles the use of an M-estimator function, for instance the Huber function, to compute the reprojection error, which enhances the resilience of our approach to outliers. Additionally, it is conducted experiments with the Kullback-Leibler loss regularizer and it is employed AdaBelief as the optimizer.

As noted in (LIN; LEE, 2021), there exist two distinct losses, one dedicated to the individuals positions and the other to the positions of their joints. Specifically, the person position loss concerns the central hip joint, i.e. a single point, while the joint loss pertains to all of the person's joints. To compute these losses, this approach suggests utilizing reprojection error instead of comparing estimations with 3D ground truth. As a result, it obtains two distinct losses: one for the pose (person position) and another for the joints (joint position). These losses correspond to two distinct neural networks, namely the person-level depth regression network and the joint-level depth regression network, as shown in (LIN; LEE, 2021). The pose loss is defined as follows:

$$\mathcal{L}_{pose} = \sum_{r=0}^{R} \frac{1}{P} \sum_{i=1}^{P} ||position_r(i)_{proj} - position_r(i)_{ref}||_{s1}. \tag{4.7}$$

Equation 4.7 outlines the pose loss calculation, where P denotes the number of individuals in the target view, $position_r(i)_{proj}$ represents the projected pose, and $position_r(i)_{ref}$ means the matched pose in the reference view, obtained using ground point matching. The variable R denotes the number of reference views, while the index $s1$ relates to the use of the Smooth L_1 loss. Additionally, the joint loss is determined by

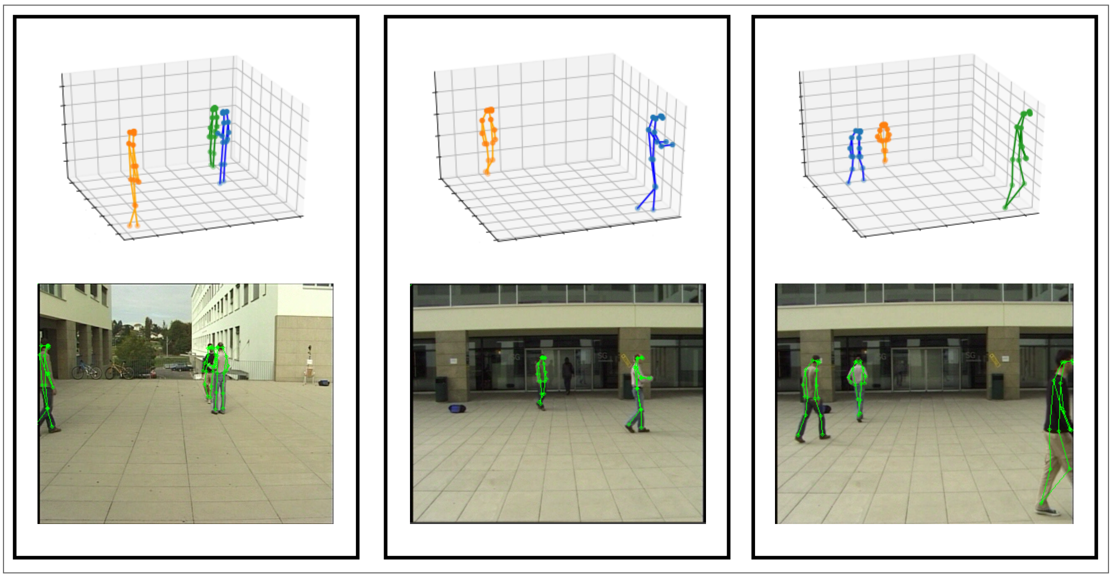
$$\mathcal{L}_{joint} = \sum_{r=0}^{R} \frac{1}{P} \sum_{i=1}^{P} \sum_{j=1}^{J} ||joint_{r,j}(i)_{proj} - joint_{r,j}(i)_{ref}||_{s1}, \tag{4.8}$$

where J represents the overall number of joints, $joint_{r,j}(i)proj$ means a joint that is projected onto the corresponding reference view, and $jointr, j(i)_{ref}$ denotes a joint from the matched 2D pose in the reference view.

5 RESULTS AND EXPERIMENTS

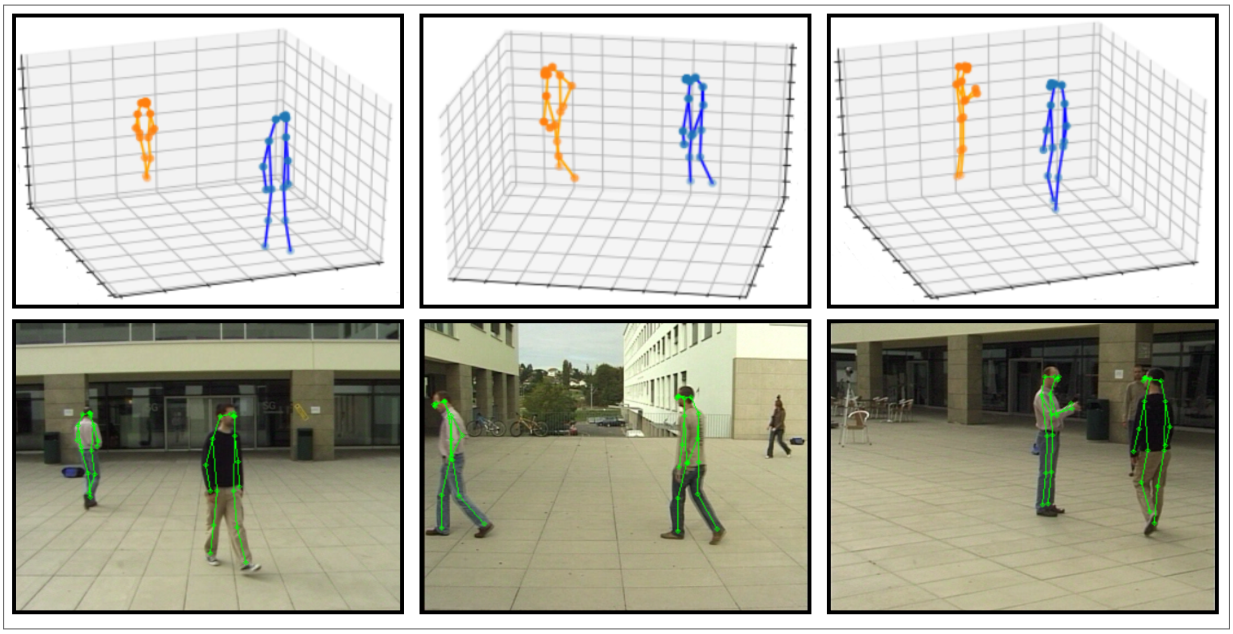
This section entails a detailed presentation of the experimental outcomes of our 3D pose estimation study, represented in Figures 32 and 33. Initially, it generates 3D poses for various individuals in a multi-view setting. Then, it proceeds with the training process by leveraging a publicly available dataset and evaluating the model's efficacy through the Percentage of Correctly Estimated Parts (PCP) metric (WANG et al., 2021). 3D pose estimation is applied by considering all humans together, with unknown identities. Each actor is associated with a person from the dataset. The evaluation of 3D pose estimation is performed by assessing the performance of each actor. After estimating the 3D pose for each actor, their performance is evaluated using PCP (Percentage of Correct Parts) individually. Therefore, the results consist of evaluating the 3D pose estimation for each actor. The configuration of the plane sweep pose estimator (number of epochs, batch size, learning rate, and other hyperparameters) is the same as described in (LIN; LEE, 2021). Finally, this work conducts an exhaustive assessment of its approach, comparing its PCP results with previous works that utilized geometric and neural network techniques. This analysis provides a comprehensive understanding of the presented model's performance in the context of existing processes and its potential for improving future research.

Figure 32 – The figure has several illustrations of this work 3D pose estimation using the Campus Dataset. The results showcase the estimated 2D skeleton and its corresponding 3D pose, which provides numerous instances for qualitative analysis.



Source: The author (2023)

Figure 33 – The Campus Dataset contains instances of 3D pose estimation using this work, with green skeletons representing the estimated 2D poses and the corresponding 3D poses displayed above for qualitative analysis.

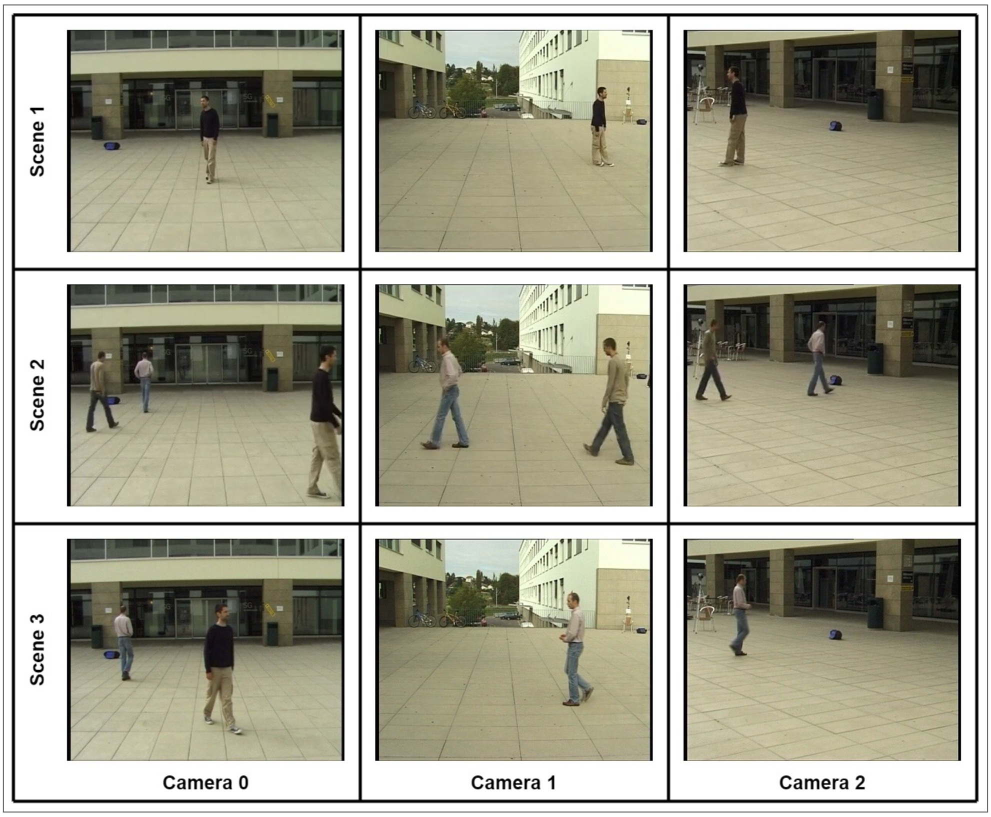


Source: The author (2023)

5.1 DATASET

In this study, the Campus dataset was utilized to estimate 2D poses using High-Resolution Net (HR-Net), a deep learning architecture that was pre-trained on the Microsoft Common Objects in Context (MS-COCO) dataset. This dataset is used because it is one of the principal benchmarks for 3D pose estimation of multiple people in a multi-view scenario. It does not undergo any pre-processing on the images. The Campus dataset is a widely-used benchmark dataset for multi-view, multi-person scenarios, which contains videos of three actors interacting with each other in an outdoor environment, captured by three cameras as illustrated in Figure 34. The Campus dataset contains 2000 frames. Although the dataset's 3D ground truth annotations are incomplete, the researchers used synthesized 3D MoCap poses from (LIN; LEE, 2021) to train their model. They compared their approach with the results obtained using geometric and supervised methods that relied on the Campus dataset, and evaluated their model's performance on frames 350-470 and 650-750, consistent with previous studies (DONG et al., 2019), (HUANG et al., 2020), (TU; WANG; ZENG, 2020). The training set consists of the remaining frames. It should be noted that the 2D poses in the dataset consist of 17 joints.

Figure 34 – The Campus Dataset includes a collection of footage captured by three calibrated and synchronized cameras, each offering a unique perspective of the same scenario. These cameras have successfully captured a range of diverse scenes, providing different viewpoints of the same events. For instance, in scene 2, camera 0 has recorded the presence of three individuals, while camera 1 and 2 only managed to capture two individuals. Likewise, in scene 3, camera 0 has recorded the presence of two people, while cameras 1 and 2 have only managed to capture one person. The multi-perspective footage captured by the Campus Dataset's cameras offers researchers a valuable resource to study and analyze the dynamics of events, as well as to develop and test novel computer vision algorithms.



Source: The author (2023)

5.2 METRICS

In this study, it has utilized the PCP as the evaluation metric, which has also been commonly employed in other related works. Given the dataset's complexity and the accuracy of 2D part detectors, the PCP (Percentage of Correct Parts) score yields more informative results when compared to metrics based solely on the Euclidean distance. In order to ensure a fair comparison of its results with those of previous studies, it has adopted the same metric. By using a standardized metric, it can ensure that it results are directly comparable to those of other studies, which will enable to draw meaningful conclusions and make informed recommendations based on the analysis of the collected data. PCP (WANG et al., 2021) is given by

$$\frac{||s_n - \hat{s}_n|| + ||e_n - \hat{e}_n||}{2} \leq \alpha||s_n - e_n||, \tag{5.1}$$

where s_n and e_n are the start and end coordinates of ground truth n-th body part, $\hat{s}_n$ and $\hat{e}_n$ are the corresponding estimations, and α is a given threshold parameter, in our case $\alpha = 0.5$.

Table 1 – Comparing PCP on Campus Dataset of ground point with backprojection approaches.

Method	Actor1	Actor2	Actor3	Average	Std
Backprojection with Adam (MSE loss)	78.0	85.1	83.0	82.0	3.6
Backprojection with Adabelief (MSE loss)	96.9	87.8	88.9	91.2	5.0
Backprojection with Adabelief and KL regularizer (MSE loss)	93.3	86.8	89.4	89.8	3.3
Backprojection with Adam (Smooth L_1 loss)	98.6	92.7	98.3	96.5	3.3
Backprojection with Adabelief (Smooth L_1 loss)	98.2	92.9	98.2	96.4	3.0
Backprojection with Adabelief and KL regularizer (Smooth L_1 loss)	97.4	92.5	98.6	96.2	3.2
Ground point matching with Adam and Smooth L_1 loss	98.4	93.4	98.6	96.8	2.9

Source: The author (2023)

5.3 COMPARISON BETWEEN BACKPROJECTION AND GROUND POINT MATCHING APPROACHES

Given the approaches of this work based on backprojection and ground point matching algorithms, it conducted a series of experiments to compare their performances. The back-projection was performed because it is quite similar to a (LIN; LEE, 2021). Since plane sweep stereo is utilized as a 3D pose estimator, the back-projection, which is also used by the plane sweep stereo, seems to be an interesting option for the matching process. As for ground point matching, it is used because of its simplicity and ability to provide a robust matching process. Instead of comparing entire poses, it now requires comparing only one single point (ground point) attached to each person. As mentioned before, to evaluate this work, it utilized two approaches, each one with a own loss process: MSE loss and smooth L_1 loss functions. By conducting these experiments and evaluations, it aimed to gain a deeper understanding of the strengths and weaknesses of the presented algorithms and identify areas where further improvements could be made.

By leveraging ground point matching and employing the smooth L_1 loss instead of the traditional mean squared error (MSE) loss, the ground point matching method has been demonstrated to produce superior outcomes in comparison to backprojection approach as evidenced by the data presented in Table 1. These findings suggest that the incorporation of ground point matching and the smooth L_1 loss can significantly enhance the accuracy and reliability of pose estimation systems. Furthermore, related to backprojection approach, it's noteworthy to state that the effective utilization of the Adabelief optimizer not only ensures rapid convergence but also facilitates strong generalization and stable training.

The study demonstrates that utilizing ground point matching with the Adam optimizer and Smooth L1 loss leads to significantly better performance in comparison to all backprojection methods, irrespective of the optimizer and loss function employed. These compelling results strongly support the notion that ground point matching offers a more effective approach for accurate 3D pose estimation. It's also worth highlighting that the demonstrated superiority of the smooth L1 loss underscores its enhanced suitability for tackling this particular challenge, likely attributed to its robustness in effectively handling outliers.

By maintaining the back-projection matching and solely modifying the loss function, this work were able to achieve considerable enhancements in the PCP values. However, it is important to note that back-projection is a multifaceted technique for matching between the target and reference views and may produce erroneous matches. Therefore, this work opted to adopt ground point matching instead. The outcome of this decision was that ground point matching, in conjunction with the Adam optimizer, outperformed back-projection matching both on average and across all actors.

Table 2 – Comparing ground point matching approach PCP on Campus Dataset with the state-of-the-art.

Method	Actor1	Actor2	Actor3	Average	Std
Belagiannis et al., 2014a	82.0	72.4	73.7	75.8	5.2
Belagiannis et al., 2014b	83.0	73.0	78.0	78.0	5
Belagiannis et al., 2015	93.5	75.7	84.4	84.5	8.9
Ershadi-Nasab et al., 2018	94.2	92.9	84.6	90.6	5.2
Dong et al., 2019	97.6	93.3	98.0	96.3	2.6
de França Silva et al., 2022	96.9	87.8	88.9	91.2	5.0
Ours	98.4	93.4	98.6	96.8	2.9
Huang et al., 2020	98.0	94.8	97.4	96.7	1.7
Tu et al., 2020	97.6	93.8	98.8	96.7	2.6
Lin and Lee, 2021	98.4	93.7	99.0	97.0	2.9

Source: The author (2023)

5.4 COMPARISON OF GROUND POINT MATCHING WITH STATE-OF-THE-ART

It is shown the table 2 comparing ground point matching approach with all other methods. The table is divided into two parts: the first part contains unsupervised/geometric methods, and the second part includes supervised methods. The proposed method outperform all the unsupervised methods with the approach using ground points in the matching process and smooth L_1 reprojection error loss. Furthermore, it outperforms on average and also in all the actors. The results from this master thesis are located at the bottom of the first part of the table since this approach, like the others, is unsupervised. However, the last three techniques are supervised. The division is based on two groups of techniques: unsupervised and supervised.

Furthermore, the table 2 illustrates the notable advancements achieved by the techniques over time. Initially, some methods, such as 3DPS, underwent refinements, leading to improvements in their performance. Subsequently, the incorporation of neural networks further revolutionized the field. Notably, significant enhancements were observed in unsupervised methods, particularly in crucial stages like the matching process, which played a pivotal role in refining the clustering of 2D poses. These improvements subsequently had a positive impact on the accuracy of 3DPS. The transition from 3DPS to neural networks marked a transformative turning point in 3D pose estimation. The adoption of supervised neural network techniques, employing 2D (LIN; LEE, 2021) or 3D (TU; WANG; ZENG, 2020) CNNs, proved to be a game-changer, resulting in substantial progress and propelling the state-of-the-art forward. The integration of neural networks brought numerous benefits, including improved generalization capabilities, better handling of complex spatial relationships, and the ability to learn from labeled datasets. Consequently, the performance of 3D pose estimation methods witnessed a significant boost, with state-of-the-art results. It is worth mentioning that the success of these neural network-based approaches is owed to both the availability of rich, annotated datasets and advancements in deep learning architectures. In summary, the results presented in Table 2 demonstrate the progressive evolution of 3D pose estimation techniques over time. The transition from traditional methods like 3DPS to the incorporation of neural networks has resulted in significant performance improvements. Moreover, supervised neural networks has propelled the state-of-the-art, showcasing the potential of deep learning approaches in pushing the boundaries of 3D pose estimation accuracy and applicability.

It is important to note that performance varies among actors, and several key factors influence these differences. One crucial aspect is occlusion, where certain parts of the actor's body may be hidden from view. The 3D pose estimation process relies on obtaining a 2D pose from the RGB images captured by the cameras. However, if a person is not fully visible to all cameras, the accuracy of the 3D pose estimation becomes more challenging. Additionally, the person's positioning can pose difficulties for estimating their 2D pose accurately. For instance, if a person is in an unusual or complex pose, it may be harder to determine their exact position from the images. Another challenging scenario occurs when people interact closely with each other. In such cases, estimating their poses becomes more challenging compared to situations where individuals are more distant from each other.

Compared to the supervised methods, it outperforms (HUANG et al., 2020) and (TU; WANG; ZENG, 2020) on average, being below (LIN; LEE, 2021) only. Moreover, considering that (HUANG et al., 2020) and (TU; WANG; ZENG, 2020) need 3D annotations, the proposed method has an impressive advantage. Enhanced results achieved through ground point matching suggest that this approach could prove more efficacious for 3D pose estimation in scenarios involving multiple views and individuals. As depicted in table 2, the standard deviation of PCP values among the three actors stands out as notably low when compared to alternative methods. Consequently, the ability to maintain robustness across different actors becomes a crucial attribute for a 3D pose estimation technique, ensuring its effective performance with diverse individuals in real-world scenarios. Additionally, it's worth noting that the practical effectiveness of unsupervised learning extends seamlessly to the realm of 3D pose estimation, thereby enlarging the horizons of its potential applications. This attribute becomes especially valuable when confronted with scenarios where acquiring labeled data presents challenges or constraints in terms of availability.

6 CONCLUSION

The present work aims to propose a novel solution to the problem of 3D pose estimation of multiple persons in a multi-view scenario, using an unsupervised approach with a simpler and more robust matching process. To achieve this, this leverages ground points to eliminate the need for 3D back-projections, allowing for a more straightforward and efficient matching process. Additionally, this method compare only one point per person, making the process more reliable and reducing the chances of errors.

Key discoveries from this study highlight the viability of leveraging reprojection error to obviate the need for 3D labeled data. Instead of resorting to intricate matching algorithms, a simplistic approach involving a reference point (a singular point of comparison as opposed to an entire 2D pose) yields noteworthy outcomes. The crux of achieving accurate results with the reprojection error lies in the precision of the matching process, underscoring the method's proficiency in making apt comparisons. When considering methods like backprojection, employing enhanced optimizers emerges as a promising avenue for refining the matching process. Notably, the choice of loss function exerts a pivotal influence on the results, with the implementation of the smooth L1 loss demonstrating its significance; yet, potential enhancements in the neural network architecture also hold promise for enhancing the precision of 3D estimations.

In contrast to preceding methodologies, this study holds a strong and commendable position, boasting exceptional outcomes. When juxtaposed with geometric approaches, this research distinguishes itself by harnessing a more robust technique grounded in deep learning principles, thereby surpassing the capabilities of traditional geometric methods. Furthermore, the matching process is notably simplified, as this work foregoes the intricacies associated with amalgamating epipolar geometry and person reidentification. In the realm of previous deep learning methods, this research exhibits a significant advantage by sidestepping the need for labeled data. This innovation is achieved by computing loss through reprojection error, enabling the generation of labels during the training process through the comparison of 2D poses, thereby eliminating the reliance on pre-annotated 3D data.

Additionally, it is noteworthy to consider the assessment of the proposed approach across a range of views, such as employing 2, 4, or 5 views. This investigation would provide insight into whether augmenting the number of views yields improved outcomes or if comparable or diminished results are obtained with fewer views. Moreover, the challenges associated with indoor environments must be acknowledged, as increased proximity and prolonged interaction among individuals could complicate matching. Thus, it is imperative to conduct the methodology across a diverse array of scenarios. Furthermore, it is imperative to extend the application of the presented approach to scenarios involving two or more actors, potentially encompassing five or even six individuals. This expansion is of paramount importance given that the typical environments targeted for monitoring or pedestrian tracking tend to be densely populated. Consequently, it becomes crucial to thoroughly assess the efficacy of the approach under such circumstances, gauging its performance in scenarios characterized by a significant volume of individuals.

Given that the neural network architecture mirrors that of (LIN; LEE, 2021), the method introduced here demonstrates comparatively inferior outcomes. This discrepancy arises due to the computation of loss using 2D poses instead of the 3D annotated poses employed in the reference work. The original neural network design was tailored for 3D pose comparison, thus prompting consideration for adjustments that render it more compatible with the evaluation of 2D poses. Such adaptations could potentially propel the proposed unsupervised technique to surpass the current state-of-the-art represented by (LIN; LEE, 2021).

Moreover, it employs a smooth L1 loss instead of comparing 2D poses with the Mean Squared Error (MSE). This modification has proved to be highly effective, and the results obtained with this approach demonstrate the enormous potential of using unsupervised methods instead of supervised ones based on 3D annotations.

The future work includes conducting experiments on more datasets to further validate our method's effectiveness and refine the loss using other regularizers such as Jensen-Shanon. By exploring the use of other regularization techniques, it hopes to further enhance the performance of the proposed method, making it even more effective for 3D pose estimation. Overall, this work provides a promising approach for pose matching that has the potential to impact several fields, including computer vision, robotics, and augmented reality.

Taking into account the foregoing insights, the prospect of attaining a 3D pose estimator without necessitating annotated 3D labels offers the tantalizing opportunity to develop a neural network exclusively reliant on captured images. This approach circumvents the resource-intensive task of manually annotating 3D poses. An intriguing proposition involves establishing a designated locale furnished with synchronized and calibrated cameras, enabling the training of the model using the captured images. Subsequently, the trained model can be harnessed for diverse applications, including but not limited to pedestrian tracking, movement intention analysis, and semantic interpretation. This trajectory envisions the progression of this master's thesis into a potent product, one brimming with potential to simplify the process of 3D pose estimation for multiple individuals.

REFERENCES

ALTEXSOFT. *Semi-Supervised Learning, Explained with Examples*. 2022. Accessed on March 10, 2023. Disponível em: <https://www.altexsoft.com/blog/semi-supervised-learning/>.

ALZUBAIDI, L.; ZHANG, J.; HUMAIDI, A. J.; AL-DUJAILI, A.; DUAN, Y.; AL-SHAMMA, O.; SANTAMARÍA, J.; FADHEL, M. A.; AL-AMIDIE, M.; FARHAN, L. Review of deep learning: Concepts, cnn architectures, challenges, applications, future directions. *Journal of big Data*, Springer, v. 8, p. 1–74, 2021.

AMIN, S.; ANDRILUKA, M.; ROHRBACH, M.; SCHIELE, B. Multi-view pictorial structures for 3d human pose estimation. In: BRISTOL, UK. *Bmvc*. [S.l.], 2013. v. 1, n. 2.

BELAGIANNIS, V.; AMIN, S.; ANDRILUKA, M.; SCHIELE, B.; NAVAB, N.; ILIC, S. 3d pictorial structures for multiple human pose estimation. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2014. p. 1669–1676.

BELAGIANNIS, V.; AMIN, S.; ANDRILUKA, M.; SCHIELE, B.; NAVAB, N.; ILIC, S. 3d pictorial structures revisited: Multiple human pose estimation. *IEEE transactions on pattern analysis and machine intelligence*, IEEE, v. 38, n. 10, p. 1929–1942, 2015.

BELAGIANNIS, V.; WANG, X.; SCHIELE, B.; FUA, P.; ILIC, S.; NAVAB, N. Multiple human pose estimation with temporally consistent 3d pictorial structures. In: SPRINGER. *Computer Vision-ECCV 2014 Workshops*. [S.l.], 2014. p. 742–754.

BERCLAZ, J.; FLEURET, F.; TURETKEN, E.; FUA, P. Multiple object tracking using k-shortest paths optimization. *IEEE transactions on pattern analysis and machine intelligence*, IEEE, v. 33, n. 9, p. 1806–1819, 2011.

BROWN, T.; MANN, B.; RYDER, N.; SUBBIAH, M.; KAPLAN, J. D.; DHARIWAL, P.; NEELAKANTAN, A.; SHYAM, P.; SASTRY, G.; ASKELL, A. et al. Language models are few-shot learners. *Advances in neural information processing systems*, v. 33, p. 1877–1901, 2020.

BRYNTE, L.; KAHL, F. Pose proposal critic: Robust pose refinement by learning reprojection errors. *arXiv preprint arXiv:2005.06262*, 2020.

BURENIUS, M.; SULLIVAN, J.; CARLSSON, S. 3d pictorial structures for multiple view articulated pose estimation. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2013. p. 3618–3625.

CIMEN, G.; MAURHOFER, C.; SUMNER, B.; GUAY, M. Ar poser: Automatically augmenting mobile pictures with digital avatars imitating poses. In: *12th international conference on computer graphics, visualization, computer vision and image processing*. [S.l.: s.n.], 2018.

CORMIER, M.; CLEPE, A.; SPECKER, A.; BEYERER, J. Where are we with human pose estimation in real-world surveillance? In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. [S.l.: s.n.], 2022. p. 591–601.

DAYAN, P.; SAHANI, M.; DEBACK, G. Unsupervised learning. *The MIT encyclopedia of the cognitive sciences*, MIT Press, p. 857–859, 1999.

DEVLIN, J.; CHANG, M.-W.; LEE, K.; TOUTANOVA, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

DONG, J.; JIANG, W.; HUANG, Q.; BAO, H.; ZHOU, X. Fast and robust multi-person 3d pose estimation from multiple views. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. [S.l.: s.n.], 2019. p. 7792–7801.

EINFALT, M.; LUDWIG, K.; LIENHART, R. Uplift and upsample: Efficient 3d human pose estimation with uplifting transformers. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. [S.l.: s.n.], 2023. p. 2903–2913.

ERSHADI-NASAB, S.; NOURY, E.; KASAEI, S.; SANAEI, E. Multiple human 3d pose estimation from multiview images. *Multimedia Tools and Applications*, Springer, v. 77, p. 15573–15601, 2018.

ERVEN, T. V.; HARREMOS, P. Rényi divergence and kullback-leibler divergence. *IEEE Transactions on Information Theory*, IEEE, v. 60, n. 7, p. 3797–3820, 2014.

FORSYTH, D. A.; PONCE, J. *Computer vision: a modern approach*. [S.l.]: prentice hall professional technical reference, 2002.

GHAHRAMANI, Z. Unsupervised learning. *Advanced Lectures on Machine Learning: ML Summer Schools 2003, Canberra, Australia, February 2-14, 2003, Tübingen, Germany, August 4-16, 2003, Revised Lectures*, Springer, p. 72–112, 2004.

GOODFELLOW, I.; POUGET-ABADIE, J.; MIRZA, M.; XU, B.; WARDE-FARLEY, D.; OZAIR, S.; COURVILLE, A.; BENGIO, Y. Generative adversarial networks. *Commun. ACM*, Association for Computing Machinery, New York, NY, USA, v. 63, n. 11, p. 139–144, oct 2020. ISSN 0001-0782. Disponível em: <https://doi.org/10.1145/3422622>.

GOOGLE. *Google ML - Descending into ML: Training and Loss*. 2022. Accessed on March 10, 2023. Disponível em: <https://developers.google.com/machine-learning/crash-course/descending-into-ml/training-and-loss>.

HADY, M. F. A.; SCHWENKER, F. Semi-supervised learning. *Handbook on Neural Information Processing*, Springer, p. 215–239, 2013.

HARTLEY, R.; ZISSERMAN, A. *Multiple view geometry in computer vision*. [S.l.]: Cambridge university press, 2003.

HOCHREITER, S.; SCHMIDHUBER, J. Long short-term memory. *Neural Comput.*, MIT Press, Cambridge, MA, USA, v. 9, n. 8, p. 1735–1780, nov 1997. ISSN 0899-7667. Disponível em: <https://doi.org/10.1162/neco.1997.9.8.1735>.

HUANG, C.; JIANG, S.; LI, Y.; ZHANG, Z.; TRAISH, J.; DENG, C.; FERGUSON, S.; XU, R. Y. D. End-to-end dynamic matching network for multi-view multi-person 3d pose estimation. In: SPRINGER. *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVIII 16*. [S.l.], 2020. p. 477–493.

HWANG, J.; PARK, S.; KWAK, N. Athlete pose estimation by a global-local network. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. [S.l.: s.n.], 2017. p. 58–65.

IONESCU, C.; PAPAVA, D.; OLARU, V.; SMINCHISESCU, C. Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE transactions on pattern analysis and machine intelligence*, IEEE, v. 36, n. 7, p. 1325–1339, 2013.

JEFFARES, A. *Supervised vs Unsupervised Learning in 3 Minutes*. 2018. Accessed on March 10, 2023. Disponível em: <https://towardsdatascience.com/supervised-vs-unsupervised-learning-in-2-minutes-72dad148f242>.

JIA, J.; LIU, J.; JIN, G.; WANG, Y. Fast and effective occlusion culling for 3d holographic displays by inverse orthographic projection with low angular sampling. *Applied Optics*, v. 53, 09 2014.

JOHNSON, S.; EVERINGHAM, M. Clustered pose and nonlinear appearance models for human pose estimation. In: *Proceedings of the British Machine Vision Conference*. [S.l.]: BMVA Press, 2010. p. 12.1–12.11. ISBN 1-901725-40-5. Doi:10.5244/C.24.12.

KIM, W.; RAMANAGOPAL, M. S.; BARTO, C.; YU, M.-Y.; ROSAEN, K.; GOUMAS, N.; VASUDEVAN, R.; JOHNSON-ROBERSON, M. Pedx: Benchmark dataset for metric 3-d pose estimation of pedestrians in complex urban intersections. *IEEE Robotics and Automation Letters*, IEEE, v. 4, n. 2, p. 1940–1947, 2019.

KINGMA, D. P.; BA, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

KITANI, K. *Camera Matrix*. [S.l.]: Carnegie Mello - THE ROBOTICS INSTITUTE, 2017.

KUHN, H. W. The hungarian method for the assignment problem. *Naval research logistics quarterly*, Wiley Online Library, v. 2, n. 1-2, p. 83–97, 1955.

LI, Z.; YE, J.; SONG, M.; HUANG, Y.; PAN, Z. Online knowledge distillation for efficient pose estimation. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. [S.l.: s.n.], 2021. p. 11740–11750.

LIMA, J. P.; ROBERTO, R.; FIGUEIREDO, L.; SIMOES, F.; TEICHRIEB, V. Generalizable multi-camera 3d pedestrian detection. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. [S.l.: s.n.], 2021. p. 1232–1240.

LIN, J.; LEE, G. H. Multi-view multi-person 3d pose estimation with plane sweep stereo. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. [S.l.: s.n.], 2021. p. 11886–11895.

LIU, B.; LIU, B. *Supervised learning*. [S.l.]: Springer, 2011.

MARINOIU, E.; ZANFIR, M.; OLARU, V.; SMINCHISESCU, C. 3d human sensing, action and emotion recognition in robot assisted therapy of children with autism. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2018. p. 2158–2167.

MITCHELL, T. M. et al. *Machine learning*. [S.l.]: McGraw-hill New York, 2007. v. 1.

MORDVINTSEV, A.; OLAH, C.; TYKA, M. Deepdream-a code example for visualizing neural networks. *Google Research*, v. 2, n. 5, 2015.

NIBALI, A.; HE, Z.; MORGAN, S.; PRENDERGAST, L. Numerical coordinate regression with convolutional neural networks. *arXiv preprint arXiv:1801.07372*, 2018.

O'SHEA, K.; NASH, R. An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*, 2015.

RAGUPATHI, G. R. *"Machine Learning" From Top Angle*. 2022. Accessed on March 10, 2023. Disponível em: <https://blogs.sap.com/2022/10/05/machine-learning-from-top-angle-view/>.

SHERSTINSKY, A. Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network. *Physica D: Nonlinear Phenomena*, Elsevier, v. 404, p. 132306, 2020.

SIGAL, L.; ISARD, M.; HAUSSECKER, H.; BLACK, M. J. Loose-limbed people: Estimating 3d human pose and motion using non-parametric belief propagation. *International journal of computer vision*, Springer, v. 98, p. 15–48, 2012.

SOCIETY, S. C. *Simplifying the Difference: Machine Learning vs Deep Learning*. 2021. Accessed on March 10, 2023. Disponível em: <https://www.scs.org.sg/articles/machine-learning-vs-deep-learning>.

SUN, K.; XIAO, B.; LIU, D.; WANG, J. Deep high-resolution representation learning for human pose estimation. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. [S.l.: s.n.], 2019. p. 5693–5703.

TU, H.; MENZIES, T. Debtfree: minimizing labeling cost in self-admitted technical debt identification using semi-supervised learning. *Empirical Software Engineering*, Springer, v. 27, n. 4, p. 80, 2022.

TU, H.; WANG, C.; ZENG, W. Voxelpose: Towards multi-camera 3d human pose estimation in wild environment. In: SPRINGER. *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*. [S.l.], 2020. p. 197–212.

VARSHINI, C.; HRUDAY, G.; CHANDU, G.; SHARIF, S. Sign language recognition. *International Journal of Engineering Research and*, V9, 06 2020.

VASWANI, A.; SHAZEER, N.; PARMAR, N.; USZKOREIT, J.; JONES, L.; GOMEZ, A. N.; KAISER, Ł.; POLOSUKHIN, I. Attention is all you need. *Advances in neural information processing systems*, v. 30, 2017.

WANG, J.; TAN, S.; ZHEN, X.; XU, S.; ZHENG, F.; HE, Z.; SHAO, L. Deep 3d human pose estimation: A review. *Computer Vision and Image Understanding*, Elsevier, v. 210, p. 103225, 2021.

XIU, Y.; LI, J.; WANG, H.; FANG, Y.; LU, C. Pose Flow: Efficient online pose tracking. In: *BMVC*. [S.l.: s.n.], 2018.

ZHENG, C.; ZHU, S.; MENDIETA, M.; YANG, T.; CHEN, C.; DING, Z. 3d human pose estimation with spatial and temporal transformers. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. [S.l.: s.n.], 2021. p. 11656–11665.

ZHU, X. J. Semi-supervised learning literature survey. University of Wisconsin-Madison Department of Computer Sciences, 2005.

ZHUANG, J.; TANG, T.; DING, Y.; TATIKONDA, S. C.; DVORNEK, N.; PAPADEMETRIS, X.; DUNCAN, J. Adabelief optimizer: Adapting stepsizes by the belief in observed gradients. *Advances in neural information processing systems*, v. 33, p. 18795–18806, 2020.