ETH zürich
pd|z Product Development Group Zurich
Produktentwicklungsgruppe Zürich

ETH Zurich - Product Development Group Zurich pd|z

Thesis Number: 1799
Supervisor(s): Hogenkamp Maarten
Professor: Prof. Mirko Meboldt
Start Date: 10.03.2025
Submission Date: 13.06.2025

SEMESTER THESIS

Surgeon Hand Pose Tracking for a Digital Surgery Platform

Nilesh Balu[1]

[1]ETH Zurich

*Corresponding author. E-mail: nibalu@student.ethz.ch

Abstract

Digital surgical platforms are gaining popularity due to their wide range of medical applications, including surgeon training, automatic performance evaluation, and remote surgical procedures. One approach to building such a platform involves creating a digital twin of the surgical environment. This thesis presents a pipeline for detecting and estimating the hand pose of a surgeon in the operating theatre. The system first deploys a hand pose estimation model (WiLoR) to detect 2D and 2.5D hand keypoints. These keypoints are then triangulated into 3D world coordinates using an optimisation-based method that ensures anatomically plausible reconstructions. Kalman filtering and outlier detection are applied to reduce temporal noise and improve robustness. Validation is conducted using marker-based motion capture techniques.
The pipeline is implemented on a compact embedded platform, the ZED Box powered by NVIDIA Jetson Orin, enabling real-time inference at 8 frames per second, sufficient for many observational and interactive tasks. Several validation strategies are used, including ArUco markers, infrared tracking, and gloves fitted with circular blobs. The blob-based method provides the most reliable results, with root mean squared error (RMSE) values in the order of 10 mm.
While the results are promising, the validation setup is limited to a controlled environment and does not fully reflect the complexity of real surgical scenes, which involve frequent occlusions and intricate hand gestures. In addition, the frame rate remains a limiting factor for high-speed surgical applications. Future developments include training the model on a domain-specific dataset, integrating tool pose constraints to improve hand pose predictions, and establishing a more robust validation framework using multi-camera motion capture or simulation environments. These enhancements are intended to improve the system's robustness, accuracy, and suitability for deployment in realistic surgical scenarios where precise, real-time hand tracking is essential.

Keywords: Digital Twin; Hand Pose Estimation; Stereo Camera; Triangulation; Motion Capture

1. Introduction

A global shortage of surgeons was predicted over a decade ago [1], and the problem has only intensified due to population growth and the rising proportion of elderly individuals. The increasing demand for surgical care underscores the shortage of skilled professionals. One way to address this is by making surgical training more accessible, scalable, and cost-effective [2]. However, this approach faces a key challenge: experienced surgeons, who traditionally serve as mentors, are increasingly burdened with clinical duties, limiting their availability for hands-on teaching. This highlights the need for alternative training methods that reduce reliance on direct supervision while still ensuring quality training.

A promising direction is the development of digital surgery platforms that can objectively assess surgical performance and offer automated, data-driven feedback. Integrating such platforms into training curricula could reduce dependency on expert instruc-

Declaration of Originality:

I hereby declare that I have written the present thesis independently and have not used other sources and aids than those stated in the bibliography.

Name of Student

tors, increasing both the efficiency and consistency of surgical education.

A particularly compelling concept in this space is the digital twin of the operating room—a virtual model that mirrors the physical environment in real time. For this model to be effective, it must accurately track all relevant agents involved in a surgical procedure, including instruments, the patient, and the surgeon. While tool pose and patient anatomy can already be tracked accurately, hand pose estimation for the surgeon remains a challenging area of research.

This thesis focuses on enhancing these digital twins by enabling robust tracking of the surgeon's hand pose. A comprehensive review of surgical skill evaluation methods [3] highlights the central role of motion and kinematic analysis—including hand trajectories—in measuring technical ability. Accurate hand pose estimation is essential for assessing dexterity, gesture precision, and procedural compliance.

To address this challenge, a pre-trained neural network is used to estimate hand pose from stereo image data. After extracting 2D keypoints from both cameras, corresponding points are triangulated to obtain their 3D positions. A filtering step is applied to smooth the trajectory and eliminate outliers caused by noise. Additionally, a validation pipeline is introduced, in which marker-based motion capture techniques provide ground truth for selected hand keypoints. The estimated 3D positions are then compared against these ground truth measurements to quantitatively evaluate the system's accuracy and reliability.

Section 2 reviews relevant literature on human body pose estimation. Section 3 describes the structure and implementation of the proposed pipeline. The results are presented in Section 4, followed by a discussion in Section 5.

2. Related Work

With the growing recognition of the importance of biomechanical analysis in both sports science and product development, there has been a steady increase in research focused on human body pose estimation. These models typically process either a single RGB image or a temporal sequence of RGB frames to estimate an ordered set of keypoints corresponding to the major joints of the human body. Depending on the field, models are trained to either estimate the whole body pose or the pose of specific parts, like the fingers in the hand.

2.1 Body Pose Estimation

Models in this category focus on estimating the major limbs of the body. A common approach is to learn the mapping between the features extracted from an image and a heatmap that represents the probability distribution of joint locations. One notable model, *Cascaded Pyramid Network (CPN)* [4], does this by cascading two feature extraction networks, one dedicated to capturing simple features and another focused on hard-to-detect features. Another state-of-the-art model, *Pose as Compositional Tokens (PCT)* [5], employs an encoder-decoder architecture to learn a codebook of compositional tokens, which are subsequently classified into different joint types. Both these models are designed for 2D pose estimation.

To infer 3D poses from 2D inputs, several advanced architectures have been developed. *TCPFormer* [6], for instance, uses transformer networks with an intermediate layer called an implicit pose proxy, which facilitates learning spatio-temporal correlation. Similarly, *WHAM* [7] uses an encoder-decoder architecture to estimate 3D pose from videos. This model imposes ground contact constraints that lead to more stable estimations.

Despite these advancements, a notable limitation of these models is the low resolution of the pose that they estimate. Most of these models are trained on datasets like *COCO* [8] or *3DPW* [9], which do not contain poses of the individual fingers. This represents a significant shortcoming, particularly in applications such as the creation of a digital twin of a surgeon, which requires precise and stable estimation of the entire hand, including all individual fingers.

However, most body pose estimation models focus primarily on the torso and major limbs, often omitting finer articulations such as those in the hands and fingers. This limitation becomes critical in applications that require detailed interaction modelling, such as surgical simulation or gesture analysis. To address this, a separate line of research has emerged focusing specifically on the estimation of hand poses.

2.2 Hand Pose Estimation

Certain models are specifically designed to estimate hand poses, with approaches generally falling into two categories based on input modality: RGB images and depth images. Across both categories, model performance is typically evaluated using the mean per-joint position error (MPJPE). As reported by [10], models achieving an MPJPE of less than 10 mm are considered highly accurate, a threshold that is widely recognised as a benchmark for high-precision pose estimation.

2.2.1 RGB Images

Models in this category take an RGB image as input and output keypoints corresponding to the 3D pose of the hand. *Hamba* [11], for instance, employs a Graph Guided State Space block to learn the spatial features of the hand joints. *WiLoR* [12] utilises a pre-trained ViT model as a backbone and incorporates a multi-scale pose refinement module that predicts pose and shape residuals. Both these models effectively fit MANO [13], a parametric hand model, onto the input image. However, the estimated 3D pose is relative to the wrist, meaning the hand pose is not placed in the 3D world coordinate frame. This representation is sometimes referred to as 2.5D. Some datasets used in training these models are *FreiHAND* [10] and *WHIM* [12].

2.2.2 Depth Images

Another category of models uses a single depth image to estimate the 3D hand pose. These models are capable of estimating joint coordinates in a global 3D frame, because the depth image directly provides the 3D coordinates of each pixel. *AWR* [14], for instance, combines regression-based methods and detection-based methods to achieve accurate hand pose estimation. It improves robustness against occlusions by employing a weight distribution mechanism that spreads information across adjacent joints. Similarly, *Efficient Virtual View Selection* [15] enhances robustness by generating multiple virtual viewpoints and fusing the results to produce a final, consistent hand pose estimation. These models are commonly trained on the *NYU* [16] and *ICVL* [17] datasets.

While dedicated hand pose estimation models have made significant progress in capturing fine-grained finger movements, they typically estimate hand pose in isolation, without considering its relationship to the rest of the body. This separation limits their applicability in scenarios that require holistic human modelling. To overcome this, recent approaches aim to estimate the full-body pose, including both body and hands, within a unified framework.

2.3 Whole-Body Pose Estimation

Estimating the pose of the body, including the fingers, arms, and torso, provides richer contextual information that can be beneficial for various downstream tasks. Consequently, it is logical to explore whole-body pose estimation models that simultaneously estimate both hand and body poses. *DWPose* [13], for example, takes an RGB image as input and predicts 2D keypoints of the full body. To extend this to 3D, *Semantic GANs* [18] convert the 2D keypoints into a 3D representation using a semantic graph attention encoder. This encoder leverages both local and global features to enhance the accuracy of the whole-body pose estimation.

However, this approach only estimates the 2.5D pose. Very little has been explored with regard to using depth images for whole-body pose estimation. This is probably due to the lack of extractable features present in the depth image, making it difficult to reliably capture the entire body, particularly in complex poses or under occlusion.

2.4 Domain Specific Datasets

Some groups have released datasets [19, 20] tailored to surgical applications. However, a significant limitation of these datasets is that they typically adopt an egocentric (first-person) perspective. For applications that require third-person viewpoints, such as tracking the hand pose in a surgical environment, this poses a significant challenge. Currently, there is no publicly available third-person dataset tailored to surgical scenarios that would be suitable for fine-tuning whole-body pose estimation models.

3. Methods

The state-of-the-art models discussed in the previous section were evaluated on a custom dataset to identify the most suitable model for integration into the pipeline. The *WiLoR* model demonstrated superior performance due to its end-to-end implementation, which combines hand detection with both 2D and 2.5D keypoint estimation. This results in a two-stage pipeline, as seen in Figure 1: first, the hand must be detected within the frame, and the joint pixels must be estimated; second, the estimated hand skeleton must be transformed into 3D world coordinates.

3.1 Hand Pose Estimation

WiLoR estimates both 2D and 2.5D keypoints using the MANO hand model, which defines 21 keypoints—four per finger and one for the wrist. The 2D and 3D coordinates are represented in an ordered array of 21 elements. However, the 3D coordinates are reported relative to the wrist position, thereby requiring triangulation to obtain the absolute 3D pose.

The model was deployed on a custom dataset, recorded on the test bench shown in Figure 2a, to evaluate its robustness to occlusions. The video input included frames, like the one illustrated in Figure 2b, in which more than 50% of the hand was occluded, particularly during the use of orthopaedic surgical tools such as drills and screwdrivers.

3.2 Triangulation

To obtain the coordinates of keypoints in the world frame, the 2D keypoints from each image must be triangulated. Without loss of generality, the origin of the world coordinate frame is assumed to coincide with the origin of the image plane of the left camera. Several triangulation algorithms exist, out of which the Direct Linear Transforms (DLT) method is explored in detail.

3.2.1 Direct Linear Transforms

Direct linear transforms are used for simple stereo setups, as seen in Figure 3, where:

- The cameras are identical with no skew
- The image planes are co-planar with aligned x axes, separated by a baseline distance b

Further, we assume that the world coordinate frame is aligned with that of the left camera. As described in [21], this formulation employs the linear camera model given by

$$\tau p = K R^T (P - C) \tag{1}$$

where

- K is the internal camera parameter matrix
- R and C are the external camera parameter matrices
- P is the 3D position of the point
- p is the corresponding 2D position of the point on the image plane
- τ is a scaling factor

This equation can be written for both cameras in the stereo camera setup, and leads to a system of linear equations with three independent equations and three unknowns (the world coordinates). Thus, this system has a unique solution, given by

$$X = b \cdot \frac{x}{(x - x')} \tag{2}$$

$$Y = b \cdot \frac{k_x y}{k_y (x - x')} \tag{3}$$

$$Z = b \cdot \frac{k_x f}{(x - x')} \tag{4}$$

In this way, direct linear transforms can be used to arrive at a straightforward yet powerful closed-form solution. However, this method has two main disadvantages.

- First, the formula is imprecise for far-away objects
- Second, this closed-form solution is very sensitive to noise in the upstream task, viz., hand keypoint detection

The first issue is a common limitation of stereo vision systems and arises due to the resolution limits of the camera. However, for this application, the relevant keypoints are typically within one meter of the camera, so this is not a significant concern. The second issue, in contrast, is more problematic. Since the hand pose is inferred independently in both frames, there is considerable noise between the two estimates, which significantly affects the triangulation accuracy.

More importantly, this approach to triangulation does not fully exploit the WiLoR model. Using the 2.5D pose estimates of the WiLoR model would significantly improve the accuracy and biomechanical plausibility of the results

3.2.2 Refinement of DLT results

This section formulates triangulation as an optimisation problem. The objective is to find the optimal transformation (p^*) that transforms the MANO hand pose (*MANO*) from the wrist coordinates to the world coordinates, such that the Euclidean distance of the corresponding points between the estimated hand pose (X) in and the pose estimated using DLT (*DLT*) is minimised. The transformation comprises of rotation, translation and scaling. The

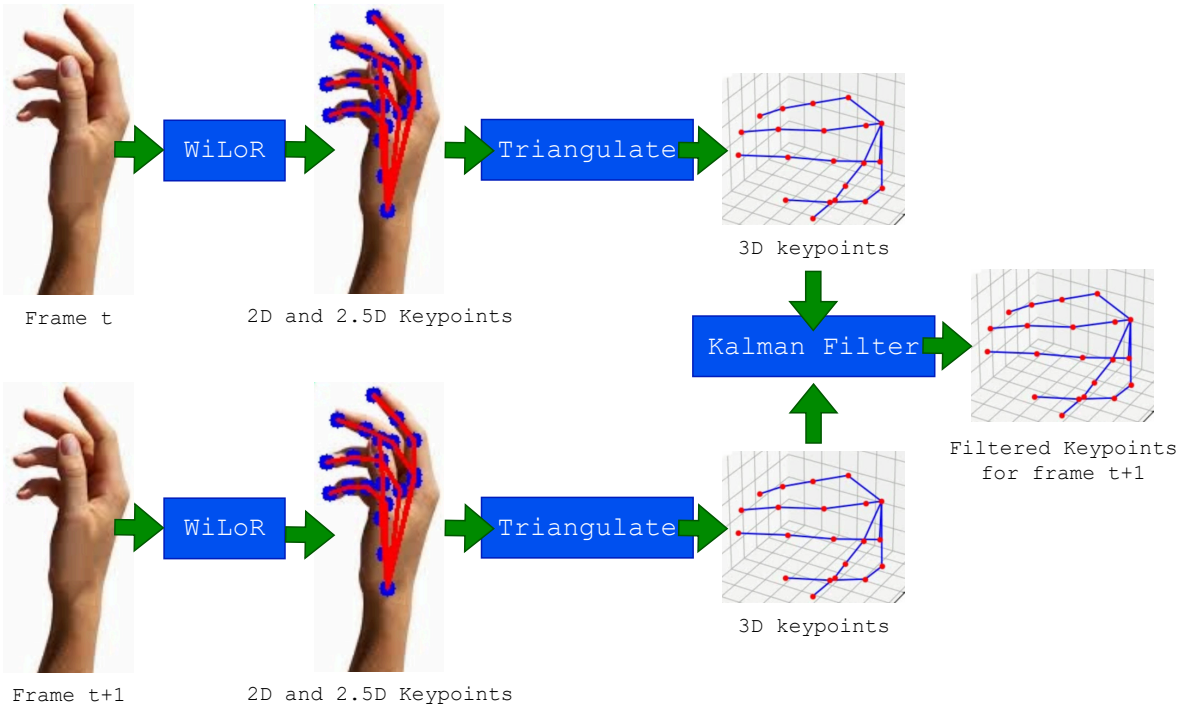


Figure 1: Structure of the pipeline: First, WiLoR estimates the 2D and 2.5D hand poses in the left and right frames at each timestep. These keypoints are then triangulated to obtain the 3D poses. The final result is smoothed across frames using a Kalman filter.

(a) The test bench **(b)** Frame from the custom dataset

Figure 2: The custom testing setup used to create data.

problem can be formulated as follows:

$$\underset{p}{\text{minimize}} \quad ||X(p) - DLT||_2 \tag{5}$$

where p is a vector containing the parameters of the transformation.

$$p = (\theta_x, \theta_y, \theta_z, t_x, t_y, t_z, s) \tag{6}$$

The transformation is performed as follows

$$X(p) = Translate \cdot R_z \cdot R_y \cdot R_x \cdot Scale \cdot MANO \tag{7}$$

where R_x, R_y and R_z are the standard rotation matrices, t_x, t_y, t_z are elements of the translation vector and s is the scaling coefficient.

To ensure the estimates do not blow up due to noise in the keypoint detections, the scaling factor (s) can be bounded to lie in

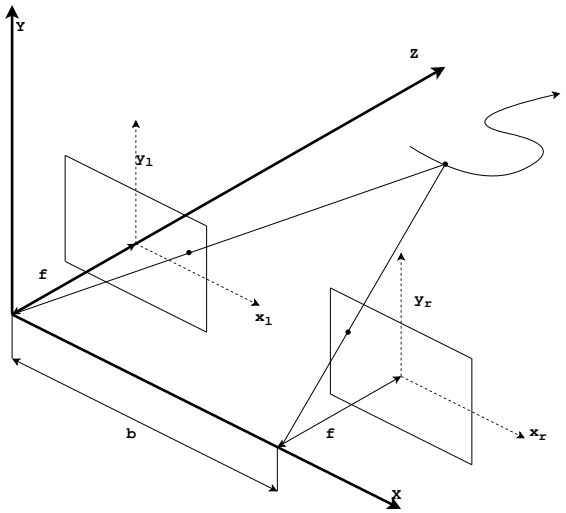


Figure 3: The stereo camera setup used in the ZEDX camera

$[0.9, 1.1]$.

$$0.9 \leq s \leq 1.1 \tag{8}$$

The Kabsch Umeyama algorithm [22] provides a closed-form solution to this unconstrained optimisation problem.

This pipeline is prone to noise. They are of two forms. First, the noise that arises because inference is carried out on a per-frame

basis. As a result, the pose estimate between frames might not be temporally consistent. This type of noise is hence referred to as temporal noise. The type of noise arises from the subtle spatial differences between the left and right input frames. This noise is referred to as spatial noise.

3.3 Kalman Filtering

Kalman Filter for Temporal Smoothing

The temporal noise in the pipeline can be mitigated by smoothing the estimates across frames using a Kalman Filter. This is a recursive Bayesian estimator for linear dynamical systems with Gaussian noise. In our case, the Kalman Filter estimates the underlying 3D position and velocity of each keypoint of the hand over time by combining noisy observations with a predictive motion model.

The hidden state vector x_t for each keypoint includes both the 3D position and velocity:

$$x_t = \begin{bmatrix} x & y & z & \dot{x} & \dot{y} & \dot{z} \end{bmatrix}^\top$$

and the observation vector y_t consists of the directly measured 3D position:

$$y_t = \begin{bmatrix} x & y & z \end{bmatrix}^\top$$

The state evolves according to the linear transition model

$$x_t = Ax_{t-1} + w_{t-1}$$

and the observation model is

$$y_t = Hx_t + v_t$$

where $w_t \sim \mathcal{N}(0, Q)$ is the process noise and $v_t \sim \mathcal{N}(0, R)$ is the observation noise, with covariance matrices Q and R respectively.

The transition and observation matrices are given by:

$$A = \begin{bmatrix} 1 & 0 & 0 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta t & 0 \\ 0 & 0 & 1 & 0 & 0 & \Delta t \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

The Kalman Filter operates in two alternating steps—prediction and update, which refine the estimate at each time step. Details can be found in the original paper by Kalman [23].

3.4 Outlier Detection

Triangulation is highly sensitive to spatial noise, which can introduce significant errors in the estimated positions. Therefore, it is essential to detect and remove outliers caused by such noise to prevent inaccurate estimations. High levels of noise often manifest as sudden jerks or abrupt changes in the triangulated positions across consecutive frames; these discontinuities can be used as a basis for identifying and rejecting outliers. Additionally, since the surgeon is expected to operate within a defined spatial region, any estimated position falling outside these operational bounds can also be classified as an outlier.

(a) ArUco marker setup **(b)** Blob markers

(c) IR marker glove **(d)** IR marker setup

Figure 4: Validation setup

3.5 Validation

To validate the accuracy of the pipeline, three different methods were used: an ArUco-based 3D triangulation setup, a blob-marker-based motion capture setup and an IR marker-based motion capture setup.

3.5.1 ArUco Marker Detection

For a preliminary validation of the triangulation stage, ArUco markers of known size were placed within the scene. The setup is shown in Figure 4a. The 3D coordinates of the marker corners were computed using both triangulation and the perspective projection method. The resulting positions were then compared.

3.5.2 The Atracsys System

To establish a baseline for validation, additional comparisons were performed using an infrared (IR) marker-based glove, as seen in Figure 4c, and a commercial optical tracking system, *Atracsys*. The *ZEDX* camera was placed above the *Atracsys* camera, as shown in Figure 4d, to simultaneously get the WiLoR estimates. The Atracsys system uses the minimum distance metric to match points before triangulation. Due to the large number of keypoints, however, using this metric is tricky. This provided the motivation to use a custom method to potentially find a more efficient way of matching keypoints.

3.5.3 Blob Marker Detection

Blob markers, as shown in Figure 4b, were used to capture the 2D keypoints of the hand pose. This was done by blob detection,

which estimated the 2D position of the centers of the blobs, along with the size of the blobs.

Figure 5: 2D keypoints from WiLoR

One important step before being able to triangulate these points is to find the correct matches. This can be formulated as an assignment problem, wherein the points from each set must be matched while satisfying certain constraints. Let:

$$p_i^L = (x_i^L, y_i^L) \quad \text{be the 2D keypoints in the left image}$$
$$p_j^R = (x_j^R, y_j^R) \quad \text{be the 2D keypoints in the right image}$$

Since this is a stereo setting, the points must lie on approximately the same epipolar line. Additionally, the disparity between the matching points must be roughly constant. These constraints can be formulated as follows:

$$|y_i^L - y_j^R| < \varepsilon_y \tag{9}$$
$$|d_{ij} - \bar{d}| < \varepsilon_d \quad \text{where } d_{ij} = x_i^L - x_j^R \tag{10}$$

Define a cost matrix $C \in \mathbb{R}^{N \times N}$ where each entry C_{ij} measures the cost of matching p_i^L to p_j^R. Let:

$$C_{ij} = \begin{cases} \infty & \text{if } |y_i^L - y_j^R| \geq \varepsilon_y \\ \alpha \cdot |y_i^L - y_j^R| + \beta \cdot (x_i^L - x_j^R - \bar{d})^2 & \text{otherwise} \end{cases}$$

We define an assignment matrix $X \in \{0,1\}^{N \times N}$ such that:

$$X_{ij} = \begin{cases} 1 & \text{if } i \text{ in the left image is assigned to } j \text{ in the right image} \\ 0 & \text{otherwise} \end{cases}$$

The assignment problem then becomes:

$$\min_X \quad \sum_{i=1}^{N} \sum_{j=1}^{N} C_{ij} \cdot X_{ij} \tag{11}$$

subject to (12)

$$\sum_{j=1}^{N} X_{ij} = 1 \quad \forall i \quad \text{(each left point assigned to one right point)} \tag{13}$$

$$\sum_{i=1}^{N} X_{ij} = 1 \quad \forall j \quad \text{(each right point assigned to one left point)} \tag{14}$$

$$X_{ij} \in \{0,1\} \quad \forall i,j \tag{15}$$

The Hungarian algorithm can be used to solve this problem. These matched keypoints were then triangulated to get the 3D coordinates and were compared to the WiLoR estimates.

4. Results

The pipeline was implemented on a *ZED Box*, which is an *NVIDIA Jetson Orin* fitted with a specialised Gigabit Multimedia Serial Link (GMSL) port to connect a *ZEDX* camera. This configuration allows for high-bandwidth, low-latency image capture, making it suitable for real-time computer vision applications such as hand pose estimation in constrained environments like surgical setups.

4.1 The WiLoR Model

The WiLoR model was deployed on the *ZED Box* to perform real-time hand pose inference. It achieved a processing speed of 8 frames per second, which is sufficient for many interactive and observational tasks. One of the key strengths of the model was its robustness to partial occlusions, an essential requirement in surgical scenes where instruments often obstruct parts of the hand. Figure 5 illustrates several example frames after inference, showing the overlay of 2D keypoints on the input image and the corresponding 2.5D keypoints projected separately. These keypoints served as the input for the subsequent triangulation stage.

4.2 Triangulation

4.2.1 DLT

The first triangulation approach explored was the classical Direct Linear Transform (DLT) method. As shown in Figure 6, the DLT algorithm produced plausible 3D reconstructions in many cases. However, it occasionally resulted in unnatural or anatomically incorrect hand poses. This limitation arises from the fact that each keypoint is triangulated independently, without considering spatial or physiological constraints that define valid hand configurations. As a result, the overall pose may not be coherent or realistic, especially under noisy or occluded conditions. These observations underline the limitations of purely geometric methods and motivate the incorporation of model-based or optimisation-driven approaches.

4.2.2 Refinement of DLT Results

This section illustrates the results obtained using the optimisation-based method. Figure 7 shows the resulting hand poses for selected frames, which appear anatomically consistent

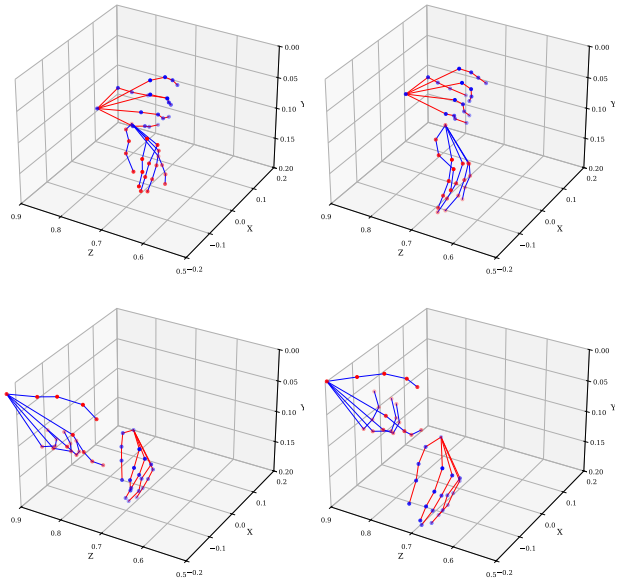


Figure 6: Results of DLT triangulation

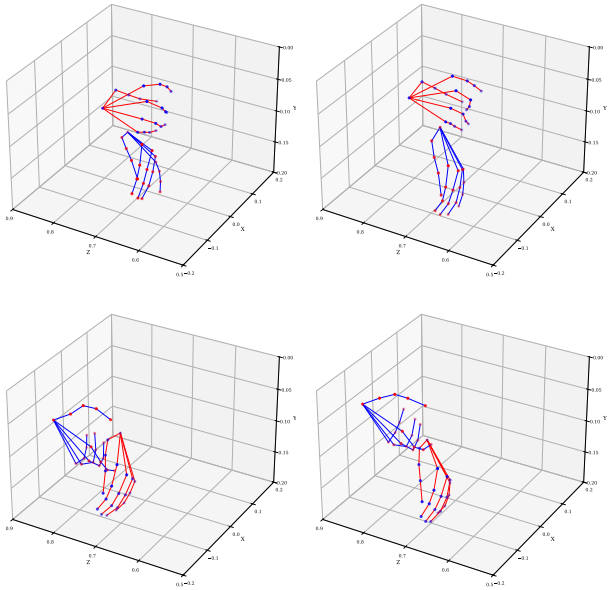


Figure 7: Results of the Kabsch Umeyama algorithm

and more natural than those obtained via DLT. The incorporation of a prior model ensures that the reconstructed poses remain within a valid range of human hand motion, even when some keypoints are ambiguous or missing due to occlusion.

4.3 Validation

4.3.1 ArUco Markers

To evaluate the accuracy of the 3D triangulation, the reconstructed positions of ArUco marker corners were compared with the positions derived using the perspective projection method. Figure 8 shows that the two estimates align closely, indicating that the triangulation method produces geometrically consistent results. The RMSE between the two methods was computed to be 4 mm.

4.3.2 Glove with Infra Red markers

Another validation scheme used was to wear a glove fitted with IR markers. The Atracsys system was then used to directly obtain the 3D world coordinates of the hand. Figure 9 compares this to the WiLoR estimates. The Atracsys estimates do not match the WiLoR estimates. In fact, these estimates do not approximately lie on the same plane. One reason explaining these poor results could be the simplicity of the matching algorithm used by the Atracsys system to match keypoints. The nearest distance method of matching might work for 2-3 keypoints in a scene, but 21 keypoints per frame is a lot.

4.3.3 Glove with Circular Blobs

To assess the hand pose detection stage of the pipeline, the predicted keypoints from WiLoR were compared with the positions of circular markers on a glove. Figure 10a compares the 2D estimates obtained using the two methods. The two estimates align closely, and the RMSE was computed to be 12 pixels (which translates to roughly 6 mm for the data used). Figure 10b compares the 3D estimates obtained using the two methods. The RMSE was computed to be 7 mm. As discussed in Section 2.2, the model based on this error is considered to be highly accurate.

This method for validation yielded more accurate results, mostly due to the custom method of matching points before triangulation. There is huge potential to build on this method by using different coloured blobs to differentiate between the different fingers of the hand. This could also yield more information about the occlusions encountered. The matching will then be performed for each set of colours.

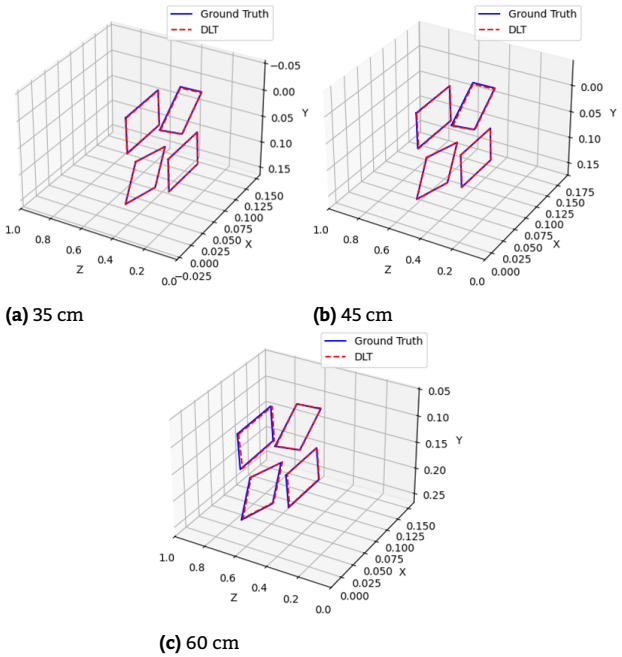
5. Conclusion and Discussion

This work presents a real-time hand pose estimation pipeline deployed on an embedded platform — the *ZED Box* powered by *Jetson Orin*. The system employs the WiLoR model to achieve robust 2.5D hand keypoint inference, even in the presence of occlusions caused by surgical tools. Triangulation methods were explored to reconstruct 3D hand poses, including a comparison between the traditional DLT approach and an optimisation-based method utilising the Kabsch-Umeyama algorithm.

The findings indicate that although DLT offers simplicity and efficiency, it lacks anatomical constraints and may yield implausible hand poses. In contrast, the optimisation-based method, supported by the MANO model, generates more natural and anatomically consistent reconstructions. Validation experiments using ArUco markers and a glove with circular blobs confirmed the accuracy of the triangulated outputs, demonstrating good alignment between projected estimates and actual positions.

The validation experiments, however, were performed in a very simple setting. The actual use case of the pipeline would involve much more complex gestures with many occlusions caused by surgical tools. These validation results, therefore, do not assess the performance of the pipeline in an actual surgery setting. The main shortcoming of marker based methods is that occlusions make it near impossible to detect the markers. Further, the hungarian algorithm cannot be solved with only partial detection of the keypoints.

Another issue is the low frame rate of 8 frames per second achieved. This rate can either be increased by using a more powerful GPU or by decreasing the complexity of the model. Another



(a) 35 cm **(b)** 45 cm

(c) 60 cm

Figure 8: Validating triangulation using ArUco markers at different distances

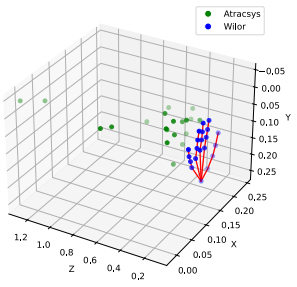


Figure 9: Results of validation using Atracsys

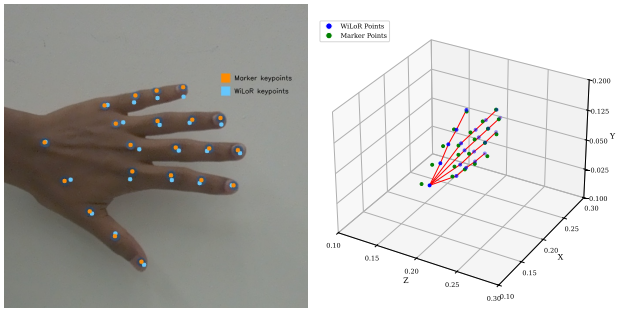
way to target this issue could be by using a Kalman filter to predict the frames in between WiLoR estimates.

6. Future Work

Now that a pipeline is set up, the next step would be to improve the performance of the model by training it on a custom dataset. This allows the model to better adapt to the specific domain, appearance, or conditions of the application setting, potentially improving accuracy and robustness compared to using only pre-trained weights. In particular, a custom dataset can help the model generalise better to uncommon hand shapes, poses, or object interactions seen in the target environment. For this, an efficient method for annotating images is essential.

Another way to improve the estimates and possibly reduce the number of outliers is to use the tool pose estimates to constrain the hand pose predictions. These constraints could either be integrated into the training process or imposed during inference.

Another scope for improvement is making a more robust validation pipeline. One way to improve results would then be to use an advanced motion capture setup with many cameras, estimating the keypoints from many viewpoints. Even in this setup, matching points for triangulation will be tricky. Other methods



(a) 2D comparison **(b)** 3D comparison

Figure 10: Validation using blob markers

would be to either annotate manually, or use simulations to test the pipeline. This would then, of course, lead to a sim-to-real gap that would need to be bridged.

Acknowledgements

I would like to sincerely thank Maarten Hogenkamp and the entire PDZ team for their invaluable support throughout this semester thesis. Their expertise, constructive feedback, and ongoing assistance greatly contributed to the successful completion of this work.

References

[1] Thomas E. Williams and E. Christopher Ellison. “Population analysis predicts a future critical shortage of general surgeons.” In: *Surgery* 144.4 (Oct. 2008), pp. 548–556. DOI: 10.1016/j.surg.2008.05.019.

[2] Matthew C. Henn et al. “How We Solved the Shortage of Cardiothoracic Surgeons: Train More or Work Longer.” In: *The Annals of Thoracic Surgery* 119.1 (Jan. 2025), pp. 235–243. DOI: 10.1016/j.athoracsur.2024.07.051.

[3] Carol E. Reiley et al. “Review of methods for objective surgical skill evaluation.” en. In: *Surgical Endoscopy* 25.2 (Feb. 2011), pp. 356–366. DOI: 10.1007/s00464-010-1190-z.

[4] Yilun Chen et al. *Cascaded Pyramid Network for Multi-Person Pose Estimation*. arXiv:1711.07319 [cs]. Apr. 2018. DOI: 10.48550/arXiv.1711.07319.

[5] Zigang Geng et al. *Human Pose as Compositional Tokens*. en. arXiv:2303.11638 [cs]. Mar. 2023. DOI: 10.48550/arXiv.2303.11638.

[6] Jiajie Liu et al. *TCPFormer: Learning Temporal Correlation with Implicit Pose Proxy for 3D Human Pose Estimation*. arXiv:2501.01770 [cs] version: 1. Jan. 2025. DOI: 10.48550/arXiv.2501.01770.

[7] Soyong Shin et al. *WHAM: Reconstructing World-grounded Humans with Accurate 3D Motion*. arXiv:2312.07531 [cs] version: 2. Apr. 2024. DOI: 10.48550/arXiv.2312.07531.

[8] Tsung-Yi Lin et al. *Microsoft COCO: Common Objects in Context*. arXiv:1405.0312 [cs]. Feb. 2015. DOI: 10.48550/arXiv.1405.0312.

[9] Diogo C. Luvizon, David Picard, and Hedi Tabia. *2D/3D Pose Estimation and Action Recognition using Multitask Deep Learning*. arXiv:1802.09232 [cs]. Mar. 2018. DOI: 10.48550/arXiv.1802.09232.

[10] Christian Zimmermann et al. *FreiHAND: A Dataset for Markerless Capture of Hand Pose and Shape from Single RGB Images*. arXiv:1909.04349 [cs]. Sept. 2019. DOI: 10.48550/arXiv.1909.04349.

[11] Haoye Dong et al. *Hamba: Single-view 3D Hand Reconstruction with Graph-guided Bi-Scanning Mamba*. arXiv:2407.09646 [cs] version: 2. Nov. 2024. DOI: 10.48550/arXiv.2407.09646.

[12] Rolandos Alexandros Potamias et al. *WiLoR: End-to-end 3D Hand Localization and Reconstruction in-the-wild*. arXiv:2409.12259 [cs] version: 1. Sept. 2024. DOI: 10.48550/arXiv.2409.12259.

[13] Zhendong Yang et al. *Effective Whole-body Pose Estimation with Two-stages Distillation*. arXiv:2307.15880 [cs]. Aug. 2023. DOI: 10.48550/arXiv.2307.15880.

[14] Weiting Huang et al. *AWR: Adaptive Weighting Regression for 3D Hand Pose Estimation*. arXiv:2007.09590 [cs] version: 1. July 2020. DOI: 10.48550/arXiv.2007.09590.

[15] Jian Cheng et al. *Efficient Virtual View Selection for 3D Hand Pose Estimation*. arXiv:2203.15458 [cs] version: 1. Mar. 2022. DOI: 10.48550/arXiv.2203.15458.

[16] Jonathan Tompson et al. "Real-Time Continuous Pose Recovery of Human Hands Using Convolutional Networks." en. In: *ACM Transactions on Graphics* 33.5 (Sept. 2014), pp. 1–10. DOI: 10.1145/2629500.

[17] Danhang Tang et al. "Latent Regression Forest: Structured Estimation of 3D Articulated Hand Posture." en. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition*. Columbus, OH, USA: IEEE, June 2014, pp. 3786–3793. DOI: 10.1109/CVPR.2014.490.

[18] Sihan Wen, Xiantan Zhu, and Zhiming Tan. *3D WholeBody Pose Estimation based on Semantic Graph Attention Network and Distance Information*. arXiv:2406.01196 [cs] version: 1. June 2024. DOI: 10.48550/arXiv.2406.01196.

[19] Nathan Louis et al. *Temporally Guided Articulated Hand Pose Tracking in Surgical Videos*. arXiv:2101.04281 [cs] version: 2. Oct. 2021. DOI: 10.48550/arXiv.2101.04281.

[20] Rui Wang et al. "POV-Surgery: A Dataset for Egocentric Hand and Tool Pose Estimation During Surgical Activities." en. In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2023*. Ed. by Hayit Greenspan et al. Cham: Springer Nature Switzerland, 2023, pp. 440–450. DOI: 10.1007/978-3-031-43996-4_42.

[21] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. en. Google-Books-ID: si3R3Pfa98QC. Cambridge University Press, 2003.

[22] S. Umeyama. "Least-squares estimation of transformation parameters between two point patterns." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13.4 (Apr. 1991), pp. 376–380. DOI: 10.1109/34.88573.

[23] R.E. Kalman. "A new approach to linear filtering and prediction problems." In: *Journal of basic Engineering* 82.1 (1960), pp. 35–45.