Automated Personnel Activities Observation in the Catheterization Laboratory

Thesis

submitted in partial fulfillment of the
requirements for the degree of

Master of Science

in

Electrical Engineering

by

Yingfeng Jiang B.Sc.
born in Fuzhou, China

This work was performed in:

Circuits and Systems Group
Department of Microelectronics
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology

Delft University of Technology

Delft University of Technology
Department of
Microelectronics

The undersigned hereby certify that they have read and recommend to the Faculty of Electrical Engineering, Mathematics and Computer Science for acceptance a thesis entitled **"Automated Personnel Activities Observation in the Catheterization Laboratory"** by **Yingfeng Jiang B.Sc.** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: 22 July 2022

Chairman: ________________________
Prof . dr. ir. J. Dauwels

Advisor: ________________________
Ir. R.M. Butler

Committee Members: ________________________
Dr. ir. J.J. van den Dobbelsteen

____
Dr. F. Fioranelli

Preface

This thesis marks the end of my two years at the Delft University of Technology. Not surprisingly, this also means the end of my many years of academic studying. This thesis project collaborates with Reinier de Graaf Gasthuis hospital, Delft, which provides the device and real recordings for testing the algorithms in Cath Lab. The thesis would not be possible without the help of many wonderful people.

First of all, I would like to express my gratitude and respect to my daily supervisor Rick Butler. His patience and guidance helped me improve my research skills. I want to thank Prof. Justin Dauwels, who gave me a lot of advice and suggestion. I also want to mention my teammates Jinchen Zeng and Renjie Dai. We exchanged ideas and discussed our project together. I thank Paula Helder-Kouwenhoven for her help in my graduation. Finally, I would like to thank my parents for their unconditional and unequivocal support.

Yingfeng Jiang B.Sc.
Delft, The Netherlands
22 July 2022

Abstract

This thesis presents a method for personnel activities observation, i.e., 3D human pose estimation and tracking, in a Catheterization Laboratory(Cath Lab). We mount five cameras from different angles in the Cath Lab, where surgeons and assistants are in similar clothes while doing surgery. Accurate 3D human pose estimation is the cornerstone of our method. Most previous 3D pose estimation methods train their models directly on a 3D pose dataset. However, these methods are not suitable for our task: i) We do not have enough 3D pose data for training because of privacy issues and specificity ii) The model needs to be retrained in different operating rooms or the camera calibration changed. To solve these problems, we decompose the 3D human pose estimation task into two stages, avoiding the need for large amounts of 3D pose data and retraining. In the first stage, we apply YOLOX and HRNet for 2D human detection and 2D pose estimation. Simultaneously, the 2D object tracking network Bytetrack tracks person identities based on detection results. Then we use a matching algorithm to match the corresponding 2D poses from multiple views and reconstruct 3D poses. Given 3D poses and tracking identities, we, at last, introduce a hybrid method tracking algorithm. By feeding 2D tracking results into the matching and tracking algorithm, we increase the accuracy of the result in a scene where people are wearing similar clothes. We fine-tune and test our method with an operating room dataset. Finally, we validate the method on data from the Cath Lab.

Contents

List of Figures

List of Tables

Abbreviations

3DPS	3D Pictorial Structure
BPE	Back-Projection Error
Cath Lab	Catheterization Laboratory
CNN	Convolutional Neural Network
HRNet	High-Resolution Network
MA	Matching Accurac
MOTA	Multiple Object Tracking Accuracy
RPE	Reprojection Error
YOLO	You Only Look Once
fps	frames per second
mm	millimeter
px	pixel
sec	second

Introduction 1

Workflow analysis is a young research field that has recently gained traction. To automate the classification of workflow and personnel activities, it is important to estimate 3D human poses and track each person in operating rooms. In this thesis, we focus on reconstructing 3D human poses and tracking people based on synchronized videos from several calibrated cameras in a scene where multiple cardiologists and lab assistants are doing surgery in a Catheterization Laboratory (Cath Lab).

Recent multi-view human pose reconstruction approaches can be divided into two types of pipeline. The first approach reconstructs 3D poses with the 3D pictorial structure (3DPS) model, given 2D poses in each camera view [11, 2] estimated by the 2D keypoint detection neural network. The second one, instead of working on 2D poses, extracts pose features from the images with a Convolutional Neural Network (CNN) first. Next, they generate 3D poses directly with 3D CNNs or pose Transformer [34, 33, 35]. These approaches are successful on public datasets because the models can learn poses in specific scenarios by training on 3D pose datasets. However, they must be retrained on different datasets before application to other scenarios, making them hard to generalize. Our algorithm will be applied to an actual Cath Lab, where a strict privacy policy is enforced. It is not easy for us to access new data for retraining. It is also hard to annotate 3D human poses manually from images. Another problem is that the cardiologists and lab assistants always wear sterile clothing in the Cath Lab. Those clothes occlude body curves and legs. Face and hair are covered with a mask and hairnet. As a result, the appearance of all clinicians is similar.

To solve these problems and make our method more generalized, we decompose our method into two stages: 2D image neural networks and 3D reconstruction. In the first stage, we detect 2D human poses with YOLOX [15] and HRNet [30] that are fine-tuned on an operating room dataset [2]. At the same time, we track people in 2D images with pre-trained Bytetrack [36], so we can get the tracking identity of each person in each view. In the second stage, we match corresponding 2D poses from multiple views based on their geometric consistency and 2D tracking results. Next, we reconstruct 3D poses with the linear triangulation method [6, 31, 16]. Given human poses in 3D and match results, we can compute the combination affinity of people between frames. Finally, we tract people in the 3D world based on that affinity.

The main contributions of our work are: i) We introduce 2D tracking results into the multi-view matching algorithm. 2D identities make it possible to trace matching results through frames. The proposed algorithm increases the matching accuracy in Cath Lab. ii) We propose a 3D human tracking algorithm based on 3D poses and 2D tracking identities. We formulate our tracking problem as a weighted bipartite graph matching problem. The weight score is computed from both 2D tracking results and 3D poses. As a result, the proposed tracking algorithm can re-identify and track people without appearance.

In the experimental chapter 3, we perform the quantitative evaluation on the dataset Operating Room [2]. Evaluation metrics are Multiple Object Tracking Accuracy(MOTA) for tracking and Back-projection error (BPE) for pose reconstruction. We also verify the scalability and the stability of our method on Our Cath Lab dataset.

1.1 Problem statement

The problem is defined as follows:
Input:
- Synchronized videos from 5 cameras.
- The camera calibration of those cameras.

Output:
- 3D pose of each person in all video frames.
- Identities of each person between frames.

We want to reconstruct 3D poses of people in 3D world coordinates from videos. Besides, if a person is visible in two consecutive frames, we should detect that this is indeed the same person. The same person in two frames should have the same identity.

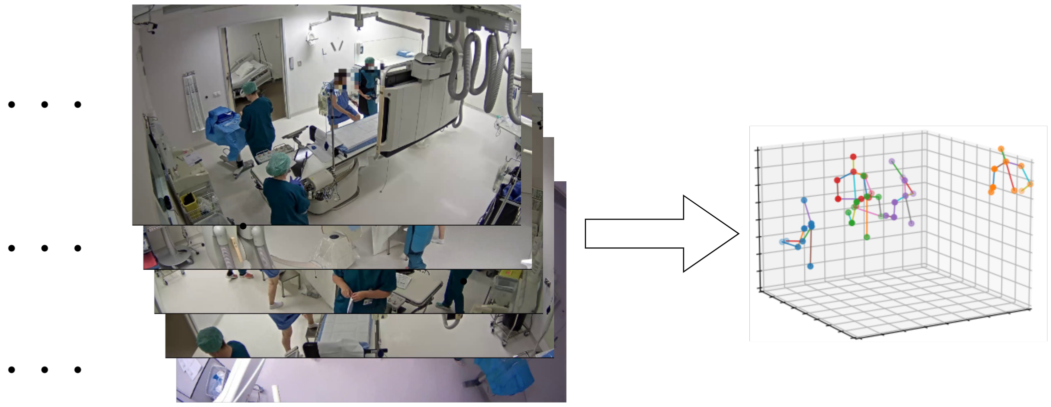


Figure 1.1: An illustration of our problem statement.

1.2 Outline

Chapter 2 discusses in detail the background, literature review and methods developed and their unsuitability in our case. The methodology in **Chapter 3** contains three stages. First, a combination of 2D neural networks is discussed to generate 2D human poses and 2D tracking identities. Second, a matching algorithm with a novel hybrid is described and shows how it combines the information of poses and tracking results. Finally, a novel method of 3D tracking is performed to track people. **Chapter 4** shows the experiment results of our method on the datasets. We try different combinations of strategies and show the increase in performance of our method from the previous one. In **Chapter 5** we make a conclusion and discuss future research.

Background and Related Work 2

2.1 Background

2.1.1 Catheterization Laboratory

The Catheterization Laboratory, also known as “Cath Lab,” is a specialized area in the Reinier de Graaf hospital. In Cath Lab, cardiologists and lab assistants perform minimally invasive tests and advanced cardiac procedures to diagnose and treat cardiovascular disease, called cardiac catheterization. In Cath Lab, there is state-of-the-art imaging technology Figure 2.1, which is used to view the arteries and check the blood environment in the heart. This provides the doctors with diagnostic information to help treat blockages in the arteries often without patients needing surgery. In addition to providing diagnostic information, Cath Lab performs catheter-based interventions or catheter-based treatments of structural heart disease for both acute and chronic cardiovascular illness.

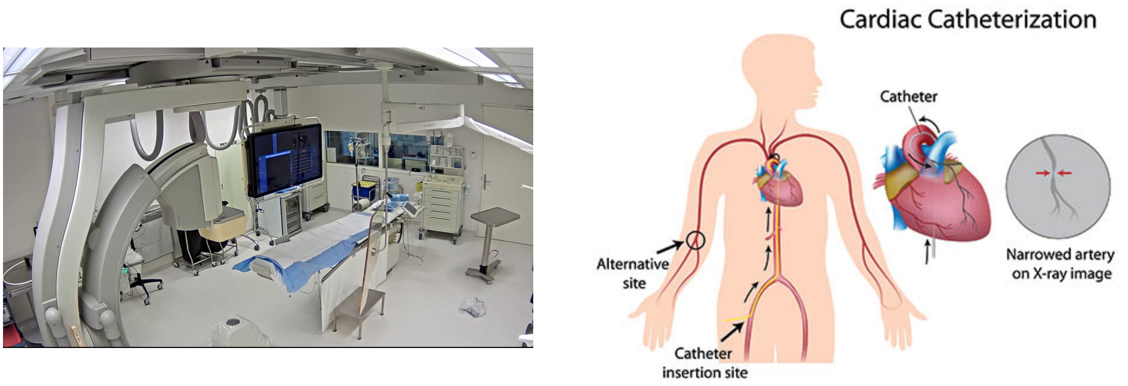


Figure 2.1: Left is an image from Cath Lab. Right is an illustration of cardiac catheterization

Cardiac catheterization is the insertion of a long, thin, plastic tube, called a catheter, into a vein in the arm to the heart. Cardiologists can obtain x-ray pictures of coronary arteries and cardiac chambers and measure pressures in the heart. Sometimes a cardiac catheterization is performed due to an emergency such as a heart attack in progress. Other times, it is performed as a diagnostic tool to check for blockages if the physician suspects the patient may have coronary artery disease. Figure 2.1 shows an illustration of cardiac catheterization.

2.1.2 Human pose

Human pose estimation has been applied in many areas, such as human-computer interaction and health care. Because of the demand in character animation, early human pose estimation applications were in film production motion capture. In human-computer interaction, poses can allow a computer to recognize human gestures so that the person can control the computer directly through movement. It can track the movement of people in specific areas, such as banks, for safety purposes. It is also applied in sports training, which can analyze the joint movement of athletes so as to guarantee that their movements are within safe limits and do not cause injury to the body. More recently, by applying human pose estimation in healthcare processes, machines can monitor operating theatres to optimize workflow. That is what we do in this paper.

The use of human pose began with motion capture systems. In this system, there are multiple calibrated tracker tags attached to various human body parts. Then, Multiple accurate cameras from different views track these tags to obtain a 3D human pose. Such a setup is too complex and time-consuming, making it unavailable for widespread use. Primarily, we cannot set up this complex system in Cath Lab, where cardiologists do surgery.

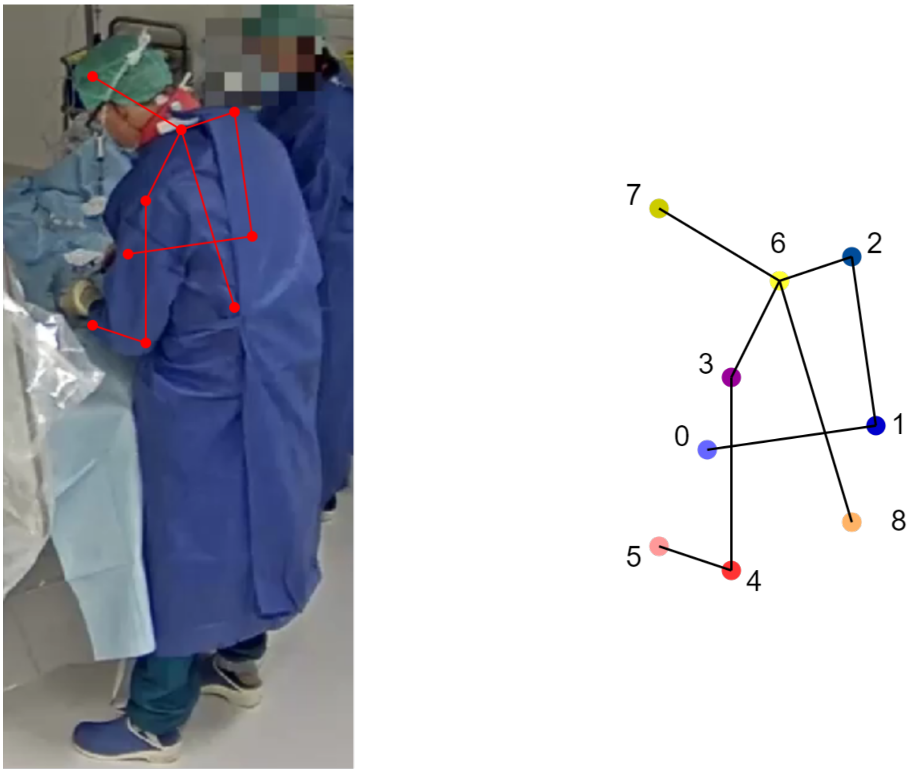


Figure 2.2: A stick model of the human body representing the various anatomical joints.

The human pose can be represented in a stick model, as shown in Figure 2.2. In this type of pose model, body joints are connected via body bones. We can represent a human pose by an order list. Because it is hard to estimate the position of legs, which are not necessary, we will only focus on the upper body posture in this report. In order, our joints are : 0: right wrist, 1: right elbow, 2: right shoulder, 3: left shoulder, 4: left elbow, 5: left wrist, 6: torso, 7: head, 8: stomach.

2.1.3 Neural network

The neural network is a modeling framework that attempts to understand the relationship between inputs and outputs. It is loosely modeled on interconnected animal neurons in a complex network. This network is made up of many biological neurons. The connections of the neurons are modeled as weights between nodes. A positive weight reflects an excitatory connection, while negative values mean inhibitory connections. All inputs are modified by weights and summed. The connections between these neurons allow the neural network to learn hierarchical features from the dataset. In addition, by using other tools such as non-linear activation functions, the neural network is also able to understand the non-linear relationship between input and output. With the back-propagation algorithm (a gradient-based learning method), the weights between individual neurons can be changed according to the objective function.

CNN:

The convolutional neural network (CNN) is a powerful type of neural network which is widely used in computer vision tasks. It plays a vital role in object detection, image classification and keypoint detection. The structure of CNN makes it extract partial information in matrix-based inputs and allows it to process two-dimensional images efficiently.

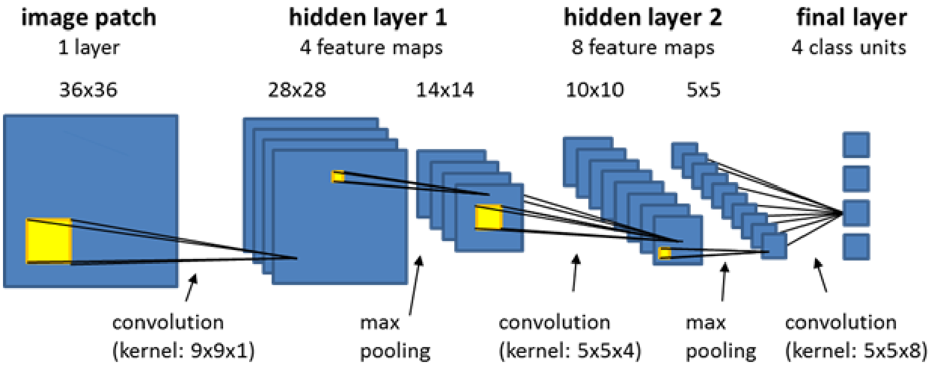


Figure 2.3: An illustration of CNN

CNN has two main operations: convolution and pooling. These operations occur in a series usually. The parameters of a convolution layer are a set of learnable filters (or kernels). When the network is trained on datasets, filters modify their internal values to learn specific image features, such as edges. The pooling layer is a form of non-linear down-sampling. It can decrease the number of the spatial distribution of features generated by the previous convolution filters. The spatial distribution of features in parameters brings translation invariance to the knowledge learned by the filters, which is an important requirement for visual understanding. Another benefit of using CNN rather than a traditional fully connected network is that it reduces the number of parameters that need to be learned, thus simplifying training. Figure 2.3 shows an example of CNN.

2.1.4 Triangulation

Triangulation is an important task in our 3D human pose reconstruction [6, 16]. This process is usually based on a multi-camera system or non-rigid structure from motion. It aims to get the corresponding points and camera projection matrices. For a point $\mathbf{X}$ in 3D world coordinates, it is projected to 2D image by the camera, satisfying the equation: $\mathbf{x}_c = \mathbf{P}_c\mathbf{X}$, where $\mathbf{x}_c$ and $\mathbf{P}_c$ are the image point and the camera projection matrix in view c. In triangulation, $\mathbf{x}_c$ and $\mathbf{P}_c$ of this projection equation are known and we want to compute the 3D point $\mathbf{X}$ from $\mathbf{x}_c$ and $\mathbf{P}_c$.

Theoretically, we only need two of these equations from different views to compute the location of a 3D point because these two lines will intersect at one point for the ideal case. However, the actual situation is not the case. Because of noise, the two lines may not have an intersection in most real cases. Moreover, the intersection point of two lines may sometimes deviate from the ground truth point. This problem will be more complicated for multi-view cases because multiple lines may have multiple different intersections. Then the problem becomes determining the optimal estimation of the 3D point $\mathbf{X}$ from multiple views. To solve this complex problem, the straightforward methods are the Linear method and the Midpoint method [16]. They can directly solve the problem. Besides, we can set the L_2 norm in the 2D image, representing the geometric distance of Euclidean space, as a cost function. Then we find the optimal 3D points that minimize the L_2 norm.

Linear method:

As described before:

$$\mathbf{x}_c = \mathbf{P}_c\mathbf{X} \tag{2.1}$$

where $\mathbf{x}_c$ is the corresponding point of the 3D joint point $\mathbf{X}$ on the c-th 2D view and $\mathbf{P}_c$ is its camera matrix:

$$\mathbf{x}_c = (x_c, y_c, 1)^{\mathrm{T}} \tag{2.2}$$

$$\mathbf{P}_c = \begin{bmatrix} \mathbf{p}_c^{1\ \mathrm{T}} \\ \mathbf{p}_c^{2\ \mathrm{T}} \\ \mathbf{p}_c^{3\ \mathrm{T}} \end{bmatrix} \left(\mathbf{P}_c \in \mathbb{R}^{3\times 4}\right) \tag{2.3}$$

If we want to solve Triangulation problem from multiple views, we can have this matrix from the points and projection matrices:

$$\mathbf{C} = \begin{bmatrix} x_1\mathbf{p}_1^{3\ \mathrm{T}} - \mathbf{p}_1^{1\ \mathrm{T}} \\ y_1\mathbf{p}_1^{3\ \mathrm{T}} - \mathbf{p}_1^{2\ \mathrm{T}} \\ x_2\mathbf{p}_2^{3\ \mathrm{T}} - \mathbf{p}_2^{1\ \mathrm{T}} \\ y_2\mathbf{p}_2^{3\ \mathrm{T}} - \mathbf{p}_2^{2\ \mathrm{T}} \\ \cdots \end{bmatrix} \tag{2.4}$$

And we want to find the 3D point $\mathbf{X}$ that can minimize:

$$\begin{aligned} &\min \|\mathbf{CX}\| \\ &\text{s.t } \|\mathbf{X}\| = 1 \end{aligned} \tag{2.5}$$

As a result, the triangulation problem becomes a least squares problem for solving a homogeneous linear equation. We call this method the Linear-Eigen method [16] (Linear method).

L_2 **method:**

Another method is to find the point $\mathbf{X}$ that minimizes L_2 norm cost function:

$$\min_{\mathbf{X}} \sum_{c} \|\mathbf{x}_c - \mathbf{P}_c \mathbf{X}\|_2 \tag{2.6}$$

Obviously, this is a nonlinear optimization problem. The Levenberg-Marquardt method [24] is the most commonly used method for optimization [6].

2.2 Related research

Multi-view 3D human pose estimation: In this work, we focus on multi-person 3D pose estimation. Some previous works are based on 3DPS models, which encode 3D locations of body joints and pairwise relations between them [4, 1, 12, 2]. A joint detector gives the likelihood of a joint being at some 3D location based on associated 2D points from all views. And skeletal constraints give the pairwise potentials between joints [4, 1]. Then, the 3D poses of multiple people are jointly inferred by maximum a posteriori estimation. Other recent methods focus on generating 3D poses with neural networks[33, 34, 35]. The features of images from all views are extracted by CNN, following with 3D neural networks [33, 35] or transformers[34]. Because we cannot have 3D pose datasets from operating rooms, we choose the traditional linear Triangulation method[6, 31] to reconstruct 3D poses.

Single-view 2D human pose estimation: There is a large amount of research on human pose estimation from a single image. Single-person pose estimation [19, 26, 32] localizes 2D body keypoints of a person in a cropped image. There are two categories of multi-person pose estimation methods: 1) top-down methods [30, 17, 8, 19] that first detect people in the image and then apply single-person pose estimation to the cropped image of each person 2) bottom-up methods [5, 27, 21] that first detect all keypoints and then group them into different people. In general, the top-down methods are more accurate, while the bottom-up methods are relatively faster. In our work, We choose the High Resolution Network[30], a state-of-the-art approach for multi-person pose detection, as the top-down pose estimator in our pipeline.

Person re-ID and multi-image matching: Person re-ID aims to identify the same person in different images[37] based on appearance. However, in our case, we cannot use other pre-trained models because people are no difference in appearance in Cath Lab. Multi-image matching is to find feature correspondences among a collection of images [18, 38]. We use the results on cycle consistency[18] to solve the correspondence problem in multi-view pose estimation.

3 Methodology

Figure 3.1 shows the pipeline of our approach. First, a combination of 2D image neural networks generates 2D keypoints and tracks the identities of people from all views. Then, given with 2D results, the 3D pose reconstruction model produces joint locations in 3D world coordinates. Given 3D poses and identities of people, a tracking method is adopted to track people in the 3D world based on keypoint locations and 2D track identities.

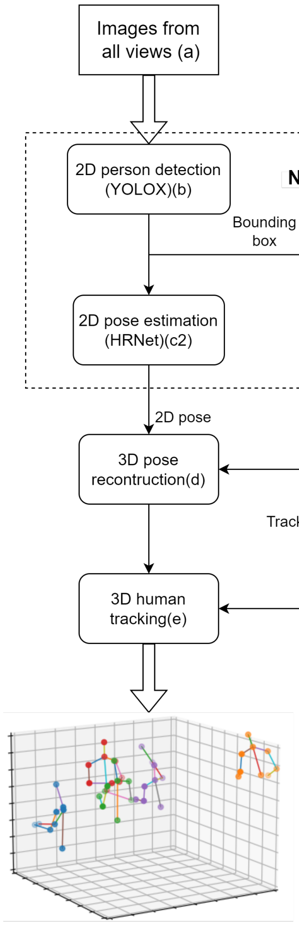

Images from all views (a)
Neural Network
2D person detection (YOLOX)(b)
Bounding box
2D pose estimation (HRNet)(c2)
2D person tracking (Bytetrack)(c1)
2D pose
3D pose recontruction(d)
Track identity
3D human tracking(e)

Figure 3.1: An overview of our approach. Given frames from the calibrated cameras(a), a fine-tuned human detector YOLOX is used to produce 2D bounding boxes in each view(b). Then 2D tracker Bytetrack generates tracking identities(c1) and 2D pose detector HRNet produces 2D poses(c2). 3D pose of each person is reconstructed based on 2D poses from each view and tracking identities(d). Finally, 3D human tracker tracks people from 3D poses and their identities(e).

3.1 2D image neural networks

3.1.1 Human detection and tracking in images

We use YOLOX [15] as our 2D human detection network. YOLOX is a high-performance object detector that makes several improvements to the YOLO series. Because of its high flexibility, YOLOX can be the detector of our multi-person tracking algorithm Bytetrack [36].

Bytetrack [36] is a potent Multi-Object Tracking network with high precision and high speed. It can track almost every detection bounding box instead of ignoring the low score ones.

We use the models that are trained by open-mmlab [7, 10]. YOLOX-x [15] is pre-trained on the COCO datasets [23] for 2D human detection firstly and then is fine-tuned on the Operating Room [2] dataset. Bytetrack is trained on the MOT17 [25] and Crowdhuman [29] datasets.

3.1.2 Human pose estimation in images

We adopt High-Resolution Network (HRNet) [30] as our top-down human pose estimator in images. HRNet can maintain high-resolution information throughout the whole network. This network contains high resolution convolution stream in parallel with traditional high-to-low convolution streams. Different resolution streams exchange information after each procession stages [30].

Our HRNet-W48 is pre-trained on COCO [23] datasets by mmpose [9] and is fine-tuned on Operating Room dataset [2].

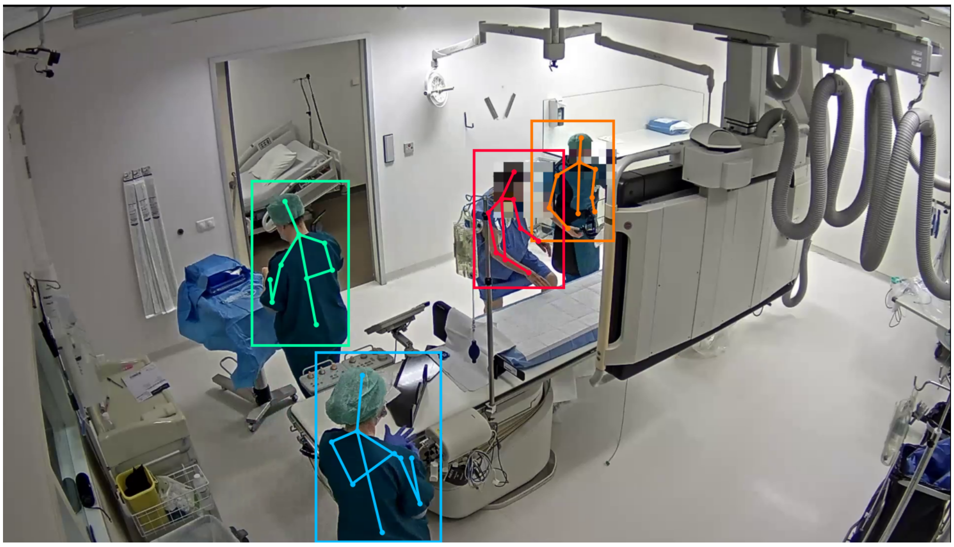


Figure 3.2: An example of the result from 2D networks

3.2 3D human pose reconstruction

Before we reconstruct the 3D poses with linear Triangulation method [6], we need to match the 2D poses belonging to the same person with a cross-view matching algorithm between views.

3.2.1 Cross-view matching with geometric and tracking affinity

Similar to the method in [11],firstly we adopt a method to measure the likelihood (a.k.a. affinity) that two detections from different views. Then, a matching algorithm establishes the correspondence matrix based on affinity. Because there are 5 views in our case, we should consider the cycle-consistency constraint. The matching results between view to view should be a closed cycle. Therefore, we cannot solve matching problem separately for each pair of views.

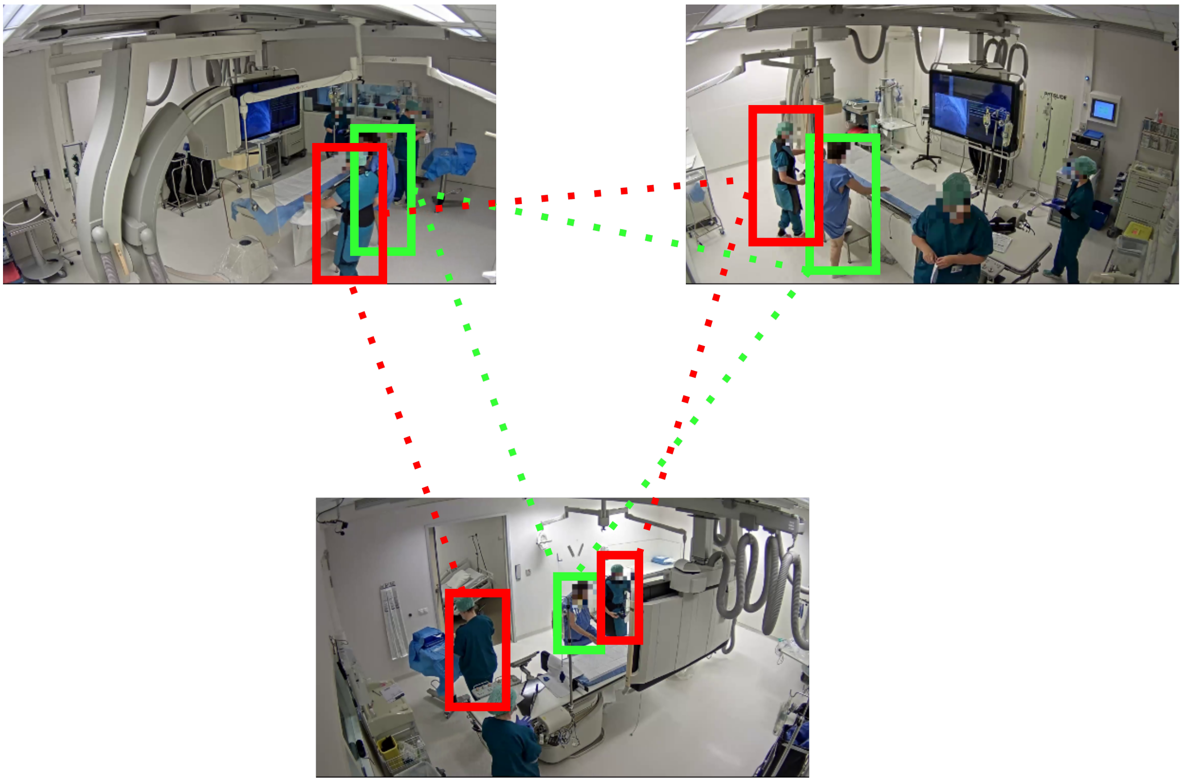


Figure 3.3: An illustration of cycle consistency. The green lines denote a set of consistent correspondences and the red lines show a set of inconsistent correspondences.

Suppose there are M detections $\{D_i \mid i = 1, \ldots, M\}$ in total in all camera views combined and $\boldsymbol{x_i} \in \mathbb{R}^{N\times 2}$ denotes the 2D pose composed of N joints of each detection. $\boldsymbol{A} \in \mathbb{R}^{M\times M}$ denotes the affinity matrix, whose element $A_{i,j}$ represents the affinity score between D_i and D_j. A partial permutation matrix $\boldsymbol{P} \in \{0, 1\}^{M\times M}$ represents the correspondences between a pair of different detections. Then we need to find the optimal partial permutation matrix $\boldsymbol{P}$ based on affinity matrix $\boldsymbol{A}$.

The original method in [11] combines appearance similarity and geometric consistency to calculate the affinity scores between two detections. The feature vectors of cropped images of each person can be extracted from a public pre-trained person re-identification network [37]. And then compute the negative Euclidean distance between the feature vectors of a detection pair as the appearance affinity score of that pair.

Besides the appearance affinity, another significant cue to associate two detections is that their 2D poses should be geometrically consistent. In other words, a joint in the first detection should lie on the epipolar line of the associate joint in the second detection. The epipolar line distance between pose $\boldsymbol{x}_i$ and pose $\boldsymbol{x}_j$ from detections D_i and D_j can be computed as:

$$d_e\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = \frac{1}{2N} \sum_{n=1}^{N} d_g\left(\boldsymbol{x}_i^n, \boldsymbol{L}_{ij}\left(\boldsymbol{x}_j^n\right)\right) + d_g\left(\boldsymbol{x}_j^n, \boldsymbol{L}_{ji}\left(\boldsymbol{x}_i^n\right)\right) \tag{3.1}$$

where $\boldsymbol{x}_i^n$ is the 2D location of the n-th joint of detection D_i, $\boldsymbol{x}_j^n$ is the n-th joint of detection D_j, D_i and D_j are from different views. $\boldsymbol{L}_{ij}\left(\boldsymbol{x}_j^n\right)$ is the epipolar line associated with $\boldsymbol{x}_j^n$ in the view of D_i. and $d_g(\cdot, l)$ is the point-to-line distance for l. After computing d_e between all pairs of detections, we normalized the negative d_e and get normalized distance d_{ij}.

The final geometric affinity scores A_{ij}^g is computed by sigmoid function:

$$A_{ij}^g = \frac{1}{1 - e^{-d_{ij}}} \tag{3.2}$$

As a result,the distances d_g are mapped to values in $(0, 1)$. And if D_i and D_j are in the same view, we set $A_{ij}^g = 0$.

Because our algorithm will be applied in the Cath Lab, where cardiologists and lab assistants wear sterile clothing, it is hard for the neural network to distinguish and re-identify them from different views. We introduce tracking identity affinity instead of using appearance affinity. We assume that our tracking algorithm Bytetrack [36] has excellent consistency between two adjacent frames. Therefore, we search the previous matching result based on the tracking ID of detection D_i and find the associate IDs from other views.

$$A_{ij}^t = \begin{cases} 1, & \text{if } I_i \, \text{in} \, m_j \\ 0, & \text{otherwise} \end{cases} \tag{3.3}$$

where A_{ij}^t presents tracking identity affinity score between D_i and D_j, I_i is the tracing identity of D_i. m_j is the previous match result that contains D_j. We combine the two affinity matrices as follows:

$$A_{ij} = \sqrt{(1 - w_g) {A_{ij}^t}^2 + w_g {A_{ij}^g}^2} \tag{3.4}$$

where A_{ij}, A_{ij}^{track} and A_{ij}^g denote values of the fused affinity score, tracking identity affinity score and geometry affinity score of detection pair (i,j).

After computing the elements of affinity matrix $\boldsymbol{A}$, we need to find the optimal partial permutation matrix $\boldsymbol{P}$. We minimize the following loss function [11] to estimate

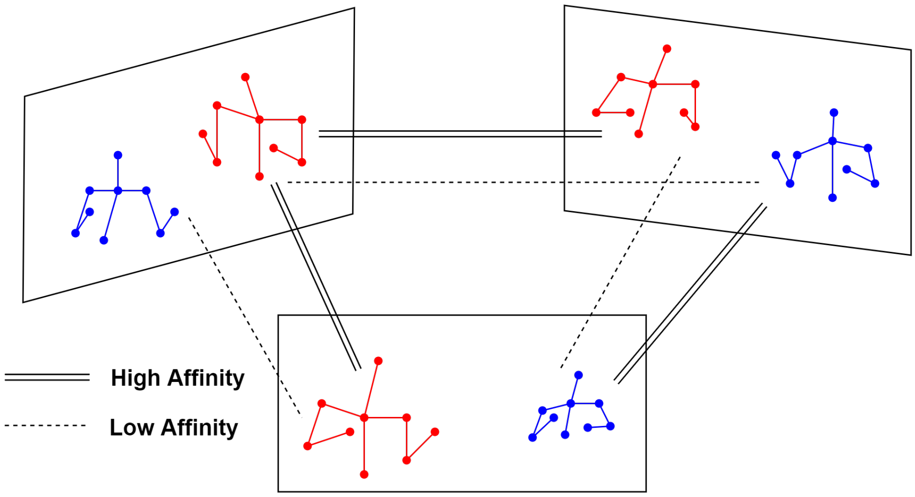


Figure 3.4: 2D poses from the same person are in the same color. A pair of 2D poses from two views has high affinity score if they are from the same person.

the optimal matrix $\boldsymbol{P}$:

$$\begin{aligned}\Lambda(\boldsymbol{P}) &= -\sum_{i=1}^{M}\sum_{j=1}^{M} A_{ij}P_{ij} + \lambda \cdot \mathrm{rank}(\boldsymbol{P}) \\ &= -\langle \boldsymbol{A}, \boldsymbol{P}\rangle + \lambda \cdot \mathrm{rank}(\boldsymbol{P})\end{aligned} \tag{3.5}$$

where λ is the Lagrange multiplier.

Instead of minimize rank($\boldsymbol{P}$) directly, we choose to minimize the nuclear norm $\|\boldsymbol{P}\|_*$. Because it is the tightest convex surrogate of rank [14], the optimization changes to a convex problem. And we replace the integer constraint on $\boldsymbol{P}$ that $\boldsymbol{P}$ is a real matrix with values in [0, 1]. Finally, we solve the following convex optimization problem with the alternating direction method of multipliers (ADMM) [3]:

$$\begin{aligned}&\min_{\boldsymbol{P}} -\langle \boldsymbol{A}, \boldsymbol{P}\rangle + \lambda\|\boldsymbol{P}\|_*, \\ &\text{s.t. } \ 0 \le \boldsymbol{P} \le 1, \\ &\qquad \boldsymbol{P}_{ij} = \boldsymbol{P}_{ji}, \\ &\qquad 0 \le \boldsymbol{P}_{ij}\mathbf{1} \le 1, 0 \le \boldsymbol{P}_{ij}^T\mathbf{1} \le 1,\end{aligned} \tag{3.6}$$

3.2.2 3D pose reconstruction with linear Triangulation method

Given the matched 2D poses of a person in different views, we can reconstruct the 3D pose. Because one error matched detection will have large impact in reconstruction, we evaluate and rank the confidence of 2D poses of each person:

$$C_i = c_{d,i} + \frac{1}{N}\sum_{n=1}^{N} c_{k,i}^n \tag{3.7}$$

where $C_i, c_{d,i}, c_{k,i}^n$ are the confidence, confidence of detection and confidence of *n-th* joint of detection D_i. The error matched detection usually has low confidence because of overlapping. We sort the detections based on their confidence and choose top three views for reconstruction. Because we do not know the pose prior information in the Cath Lab, we cannot generate 3D poses based on the 3D Pictorial Structure model (3DPS) as [11, 4] do. Instead, we adopt traditional linear Triangulation method [6] with the views we choose.

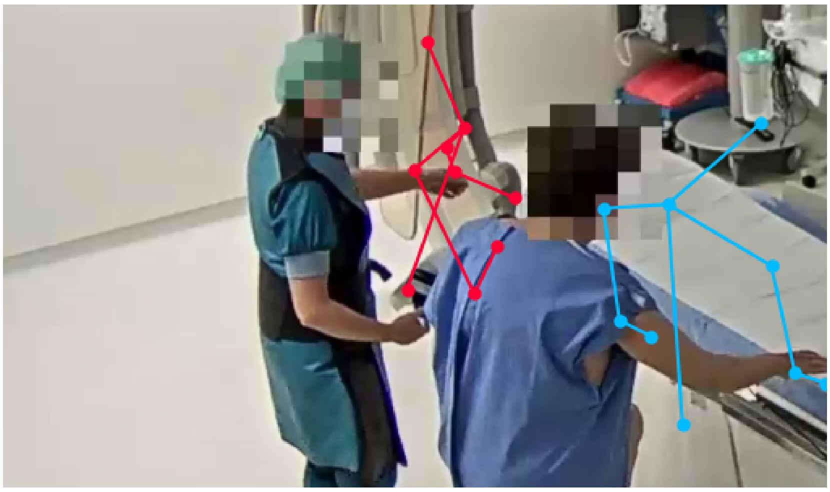


Figure 3.5: An example of error caused by the wrong matching result

3.3 Track person in 3D with hybrid affinity method

The previous modules have generated the 3D poses from the 2D poses in one frame. Then we start to track people in timeline, i.e., we need to transfer the historical person identity to the poses in new frames. Even if we have the 2D tracking identities of each 2D detection, we cannot use them directly. Because there may be tracking error in one view, and the match result is not always correct. A better option is to consider both 3D poses and 2D tracking results.

We retain the historical states of persons in the scene as tracked targets. The problem becomes associating these historical targets with the newly generated targets. Suppose there are M people's poses (targets) $\{T_{i,t} \mid i = 1, \ldots, M\}$ generated in the new frame and N tracked targets $\{T_{j,t'} \mid j = 1, \ldots, N\}$ in the previous frame. We use $\mathbf{X}^k_{i,t} \in \mathbb{R}^3$ to present the 3D location of k-th joint of $T_{i,t}$. Existing target $T_{j,t'}$ has a absolute 3D tracking identity $e_{j,t}$. For a frame at time t, a list of track identities $\mathbf{m}_{i,t} = [I_{i,t,1}, \ldots, I_{i,t,5}]$ presents the match result of $T_{i,t}$, where $I_{i,t,c} = \{1, 2, \ldots\}$ is tracking identity of that target in camera view c.

We can compute an affinity matrix $\boldsymbol{B} \in \mathbb{R}^{M \times N}$ between each new target and old target. Technically, this is a weighted bipartite graph matching problem, which can be solved efficiently with the Hungarian algorithm [22]. Therefore, our problem is to measure the affinity of each pair of old-new targets. Because we only want to match targets from different frames, we omit the index i, j in the following discussion for notation simplicity.

Given a pair of target $(T_{t'}, T_t)$, the affinity is measured from both 3D geometric correspondences and track identity affinity:

$$B\left(T_{t'}, T_t\right) = \sum_{k=1}^{K} B_{3D}\left(\mathbf{X}_{t'}^k, \mathbf{X}_t^k\right) + B_T(T_{t'}, T_t) \tag{3.8}$$

The 3D correspondence is computed based on the sum of the distance of new targets joint $\mathbf{X}_t^k$ and previous targets joint $\mathbf{X}_{t'}^k$ in the 3D world coordinate. Because the spines of cardiologists and lab assistants are very stable between frames, we only compute the sum of the distance from Head, Torso and Stomach three joints. Each joint is computed independently, so we omit the index k in the following discussion for notation simplicity:

$$B_{3D}\left(\mathbf{X}_{t'}, \mathbf{X}_t\right) = w_{3D}\left(1 - \frac{\|\mathbf{X}_t - \mathbf{X}_{t'}\|}{th_{3D}}\right) \tag{3.9}$$

where w_{3D} stands for the weight of 3D correspondence. th_{3D} is a threshold of 3D distance. If the distance is larger than th_{3D}, B_{3D} will be less than 0. B_{3D}>0 indicates these two joints have a probability that they come from the same person.

Because the used 2D tracking algorithm has high accuracy, we assume that the same target has a almost the same $\mathbf{m}$ between frames. We can measure the affinity with the number of equal tracking identities. The affinity of track identities of two targets can be computed as:

$$B_T(T_{t'}, T_t) = w_t \frac{C\left(\mathbf{m}_t, \mathbf{m}_{t'}\right)}{N\left(\mathbf{m}_t, \mathbf{m}_{t'}\right) + b} \tag{3.10}$$

where w_t is the weight of track identity affinity. $N\left(\right)$ counts the number of non zero I_c in $\mathbf{m}_t$ and $\mathbf{m}_{t'}$, i.e., the number of views where the target appears . $C\left(\right)$ computes the number of I_c are the same. b decreases the affinity when the number of views is low. Algorithm 1 shows the procedure.

Given previous affinity measurement, we introduce how we initialize and update targets. As for the targets initialization in the first frame, we directly save their poses and identity information. To update the targets in later frames, we compute the affinity matrix between new and old targets as Equation 3.8 and then solve this association problem with the Hungarian algorithm [22]. Each new target is either assigned to an existing target or labelled as unmatched based on the association results. If a previous target has low affinity to all new targets, we mark it as missing target and save its poses and $\mathbf{m}$. The unmatched targets will first be matched with previous missing targets, and the rest will be labelled as additional targets. The previous targets , which are not assigned to any new target, will also be saved in memory as missing targets for the next frame.

Algorithm 1: 3D tracking procedure

```
Input: New 3D human targets $\mathbb{T}_t = \{T_{j,t} \mid j = 1, \dots, M\}$
Previous targets $\mathbb{T}_{t'} = \left\{T_{i,t'} \mid j = 1, \dots, N\right\}$
Previous unmatched targets $\mathbb{U}_{t'}$;
Output: Set $\mathbb{T}_{j,t}$ with identity $e_{j,t}$ ;
Initialisation: $\mathbf{B} \leftarrow \mathbf{B}_{N \times M} \in 0^{N \times M}, e_t \leftarrow 0$;
/* Compute affinity matrix */
foreach $\mathbf{B}(i,j) \in \mathbf{B}$ do
    $\mathbf{B}(i,j) \leftarrow B\left(T_{i,t'}, T_{j,t}\right)$
/* Solve matching problem */
$Indices_{T_{t'}}, Indices_{T_t} \leftarrow HungarianAlgorithm(\mathbf{B})$
/* Update identity */
foreach $(i,j) \in Indices_{T_{t'}}, Indices_{T_t}$ do
    $e_{j,t} \leftarrow e_{i,t'}$
/* Match with previous unmatched targets */
foreach $U_{t'} \in \mathbb{U}_{t'}, j \notin Indices_{T_t}$ do
    if $B_u\left(U_{t'}, T_{j,t}\right) > threshold$ then
        $e_{j,t} \leftarrow e_{U_{t'}}$;
        $\mathbb{U}_t \leftarrow \mathbb{U}_{t'}$ - $U_{t'}$;
/* Add new targets */
foreach $T_{j,t},\ e_{j,t} = 0$ do
    $e_{j,t} \leftarrow NewIdentity()$
/* Save unmatched targets */
foreach $i \notin Indices_{T_{t'}}$ do
    $\mathbb{U}_t \leftarrow \mathbb{U}_t \cup T_{i,t'}$
```

4 Experiment and Discussion

4.1 Datasets

Operating Room [2]: It is a dataset consisting of ten recordings captured by five calibrated cameras. Four to six people are doing acting surgery in an operating room. In this case, the recordings are not of actual clinical procedures. Instead, the activities are played by actors. The videos run at a resolution of 1280x720 pixels with a framerate of 1 Hz. The dataset contains a total of 700 frames per view. It has the annotation of upper body 2D pose, bounding box and identity of each person from all views.

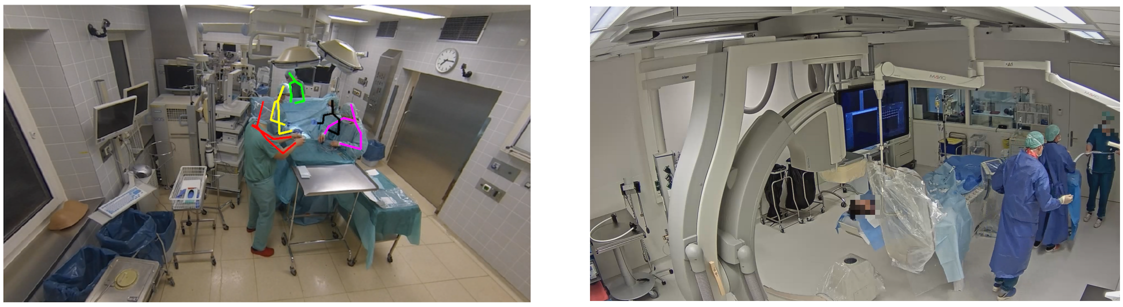


Figure 4.1: Left is an image from Operating Room. Right is from Our Cath Lab

Our Cath Lab: Five calibrated cameras capture this dataset in Cath Lab in Reinier de Graaf Gasthuis, Delft, NL, during real Cardiac catheterization surgery. The videos run at a resolution of 1920x1080 pixels with a framerate of 25 Hz. For the lack of ground truth, we qualitatively evaluate our approach to this dataset. Because of privacy, the faces of people are blurred.

Table 4.1: Description of the datasets, Operating Room and Our Cath Lab

Datasets	**Operating Room**	**Our Cath Lab**
Cameras	5	5
Frame dt	1sec	0.04sec
Resolution	1280x720px	1920x1080px
Real	No, acted	Yes
Environment	Instrument Room	Cath Lab
Annotated	Yes	No

4.2 Fine-tuning

In this section, we introduce our fine-tuning procedure for HRNet and YOLOX.

4.2.1 HRNet

We fine-tune our human pose estimator HRNet on Operating Room dataset. The bounding boxes will be the input during fine-tuning because HRNet is a top-down pose network. The model was pretrained on COCO2017 [23] dataset. We modify the number of output channels to 9 to fix the keypoints in Operating Room. The loss function we chose is Joint mean squared error(JointMSE). The optimizer is Adam [20] and the learning rate is 1e-4. Figure 4.2 shows the Loss and Precision during training. The network converged rapidly after six epochs.

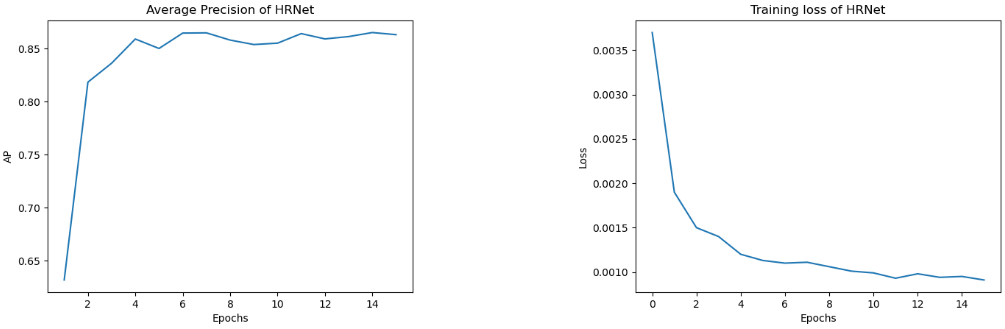


Figure 4.2: The Average Precision and Loss of HRNEt during training

We also fine-tune some other human pose networks. Table 4.2 shows the compassion between the result of different models.

Table 4.2: Fine-tuning results of different pose models

Pose Model	AP .50:.95	AP .50	AP .75
HRNet-w48 [30]	0.8601	0.9981	0.9771
ResNet50 [17]	0.7913	0.9832	0.9645
Alphapose [13]	0.7642	0.9803	0.9322

4.2.2 YOLOX

We fine-tune our human detector YOLOX on Operating Room dataset as well. The model was pretrained on COCO2017 [23] dataset. We modify the number of output channels to 1 because we only need to detect humans. The optimizer is Stochastic Gradient Descent(SGD). And we set the learning rate as 1e-4. Figure 4.3 shows the Loss and Precision during training. The network converged after 20 epochs.

Table 4.3: Fine-tuning results of different human detection models.

Detection Model	AP .50:.95	AP .50	AP .75
YOLOX [15]	0.8024	0.9715	0.9381
Faster-RCNN [28]	0.7313	0.9363	0.8810

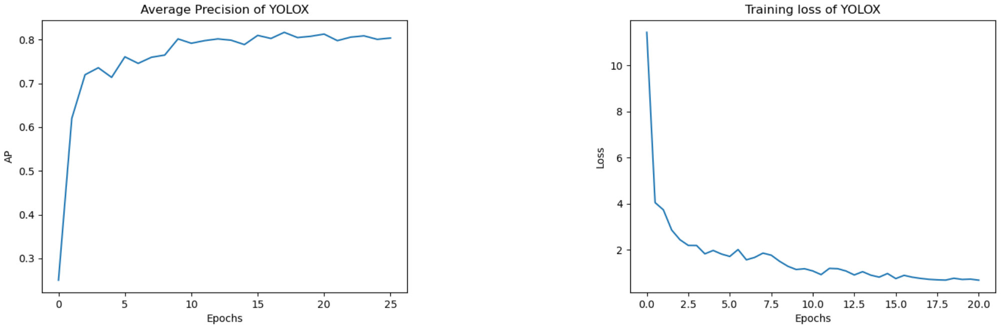


Figure 4.3: The Average Precision and Loss of YOLOX during training

4.3 Evaluation metrics

4.3.1 Back-projection Error(BPE)

Instead of measuring the distance between ground truth 3D points and joints reconstructed, we compute the distance from the back-projection line to the joint (Figure 4.4). We suppose that cameras are well-calibrated and the projection matrix of camera c is provided as $\mathrm{P}_c \in \mathbb{R}^{3\times 4}$. The back-projection line, a ray in 3D space, is computed as:

$$\tilde{\mathbf{X}}^n_{t,c}\left(\mu; \mathbf{x}^n_{t,c}\right) = \mathrm{P}^+_c \mathbf{x}^n_{t,c} + \mu \mathbf{X}_c \tag{4.1}$$

where P^+_c is the pseudo-inverse of P_c, $\mathbf{x}^n_{t,c}$ is the 2D location of n-th joint and $\mathbf{X}_c$ is the 3D location of the camera center. μ is the variable for the line function.

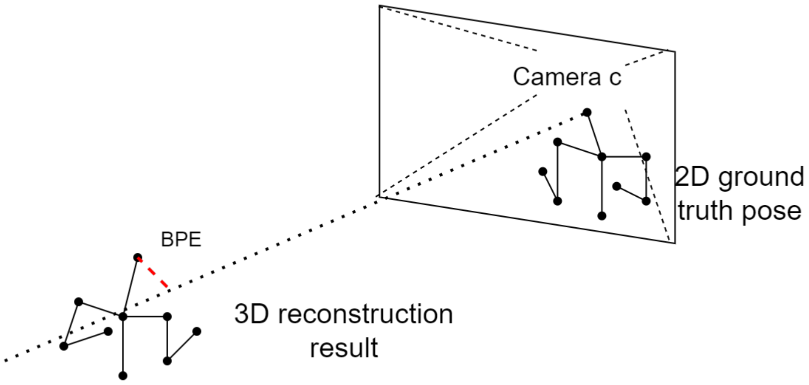


Figure 4.4: An illustration of BPE.

BPE for n-th joint from a view can be computed as:

$$\mathrm{BPE}^n_c = \frac{1}{F}\sum_t^F d_l(\tilde{\mathbf{X}}^n_{t,c}\left(\mu\right), \mathbf{X}^n_t) \tag{4.2}$$

where $d_l(,)$ is the 3D point-to-line distance. $\tilde{\mathbf{X}}^n_{t,c}\left(\mu\right)$ is the back-projection line for $\mathbf{x}^n_{t,c}$ from camera c. F is the total number of frames.

4.3.2 Reprojection Error(RPE)

Besides BPE, Reprojection Error(RPE) is another metric for evaluating the reconstruction. We reproject our reconstructed poses to the 2D images. Then we measure the distance between the ground truth 2D point and the reprojection point in the image. As shown in Figure 4.5, **X**, **O** and **x** are the reconstructed point, the center of the camera and the 2D ground truth point. The red line is RPE.

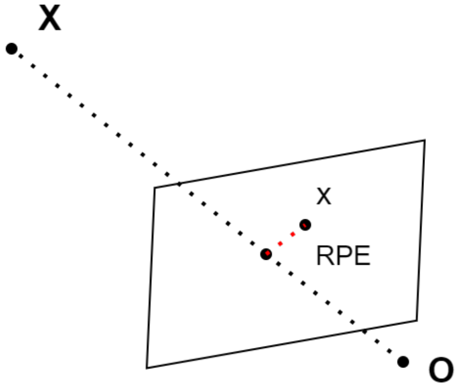


Figure 4.5: An illustration of Reprojection Error.

The reprojected point of n-th joint can be directly computed:

$$\tilde{\mathbf{x}}_{t,c}^{n} = \mathbf{P}_c \mathbf{X}_t^n \tag{4.3}$$

where $\mathbf{P}_c$ is the projection matrix of camera c. RPE for n-th joint from view c can be computed as:

$$\mathrm{RPE}_c^n = \frac{1}{F} \sum_t^F \left\| \tilde{\mathbf{x}}_{t,c}^{n} - \mathbf{x}_{t,c}^{n} \right\|_2 \tag{4.4}$$

4.3.3 Multiple Object Tracking Accuracy(MOTA)

We choose Multiple Object Tracking Accuracy(MOTA) metric to evaluate our 3D tracking result.

$$\mathrm{MOTA} = 1 - \frac{\sum_t (m_t + fp_t + mme_t)}{\sum_t g_t} \tag{4.5}$$

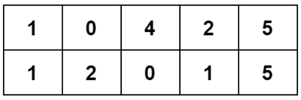
where m_t, fp_t ,g_t and mme_t are the number of misses, of false positives, ground truth detection and of mismatches respectively for time t.

4.3.4 Matching Accuracy(MA)

We introduce Matching Accuracy as the evaluation metric of our multi-view matching algorithm:

$$\mathrm{MA} = \frac{\sum_t c_t}{\sum_t d_t} \tag{4.6}$$

where c_t is the number of detection that is matched correctly, d_t is the number of total detection. Figure 4.6shows an example of a matching result for a person. The first row is the ground truth, and the second is the matching result. Each row presents the identity of detection from each view. For this person, only two detections are matching correctly.



1	0	4	2	5
1	2	0	1	5

Figure 4.6: An example of matching result.

4.4 Ablation analysis results

4.4.1 Different Triangulation methods

There are many Triangulation methods we can choose. We test Linear method, the simplest one, and L_2 norm method, the most common one [6], on our datasets.
Table 4.4 shows BPE and RPE of each joint reconstructed by two Triangulation methods on Operating Room dataset. We compute BPE from all views and take an average over people and views. Linear method has more minor errors in BPE. The L_2 method is good at RPE because RPE is the cost function that L_2 method wants to minimize. Overall it seems that the two methods have little difference in precision. We can find that both methods reconstruct Torso, Head and two Shoulders with less error. The reason may be that it is easier to estimate these four joints than others when people are wearing sterile clothing.

Table 4.4: BPE and RPE of each joint in Operating Room

Operating Room	Linear:BPE(mm)	Linear:RPE(px)	L_2:BPE(mm)	L_2:RPE(px)
Right Wrist	76.54	18.06	78.82	17.97
Right Elbow	66.56	16.26	68.51	16.19
Right Shoulder	53.16	13.73	53.60	13.42
Left Shoulder	54.05	13.59	54.23	13.00
Left Elbow	65.62	16.05	67.13	15.81
Left Wrist	76.53	17.76	78.67	17.55
Torso	50.98	13.84	51.31	13.49
Head	54.13	15.80	53.73	15.18
Stomach	68.04	15.35	70.71	15.33
Average	62.84	15.64	64.08	15.34

Table 4.5 shows BPE and RPE measured from each camera view.

We also test these two Triangulation methods on Our Cath Lab dataset. This time, we let the results from 2D pose estimation as reference, i.e., we measure the difference between 2D estimated poses and 3D reconstructed poses. As Table 4.6 shows,

Table 4.5: BPE and RPE from each camera view in Operating Room

Operating Room	Camera1	Camera2	Camera3	Camera4	Camera5
Linear:BPE(mm)	76.54	53.93	60.00	61.27	62.98
Linear:RPE(px)	19.22	11.75	16.06	15.38	15.67
L_2:BPE(mm)	77.11	56.31	62.41	62.59	62.42
L_2:RPE(px)	18.66	11.96	16.11	15.12	14.84

the results have larger errors than that of Operating Room on both BPE and RPE. Although the change of resolution may cause an increase in RPE, BPE is an absolute value. L_2 method is better than Linear method in both BPE and RPE on this dataset.

Table 4.6: BPE and RPE of each joint in Our Cath Lab

Our Cath Lab	Linear:BPE(mm)	Linear:RPE(px)	L_2:BPE(mm)	L_2:RPE(px)
Right Wrist	111.31	39.98	108.91	39.61
Right Elbow	110.92	40.27	109.16	39.48
Right Shoulder	109.35	40.74	107.55	39.89
Left Shoulder	113.26	41.14	110.37	40.63
Left Elbow	124.47	43.23	120.51	43.25
Left Wrist	124.29	43.08	121.05	42.88
Torso	109.51	40.39	106.73	39.76
Head	107.42	41.06	104.46	40.21
Stomach	117.90	40.86	114.56	40.80
Average	114.27	41.20	111.48	40.72

Table 4.7 shows BPE and RPE measured from each camera view in Our Cath Lab. Camera 2 and Camera 4 have larger error than other views. We suspect that they are not calibrated very well.

Table 4.7: BPE and RPE from each camera view in Our Cath Lab

Our Cath Lab	Camera1	Camera2	Camera3	Camera4	Camera5
Linear:BPE(mm)	93.71	160.10	94.29	121.13	110.09
Linear:RPE(px)	30.77	52.61	33.46	45.62	52.39
L_2:BPE(mm)	103.19	168.35	108.35	132.55	82.05
L_2:RPE(px)	32.88	55.57	36.40	49.05	38.12

Table 4.8 shows the processing speed of two Triangulation methods on the same device. Linear method is much faster than L_2 method. Although L_2 method has slight better precision than Linear method in some scenarios, we choose Linear method as our pose reconstruction algorithm.

4.4.2 Appearance or geometry or 2D tracking in matching

As described in section 3.2, our approach combines 2D tracking results and geometry information to construct the affinity matrix. Here, we compare it with the alternatives using appearance or geometry alone and their combination. We test them on Operating

Table 4.8: The speed of different methods

Triangulation	Linear method	L_2 method
Speed	4.2fps	18.1fps

Room dataset. The result is shown in Table 4.4.2. There is no difference in the appearance of people in our scenario, so computing affinity based on appearance has worse performance than others. Our method did the best in MA. Good matching is the cornerstone of 3D reconstruction and 3D tracking.

Table 4.9: Comparison between different combinations of affinity

Matching	MOTA	BPE	MA
Appearance only	0.251	1527	0.154
Geometry only	0.893	102.4	0.860
Appearance+Geometry [11]	0.755	178.6	0.719
Ours	0.952	62.84	0.941

4.4.3 Geometry or 2D tracking in 3D tracking

Given the 3D poses in one frame, we use both pose geometry information and 2D tracking results to track targets between frames.We test different combinations of them with the same reconstruction result. Table 4.4.3 shows the results.

Table 4.10: Results of tracking

Tracking	MOTA
Geometry only	0.896
2D Tracking only	0.741
Ours	0.952

4.4.4 Top three views or all views

The traditional Linear Triangulation method uses all detections to reconstruct 3D poses, including the detection matched incorrectly. When someone is passing by another person, they will appear to overlap each other from some views. The overlap may cause a wrong match result. When we choose the wrong detection in construction, the result will pan from the truth. As described in section 3.2.2, we try to evaluate each detection with confidence to overcome this problem. Then, we only choose the top three views for reconstruction. In this way, we can avoid choosing wrong-matched detections because they usually have low confidence.

Table 4.11 shows the result of simulation under different MA. Choosing all views in reconstruction has better result when MA is very high. But it is very sensitive to the change of MA. When matching result is worse than before, the precision decrease rapidly.

Table 4.11: Comparison between Top three views and All views

MA	0.96	0.89	0.85
Top three:MOTA	0.952	0.901	0.870
Top three:BPE	62.84	93.60	107.3
All:MOTA	0.967	0.871	0.802
All:BPE	57.10	104.5	125.1

4.5 Qualitative evaluation

In this section, we will show some Qualitative results.

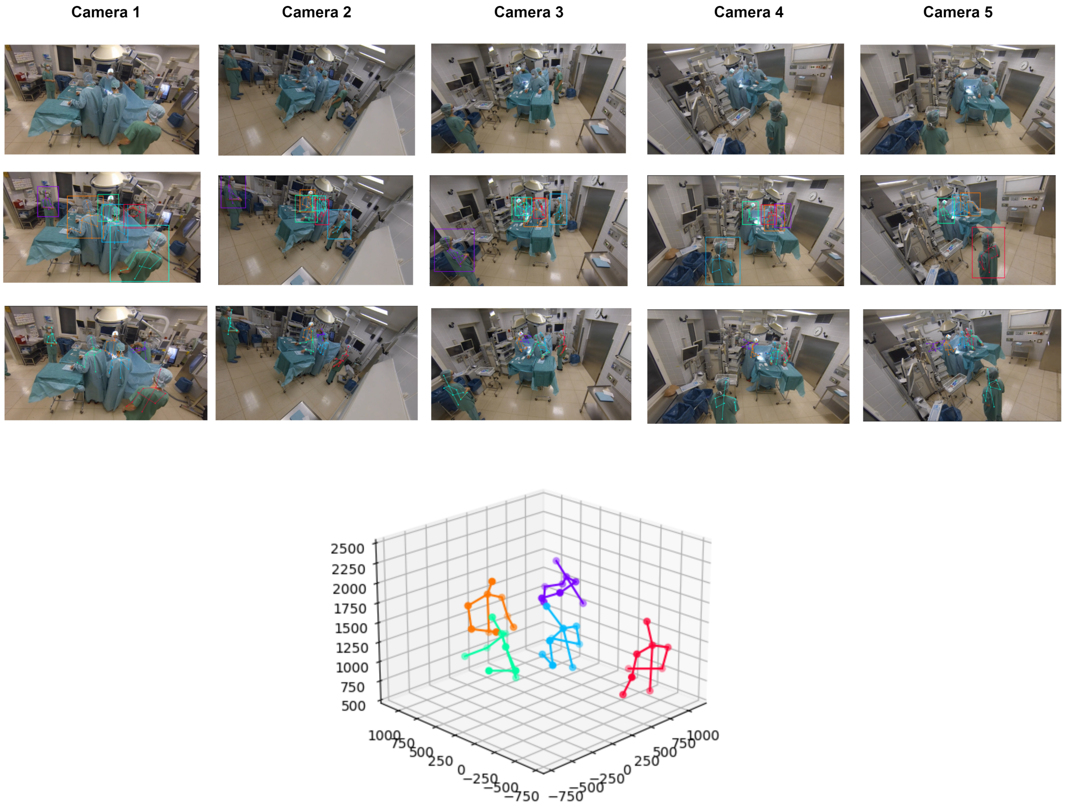


Figure 4.7: Qualitative results on the Operating Room

Figure 4.8 and Figure 4.7 show some representative results of our approach on the Operating Room and Our datasets. Given images from all views, our approach can reconstruct their 3D poses and track them. The first row shows the original input images from all views. The second shows the bounding boxes and 2D poses after 2D image detection. The third presents the reprojection 2D poses after 3D reconstruction. 3D poses in 3D world coordinates are below them. Colors present their identities.

Figure 4.9 shows the trajectory of one person in Operating Room between 1-10 frames. The red line present the trajectory of his stomach.

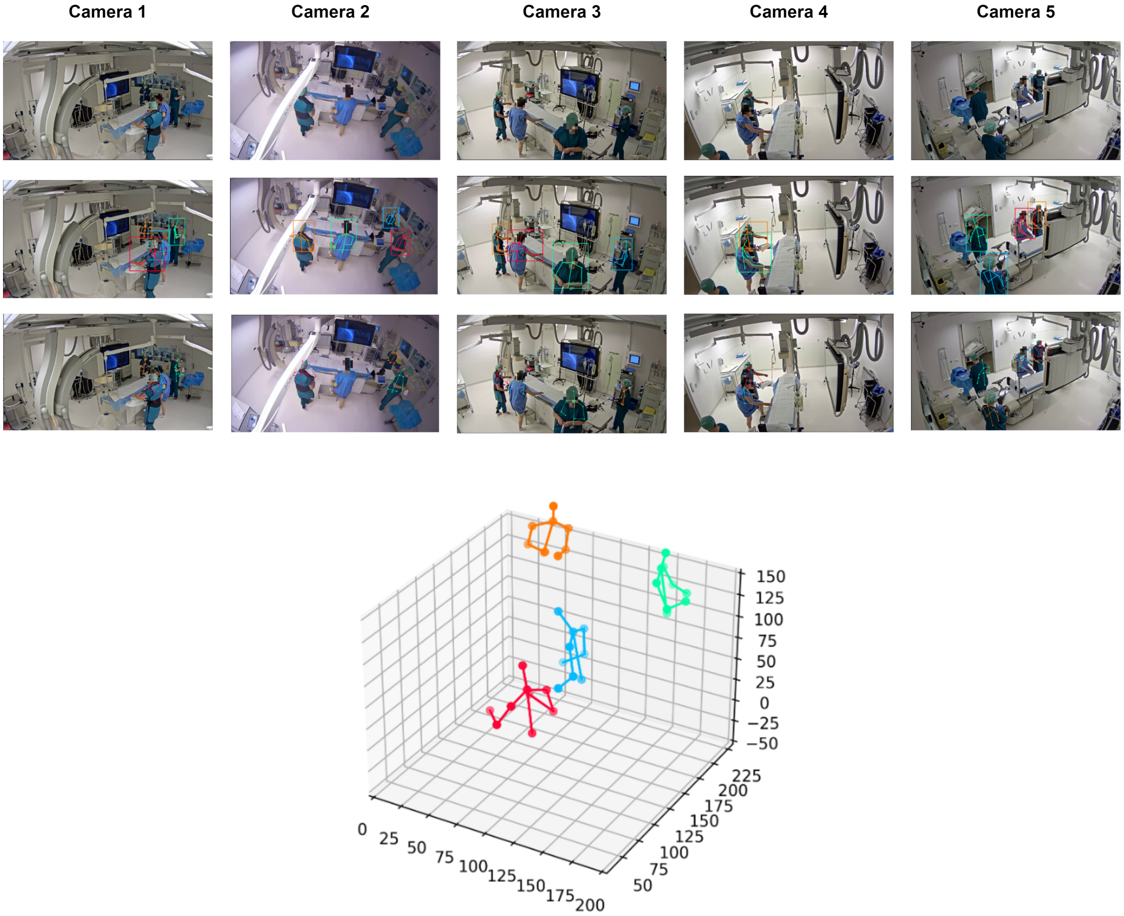


Figure 4.8: Qualitative results on Our dataset.

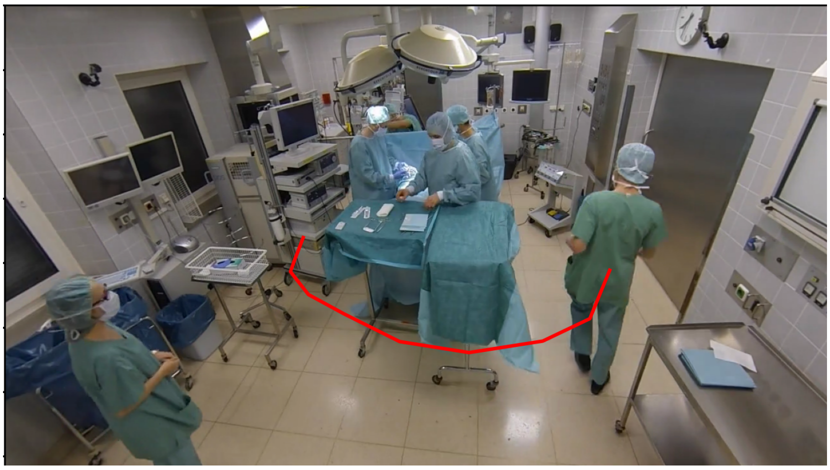


Figure 4.9: Trajectory of one person 1-10

Figure 4.10 shows the trajectory between 1-70 frames.

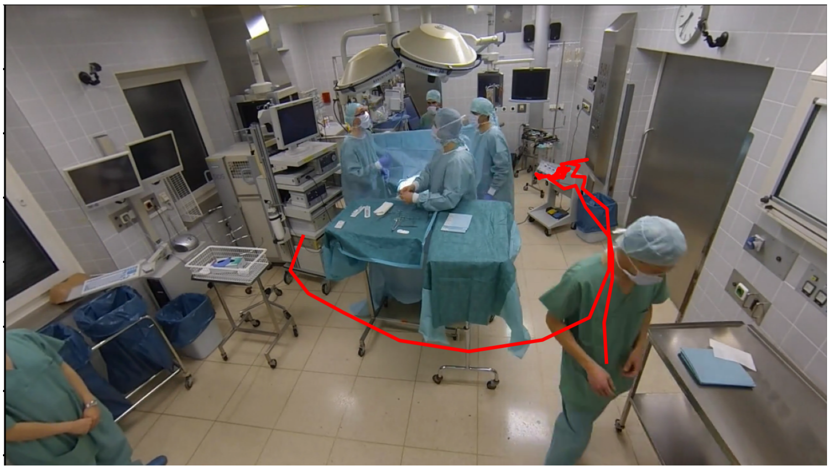


Figure 4.10: Trajectory of one person 1-70

Conclusion 5

5.1 Conclusion

In this work, we present a method for multi-human 3D pose estimation and tracking from multiple camera views in Cath Lab. Because we do not have 3D pose data, we choose a traditional 2D-3D pipeline combined with state-of-art modules. By introducing 2D tracking results into cross-view matching and 3D tracking, our method can reach high tracking accuracy in a scenario where there is no difference in appearance between people. Because our method does not need prior information during 3D reconstruction, it has high generalisability. In experiments, our solution achieves high accuracy and efficiency on the datasets.

5.2 Future Work

The method presented in this work attempts to find the solution under the condition that there is no 3D data. Even a little 3D pose data can bring an improvement in 3D pose reconstruction. We can have 3DPS from 3D pose data and reconstruct human poses with 3DPS. Because of the modular design, the Triangulation module can be replaced individually. If we have a mount of 3D pose data in Cath Lab in the future, the 3D-CNN neural network [33] will be a better alternative.

Besides, our method can be applied in 3D object tracking as well. In our method, the inputs of 3D modules are keypoints and bounding boxes, which can be from people or objects. If the 2D neural networks are trained on object datasets, the method will reconstruct object 3D keypoints and track them.

Bibliography

[1] Vasileios Belagiannis, Sikandar Amin, Mykhaylo Andriluka, Bernt Schiele, Nassir Navab, and Slobodan Ilic. 3d pictorial structures revisited: Multiple human pose estimation. *IEEE transactions on pattern analysis and machine intelligence*, 38(10):1929–1942, 2015.

[2] Vasileios Belagiannis, Xinchao Wang, Horesh Beny Shitrit, Kiyoshi Hashimoto, Ralf Stauder, Yoshimitsu Aoki, Michael Kranzfelder, Armin Schneider, Pascal Fua, Slobodan Ilic, Hubertus Feussner, and Nassir Navab. Parsing human skeletons in an operating room. *Mach. Vision Appl.*, 27(7):1035–1046, oct 2016.

[3] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011.

[4] Magnus Burenius, Josephine Sullivan, and Stefan Carlsson. 3d pictorial structures for multiple view articulated pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3618–3625, 2013.

[5] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, 2017.

[6] Jia Chen, Dongli Wu, Peng Song, Fuqin Deng, Ying He, and Shiyan Pang. Multi-view triangulation: Systematic comparison and an improved method. *IEEE Access*, 8:21017–21027, 2020.

[7] Kai Chen and Jiaqi Wang. MMDetection: Open mmlab detection toolbox and benchmark, 2019.

[8] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7103–7112, 2018.

[9] MMPose Contributors. Openmmlab pose estimation toolbox and benchmark. https://github.com/open-mmlab/mmpose, 2020.

[10] MMTracking Contributors. MMTracking: OpenMMLab video perception toolbox and benchmark. https://github.com/open-mmlab/mmtracking, 2020.

[11] Junting Dong, Wen Jiang, Qixing Huang, Hujun Bao, and Xiaowei Zhou. Fast and robust multi-person 3d pose estimation from multiple views, 2019.

[12] Sara Ershadi-Nasab, Erfan Noury, Shohreh Kasaei, and Esmaeil Sanaei. Multiple human 3d pose estimation from multiview images. *Multimedia Tools and Applications*, 77(12):15573–15601, 2018.

[13] Hao-Shu Fang, Shuqin Xie, Yu-Wing Tai, and Cewu Lu. RMPE: Regional multi-person pose estimation. In *ICCV*, 2017.

[14] Maryam Fazel. *Matrix rank minimization with applications.* PhD thesis, PhD thesis, Stanford University, 2002.

[15] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021, 2021.

[16] Richard I. Hartley and Peter Sturm. Triangulation. *Computer Vision and Image Understanding*, 68(2):146–157, 1997.

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[18] Qi-Xing Huang and Leonidas Guibas. Consistent shape maps via semidefinite programming. In *Computer graphics forum*, volume 32, pages 177–186. Wiley Online Library, 2013.

[19] Shaoli Huang, Mingming Gong, and Dacheng Tao. A coarse-fine network for keypoint localization. In *Proceedings of the IEEE international conference on computer vision*, pages 3028–3037, 2017.

[20] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014.

[21] Muhammed Kocabas, Salih Karagoz, and Emre Akbas. Multiposenet: Fast multi-person pose estimation using pose residual network. In *Proceedings of the European conference on computer vision (ECCV)*, pages 417–433, 2018.

[22] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[23] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2014. cite arxiv:1405.0312.

[24] K. Madsen, H. B. Nielsen, and O. Tingleff. Methods for non-linear least squares problems (2nd ed.), 2004.

[25] A. Milan, L. Leal-Taixé, I. Reid, S. Roth, and K. Schindler. MOT16: A benchmark for multi-object tracking. *arXiv:1603.00831 [cs]*, March 2016. arXiv: 1603.00831.

[26] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *European conference on computer vision*, pages 483–499. Springer, 2016.

[27] Leonid Pishchulin, Eldar Insafutdinov, Siyu Tang, Bjoern Andres, Mykhaylo Andriluka, Peter Gehler, and Bernt Schiele. Deepcut: Joint subset partition and labeling for multi person pose estimation, 2015.

[28] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks, 2015.

[29] Shuai Shao, Zijian Zhao, Boxun Li, Tete Xiao, Gang Yu, Xiangyu Zhang, and Jian Sun. Crowdhuman: A benchmark for detecting human in a crowd. 2018.

[30] Ke Sun, Yang Zhao, Borui Jiang, Tianheng Cheng, Bin Xiao, Dong Liu, Yadong Mu, Xinggang Wang, Wenyu Liu, and Jingdong Wang. High-resolution representations for labeling pixels and regions, 2019.

[31] Julian Tanke and Juergen Gall. Iterative greedy matching for 3d human pose tracking from multiple views. In Gernot A. Fink, Simone Frintrop, and Xiaoyi Jiang, editors, *Pattern Recognition*, pages 537–550, Cham, 2019. Springer International Publishing.

[32] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1653–1660, 2014.

[33] Hanyue Tu, Chunyu Wang, and Wenjun Zeng. Voxelpose: Towards multi-camera 3d human pose estimation in wild environment, 2020.

[34] Tao Wang, Jianfeng Zhang, Yujun Cai, Shuicheng Yan, and Jiashi Feng. Direct multi-view multi-person 3d pose estimation, 2021.

[35] Size Wu, Sheng Jin, Wentao Liu, Lei Bai, Chen Qian, Dong Liu, and Wanli Ouyang. Graph-based 3d multi-person pose estimation using multi-view images, 2021.

[36] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box, 2021.

[37] Zhun Zhong, Liang Zheng, Zhedong Zheng, Shaozi Li, and Yi Yang. Camera style adaptation for person re-identification, 2017.

[38] Xiaowei Zhou, Menglong Zhu, and Kostas Daniilidis. Multi-image matching via fast alternating minimization. In *Proceedings of the IEEE international conference on computer vision*, pages 4032–4040, 2015.