2D human pose tracking in the cardiac catheterisation laboratory with BYTE

Butler, Rick M.; Vijfvinkel, Teddy S.; Frassini, Emanuele; van Riel, Sjors; Bachvarov, Chavdar; Constandse, Jan; van der Elst, Maarten; van den Dobbelsteen, John J.; Hendriks, Benno H.W.

DOI
10.1016/j.medengphy.2024.104270

Publication date
2025

Document Version
Final published version

Published in
Medical Engineering and Physics

Medical Engineering and Physics 135 (2025) 104270

Contents lists available at ScienceDirect

Medical Engineering and Physics

journal homepage: www.elsevier.com/locate/medengphy

Paper

2D human pose tracking in the cardiac catheterisation laboratory with BYTE

Rick M. Butler [a,*], Teddy S. Vijfvinkel [a], Emanuele Frassini [a], Sjors van Riel [c], Chavdar Bachvarov [c], Jan Constandse [b], Maarten van der Elst [a,b], John J. van den Dobbelsteen [a], Benno H.W. Hendriks [a,c]

[a] Delft University of Technology, Delft, the Netherlands
[b] Reinier de Graaf Gasthuis, Delft, the Netherlands
[c] Philips Healthcare, Best, the Netherlands

ARTICLE INFO

Keywords:
Cardiac catheterisation laboratory
Computer vision
2D pose tracking
Workflow analysis

ABSTRACT

Workflow insights can enable safety- and efficiency improvements in the Cardiac Catheterisation Laboratory (Cath Lab). Human pose tracklets from video footage can provide a source of workflow information. However, occlusions and visual similarity between personnel make the Cath Lab a challenging environment for the re-identification of individuals. We propose a human pose tracker that addresses these problems specifically, and test it on recordings of real coronary angiograms. This tracker uses no visual information for re-identification, and instead employs object keypoint similarity between detections and predictions from a third-order motion model. Algorithm performance is measured on Cath Lab footage using Higher-Order Tracking Accuracy (HOTA). To evaluate its stability during procedures, this is done separately for five different surgical steps of the procedure. We achieve up to 0.71 HOTA where tested state-of-the-art pose trackers score up to 0.65 on the used dataset. We observe that the pose tracker HOTA performance varies with up to 10 percentage point (pp) between workflow phases, where tested state-of-the-art trackers show differences of up to 23 pp. In addition, the tracker achieves up to 22.5 frames per second, which is 9 frames per second faster than the current state-of-the-art on our setup in the Cath Lab. The fast and consistent short-term performance of the provided algorithm makes it suitable for use in workflow analysis in the Cath Lab and opens the door to real-time use-cases. Our code is publicly available at https://github.com/RM-8vt13r/PoseBYTE.

1. Introduction

The emerging field of workflow analysis promises tools for the analysis and improvement of surgical procedures [1–3]. Insights into workflow could be used to improve e.g. procedure efficiency and safety through personnel training. We investigate a tool for workflow analysis in the Cardiac Catheterisation Laboratory (Cath Lab): a specialised Operating Room (OR) for minimally invasive cardiovascular procedures. The Cath Lab is equipped for its specialised purpose with a fixed X-Ray imaging system containing a 'C-Arm' mount, a monitor, and a radiation shield.

One diagnostic procedure carried out in the Cath Lab is the coronary angiogram (CAG) [4]. During a CAG, cardiovascular access is established through the wrist or groin area using a catheter. A contrast fluid is administered directly into the coronary artery to detect anomalies on a captured X-Ray image. Reference [5] provides a description of the CAG in terms of consecutive workflow steps. Its well-defined nature makes the CAG suitable for explorative workflow study.

Manual workflow recognition is labour-intensive. In contrast, computer-assisted automation [6–8] is cost-effective, scalable, and enables real-time use-cases and assistance [9]. Multi-object keypoint detection can serve as a stepping stone to activity recognition [10–12]. 2D keypoint detectors—or pose estimators—localise predefined objects and their keypoints in continuous image pixel (px) space. They quantify detection confidence with a score per detected keypoint.

Multi-Object Tracking builds on detection by assigning the same identifier (ID) to the same object in different video frames. A tracker

* Corresponding author.
E-mail address: r.m.butler@tudelft.nl (R.M. Butler).

https://doi.org/10.1016/j.medengphy.2024.104270
Received 5 August 2024; Received in revised form 22 October 2024; Accepted 1 December 2024

Available online 5 December 2024

outputs a set of tracklets, each of which contains the per-frame detections of a unique object.

Many human pose tracking algorithms exist [13–16], which were benchmarked in general environments [17]. Several existing human pose trackers wrongfully swap identities or merge pose in the Cath Lab, as personnel occlusion and visual similarity are common.

In this paper we adapt BYTE [18]—a state-of-the-art bounding box tracker—for pose tracking in the Cath Lab. BYTE re-identifies objects or persons by comparing bounding box detections on subsequent frames. Persons pass each other regularly in the Cath Lab, after which their bounding boxes will be hard to distinguish from geometry alone. The visual features that BYTE uses to mitigate this problem are less effective here than in the general case, because everyone is dressed very similarly. Poses provide more geometric information that can be used for re-identification than a bounding box, by specifying keypoint coordinates. Therefore, we replace the use of bounding boxes in BYTE by human poses such that, after or during occlusion by a person or object, a person can be re-identified by posture. Additionally we extend the constant-velocity motion model that BYTE uses with acceleration and jerk to model more complex movement. These changes mitigate occlusion-induced problems like identity swaps or lost tracklets. As visual similarity between personnel can cause identity swaps, the tracker uses no image data. In the remaining text, we refer to the proposed method as 'PoseBYTE', indicating its utilisation of human pose data rather than bounding boxes for re-identification.

CAG workflow phases [5] differ in terms of appearance and movement. For instance, whilst the patient walks to the operating table there is a lot of movement and occlusion from ongoing preparations. During the intervention there are fewer people and less walking, and more subtle hand- and head motion. The lights being switched on or off during different phases causes visual differences. For accurate workflow analysis from poses, it is important that a pose tracker works throughout a procedure. Therefore, we test PoseBYTE separately during five different workflow phases.

Annotated video data are necessary to test pose trackers. We use 30-second video sequences of five CAG workflow phases from the Cath Lab of the Reinier de Graaf Gasthuis hospital, Delft, NL, all filmed from four viewpoints. Ground-truth human pose tracklets were annotated in the footage to evaluate metrics.

Section 2 starts with a description of our dataset, algorithm and experiments. Section 3 lists results and discusses those that stand out. These are further interpreted in section 4. Finally, section 5 gives a summary.

2. Methods

2.1. Dataset

The recording of CAG procedures in the Reinier de Graaf Gasthuis hospital, Delft, NL was approved by the Medical Ethics Committee Leiden The Hague Delft (protocol number Z19.057, 30-10-2019) and the hospital board. Informed consent was collected from all filmed patients and staff. Procedures were recorded in the hospital Cath Lab from four different viewpoints (Axis M1125) in a resolution of 1920 px × 1080 px and framerate of 25 frames per second (fps). A cardiologist, scrub nurse, up to two lab assistants, and the patient were present during each procedure.

We annotate poses in ten procedure recordings, each performed by a different medical team for variability. 51 frames were sampled uniformly over 30 seconds per procedure, from each of the four synchronised viewpoints. This gives a total of $10 \times 51 \times 4 = 2040$ annotated frames. The video sequences were hand-selected to show five different workflow phases, each taken from two different procedures:

- the patient entering and lying down,
- realisation of endovascular access,
- Use of ultrasound to detect the radial artery for endovascular access,
- X-Ray imaging, and
- closure of the entrywound.

The annotations were made in Computer Vision Annotation Tool (CVAT) [19][1] by two authors with an engineering background. Annotation quality was checked by another author who has been a practicing interventional cardiologist for over 13 years. Occluded persons and reflections in the monitor or windows were not labelled.

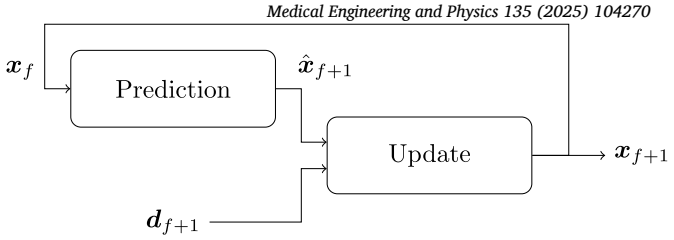

Fig. 1. Schematic of a Kalman filter [20]. A state $\boldsymbol{x}_f$ is kept internally. Given a noisy measurement $\boldsymbol{d}_{f+1}$ and a model prediction $\hat{\boldsymbol{x}}_{f+1}$ based on $\hat{\boldsymbol{x}}_f$, a new state $\boldsymbol{x}_{f+1}$ is produced. $\boldsymbol{x}_{f+1}$ contains a denoised measurement and estimated hidden variables that are used in the prediction model.

2.2. Pose detection

2D human poses are detected per frame with a keypoint detector and serve as input to PoseBYTE. Section 2.4.5 discusses the tested detectors.

2.3. PoseBYTE

2.3.1. BYTE

BYTE [18] is a state-of-the-art tracking algorithm that re-identifies bounding boxes produced by an object detector [21] between frames. It keeps a set of tracklets, each of which keeps a state column vector with its position $\boldsymbol{p}_f$ [px] and velocity $\boldsymbol{v}_f$ [px f^{-1}] (pixels per frame)

$$\boldsymbol{x}_f = \begin{bmatrix} \boldsymbol{p}_f \\ \boldsymbol{v}_f \end{bmatrix} \tag{1}$$

per frame f. Here, the unit f is the time duration of a single frame. On each frame, a Kalman filter [20] produces a model-based prediction $\hat{\boldsymbol{p}}_{f+1} = \boldsymbol{p}_f + \boldsymbol{v}_f$ as visualised in Fig. 1. This is called the prediction step.

Bounding box detection confidences are classified as high or low with a threshold γ_{high}. A similarity metric, e.g., Intersection over Union (IoU) [22] or a visual re-identification feature, measures resemblance between high-confidence detections and tracklet predictions $\hat{\boldsymbol{p}}_{f+1}$. Similarity scores above threshold σ_{high} are used in the Hungarian algorithm [23] to match detections to predictions. The Kalman filter provides an updated state $\boldsymbol{x}_{f+1}$ from each match, shown in the right part of Fig. 1. This is called the update step.

Low-confidence detections are matched to remaining tracklets with a similarity metric that does not rely on visual information. Here, another similarity score threshold $\sigma_{\text{low}} < \sigma_{\text{high}}$ is applied. Unmatched tracklets are labelled as 'lost', their new state being predicted on each frame until i) they can be matched to a detection and the tracklet continues, or ii) f_{mem} frames have passed and the tracklet ends. Remaining high-confidence boxes seed new tracklets, which are confirmed on the next frame with a similarity threshold $\sigma_{\text{new}} < \sigma_{\text{high}}$ before proceeding as usual.

2.3.2. Pose tracking

We adapt the Kalman filter to store coordinates and velocities per keypoint rather than per object. Position and velocity in (1) become

[1] Code available: https://github.com/cvat-ai/cvat.

$$\boldsymbol{p}_f = \begin{bmatrix} x_f^1 \\ y_f^1 \\ \vdots \\ x_f^{|\mathcal{K}|} \\ y_f^{|\mathcal{K}|} \end{bmatrix}, \boldsymbol{v}_f = \begin{bmatrix} vx_f^1 \\ vy_f^1 \\ \vdots \\ vx_f^{|\mathcal{K}|} \\ vy_f^{|\mathcal{K}|} \end{bmatrix}, \tag{2}$$

where $\mathcal{K}$ and $|\square|$ denote the set of keypoint classes and set cardinality operator, and x_f^k, y_f^k, vx_f^k, vy_f^k are horizontal (x) and vertical (y) position and speed of keypoint $k \in \mathcal{K}$ on frame f.

If and only if i) a tracklet and a pose detection are matched, and ii) a keypoint in the pose has a confidence below γ_{kp}, we exclude this keypoint from the update step. This is done by leaving out the rows corresponding to this keypoint in the Kalman filter observation matrix and observation vector during the update. Thus, in this case, the state of this keypoint on the next frame is purely its model-based prediction. If a keypoint has a confidence below γ_{kp} when starting a new tracklet, we apply a large 10 000 px observation uncertainty to it instead as we need to initialise a full initial state. When a tracklet is not matched or no keypoints remain after thresholding, the tracklet is lost but can be found again as described in section 2.3.1. Whilst a tracklet is lost, predicted keypoints act for future matching only and are not saved as part of the tracklet. Tracklet keypoint coordinates are taken from the Kalman filter update step, and confidences copied from the detector.

We use the mean of all nonzero keypoint confidences as pose score, and the tightest-fit bounding box as approximate segmentation area. Object Keypoint Similarity (OKS) [24] is used as Similarity score, in which calculation the Kalman filter prediction is treated as ground truth. We do not use any visual clues, as similarities between personnel can make this an unreliable feature for pose tracking in the Cath Lab.

2.3.3. Higher-order movement

BYTE uses a constant-velocity model for state prediction. In human movement we can suspect higher-order positional derivatives to be involved [25]. Therefore, we add acceleration $\boldsymbol{a}_f$ and jerk $\boldsymbol{j}_f$ to the model. The state vector becomes

$$\boldsymbol{x}_f = \begin{bmatrix} \boldsymbol{p}_f \\ \boldsymbol{v}_f \\ \boldsymbol{a}_f \\ \boldsymbol{j}_f \end{bmatrix}, \boldsymbol{a}_f = \begin{bmatrix} ax_f^1 \\ ay_f^1 \\ \vdots \\ ax_f^{|\mathcal{K}|} \\ ay_f^{|\mathcal{K}|} \end{bmatrix}, \boldsymbol{j}_f = \begin{bmatrix} jx_f^1 \\ jy_f^1 \\ \vdots \\ jx_f^{|\mathcal{K}|} \\ jy_f^{|\mathcal{K}|} \end{bmatrix}, \tag{3}$$

where $\boldsymbol{p}_f$ and $\boldsymbol{v}_f$ are given by (2) and ax_f^k, ay_f^k, jx_f^k, jy_f^k are acceleration and jerk. In the prediction step we assume that these decrease linearly over time and update them as

$$\begin{bmatrix} \hat{\boldsymbol{a}}_{f+1} \\ \hat{\boldsymbol{j}}_{f+1} \end{bmatrix} = \left(\begin{bmatrix} \alpha_a & \alpha_j \\ 0 & \alpha_j \end{bmatrix} \otimes I_{2|\mathcal{K}|} \right) \begin{bmatrix} \boldsymbol{a}_f \\ \boldsymbol{j}_f \end{bmatrix}, \tag{4}$$

where α_a and α_j are memory factors, $\otimes$ denotes the Kronecker product, and $I_x \in \mathbb{R}^{x \times x}$ is an identity matrix. Next we predict velocity and position with the 3rd-order derivative motion equations

$$\begin{bmatrix} \hat{\boldsymbol{p}}_{f+1} \\ \hat{\boldsymbol{v}}_{f+1} \end{bmatrix} = \left(\begin{bmatrix} 1 & 1 & 1/2 & 1/6 \\ 0 & 1 & 1 & 1/2 \end{bmatrix} \otimes I_{2|\mathcal{K}|} \right) \begin{bmatrix} \hat{\boldsymbol{p}}_f \\ \hat{\boldsymbol{v}}_f \\ \hat{\boldsymbol{a}}_{f+1} \\ \hat{\boldsymbol{j}}_{f+1} \end{bmatrix}. \tag{5}$$

We predict in these two steps to ensure that $\boldsymbol{a}_f$ and $\boldsymbol{j}_f$ have no effect on $\boldsymbol{x}_{f+1}$ if the respective memory factor is 0. Note that, if $\alpha_a = 0$ but $\alpha_j \neq 0$, jerk still causes some acceleration in (4).

2.4. Experimental setup

2.4.1. Metrics

Detection- and tracking performance are evaluated with Detection Accuracy (DA) and Association Accuracy (AA) [26]. Higher-Order Tracking Accuracy (HOTA) $= \sqrt{\text{DA} \cdot \text{AA}}$ aggregates these metrics into a single score. We match detections to annotations as described in [26] with OKS as localisation similarity. DA, AA and HOTA are evaluated over a range of OKS thresholds from 0.5 to 0.95 with step size 0.05, and report the average as per convention [24,26]. Additionally, we measure average algorithm speed in [fps].

Table 1
PoseBYTE parameters used for all experiments in section 2.4.

γ_{high}	γ_{kp}	σ_{high}	σ_{low}	σ_{new}	α_a	α_j	f_{mem}
0.5	0.3	0.8	0.5	0.65	0.4	0.8	50

Table 2
Pose detection models tested in section 2.4.5. Here, OpenPifPaf30T is the only model with built-in tracking.

Model	Details
AlphaPose50	ResNet50 [27]+YOLOv3-SPP [28,29]+FastPose [13]
AlphaPose152	ResNet152 [27]+YOLOv3-SPP [28,29]+FastPose [13]
OpenPifPaf16C	ShuffleNetV2K16 [14,30]+CifCaf [14]
OpenPifPaf30C	ShuffleNetV2K30 [14,30]+CifCaf [14]
OpenPifPaf30T	tShuffleNetV2K30 [14,30]+TrackingPose [14]

2.4.2. Workflow phase

Metrics are evaluated separately on each annotated workflow phase from section 2.1. This way we observe situational effects on pose tracking.

2.4.3. Parameters

We use the PoseBYTE parameters from Table 1. Optimal values for α_a and α_j are found by ranging each from 0 to 0.9 and evaluating HOTA for all workflow phases jointly. We exclude memory factors of 1 to prevent instability in the Kalman prediction step.

2.4.4. Ablation study

We test the contribution of each PoseBYTE component on HOTA and speed. As a baseline we test bounding box tracking using IoU as object similarity. Here, we use tightest-fit bounding boxes around each pose for tracking but still evaluate metrics on keypoints for consistency. Undetected keypoints are estimated by translating and scaling the last detected pose to tightly fit the new bounding box after each prediction step. Secondly, we add pose data and OKS in the Kalman filter as described in section 2.3.2. Finally, we include the acceleration and jerk from section 2.3.3.

2.4.5. Pose detector

Table 2 introduces all tested pose detectors and their abbreviations in this paper. All tests are carried out with AlphaPose152 unless explicitly stated otherwise. No detectors are re-trained, i.e., the pre-trained models and code linked in the respective citations from Table 2 are used. As the optimal values for α_a and α_j rely heavily on the pose detector, we select these to maximise HOTA separately for each detector. Other parameters are kept the same in accordance to section 2.4.3. We provide baseline tracking results from AlphaPose152+Human-ReID [13] and OpenPifPaf30T.

2.4.6. Qualitative results

For demonstrative purposes we show example pose tracklets from AlphaPose152 with Human-ReID or PoseBYTE in the 'Patient entry' phase. Frames are selected to highlight problems solved or introduced by PoseBYTE. We only show keypoints with a detection confidence of at least γ_{kp}.

Table 3
DA for different acceleration- and jerk memory factors, evaluated jointly over all workflow phases.

α_a \ α_j	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
0.0	.64	.64	.64	.64	.64	.64	.64	.64	.65	.64
0.1	.64	.64	.64	.64	.64	.64	.64	.64	.65	.64
0.2	.64	.64	.64	.64	.64	.64	.64	.64	.64	.63
0.3	.64	.64	.64	.64	.64	.64	.64	.64	.64	.63
0.4	.64	.64	.64	.64	.64	.64	.64	.64	.64	.63
0.5	.64	.64	.64	.64	.64	.64	.64	.64	.64	.63
0.6	.64	.64	.64	.64	.64	.64	.64	.64	.64	.63
0.7	.64	.64	.64	.64	.64	.64	.64	.64	.63	.63
0.8	.64	.64	.64	.64	.64	.64	.64	.64	.63	.63
0.9	.64	.64	.64	.64	.64	.64	.64	.63	.63	.63

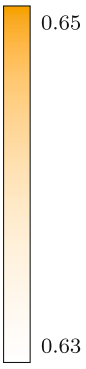

Table 4
AA for different acceleration- and jerk memory factors, evaluated jointly over all workflow phases.

α_a \ α_j	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
0.0	.78	.78	.78	.77	.77	.78	.78	.78	.77	.77
0.1	.78	.77	.77	.77	.77	.78	.78	.77	.76	.76
0.2	.77	.78	.77	.77	.78	.78	.78	.77	.76	.75
0.3	.77	.77	.77	.77	.78	.78	.77	.77	.76	.76
0.4	.78	.78	.78	.77	.78	.78	.77	.77	.77	.76
0.5	.77	.77	.78	.78	.78	.78	.77	.76	.77	.76
0.6	.78	.78	.78	.78	.78	.78	.77	.76	.77	.75
0.7	.78	.78	.78	.77	.78	.77	.76	.76	.75	.75
0.8	.77	.77	.78	.77	.76	.76	.77	.77	.75	.75
0.9	.77	.77	.76	.76	.77	.77	.76	.76	.75	.72

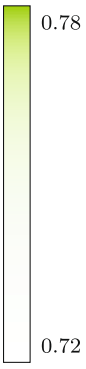

Table 5
PoseBYTE HOTA and speed in fps per workflow phase with the parameters from Table 1 after each addition from section 2.3.

BYTE	OKS	α_a	α_j	Patient entry		Wrist access		Ultrasound		X-Ray		Wound closure		Total	
				HOTA	fps	HOTA	fps	HOTA	fps	HOTA	fps	HOTA	fps	HOTA	fps
✓				0.74	21.7	0.69	27.5	0.75	18.6	0.66	20.8	0.61	24.4	0.70	22.2
✓	✓			0.73	21.8	0.71	27.8	0.75	18.1	0.68	22.8	0.64	25.3	0.71	22.7
✓	✓	0.7		0.73	21.5	0.69	27.5	0.75	18.2	0.68	22.9	0.64	24.4	0.70	22.5
✓	✓	0.2	0.5	0.73	21.3	0.69	26.7	0.75	18.0	0.68	23.2	0.65	25.2	0.71	22.5

Table 6
HOTA and speed in fps of PoseBYTE and other trackers.

Model	Patient entry		Wrist access		Ultrasound		X-Ray		Wound closure		Total	
	HOTA	fps	HOTA	fps	HOTA	fps	HOTA	fps	HOTA	fps	HOTA	fps
AlphaPose50+Human-ReID	0.55	12.4	0.69	15.8	0.74	11.0	0.73	14.4	0.52	16.1	0.65	13.7
AlphaPose50+PoseBYTE(ours)	0.74	17.1	0.68	25.8	0.72	14.8	0.70	20.3	0.65	20.6	0.70	19.1
AlphaPose152+Human-ReID	0.55	12.2	0.70	15.4	0.76	11.0	0.73	14.2	0.53	16.0	0.65	13.5
AlphaPose152+PoseBYTE(ours)	0.73	21.3	0.69	26.7	0.75	18.0	0.68	23.2	0.65	25.2	0.71	22.5
OpenPifPaf16C+PoseBYTE(ours)	0.71	9.4	0.54	9.8	0.62	8.8	0.65	9.7	0.64	9.8	0.64	9.5
OpenPifPaf30T	0.60	4.2	0.47	4.3	0.53	4.2	0.54	4.3	0.51	4.3	0.53	4.3
OpenPifPaf30C+PoseBYTE(ours)	0.77	5.0	0.68	5.1	0.74	4.8	0.75	5.1	0.69	5.1	0.73	5.0
OpenPose+PoseBYTE(ours)	0.71	10.1	0.60	10.1	0.75	9.9	0.74	9.8	0.62	10.2	0.69	10.0

3. Results

Table 3 shows DA for a range of memory factors. Scores range from 0.63 to 0.65, where a low acceleration factor seems to be preferred. Table 4 shows AA in a similar fashion. Here, scores range from 0.72 to 0.78 and are mostly uniform around 0.78 when α_a and α_j are lower than 0.8. Consequently, HOTA ranges from 0.67 to 0.71 with a preference for low α_a and α_j.

OpenPose achieves up to 0.69 HOTA with α_a and α_j below 0.6, and achieves the best AA when $\alpha_a^2 + \alpha_j^2 \approx 0.55^2$. OpenPifPaf16C prefers low α_a and α_j, and scores up to 0.64 HOTA. OpenPifPaf30C yields up to 0.73 HOTA, when $\alpha_a^2 + \alpha_j^2 \leq 0.55^2$ holds. Finally, AlphaPose50 scores up to 0.70 HOTA and prefers both factors between 0.4 and 0.9.

The optimal memory factors differ per workflow phase. During 'Patient entry', keeping both memory factors below 0.8 approaches a HOTA of 0.74. During 'Wrist access' making both factors 0 yields the best HOTA of 0.70. The 'Ultrasound' phase prefers factors of $\alpha_a^2 + \alpha_j^2 \approx 0.6^2$ for a HOTA of 0.75. 'X-Ray' yields 0.71 HOTA for $\alpha_j \approx 0.9 - 0.4\alpha_a$. During 'Wound closure' the best DA of 0.56 is achieved for $\alpha_a^2 + \alpha_j^2 \approx 0.8$, and the best AA of 0.76 for $\alpha_a^2 + \alpha_j^2 \approx 0.55$.

Table 5 shows HOTA and inference speed per added PoseBYTE component. Tracking poses instead of bounding boxes increases HOTA by 0 pp to 3 pp depending on the phase. An exception is the 'Patient entry' phase, on which HOTA decreases by 1 pp. Adding α_a keeps results mostly the same, where HOTA decreases by 2 pp during 'Wrist access'. The addition of α_j increases HOTA by 1 pp on the 'Wound closure' phase. Speed changes per added component seem negligible, where the largest observed change on all phases jointly is 0.5 fps. We observe speed differences per workflow phase, where 'Ultrasound' yields the lowest speeds of 18.0 fps to 18.6 fps and 'Wrist access' the highest of 26.7 fps to 27.8 fps.

For OpenPifPaf16C, which achieves a HOTA score of 0.44 with BYTE, the addition of OKS yields a HOTA gain of 20 pp. OpenPifPaf30C sees a similar increase from 0.58 to 0.73. OpenPose gains 5 pp with OKS over 0.64 HOTA with BYTE.

We show results for all considered pose detectors and trackers in Table 6. The best HOTA of 0.73 is achieved by OpenPifPaf30C+ PoseBYTE,

with a speed of 5.0 fps. It is closely followed with 0.71 HOTA by AlphaPose152+ PoseBYTE—the fastest model at 22.5 fps. AlphaPose50+ PoseBYTE comes close with 0.70 HOTA at 19.1 fps. The lowest HOTA and speed come from OpenPifPaf30T: 0.53 at 4.3 fps

AlphaPose and OpenPifPaf achieve higher HOTA and speed with PoseBYTE than with their own trackers—Human-ReID and TrackingPose. For AlphaPose the HOTA improvement is up to 6 pp whereas for OpenPifPaf it is 20 pp. Looking per phase, Human-ReID outperforms PoseBYTE by up to 5 pp during 'Wrist access', 'Ultrasound' and 'X-Ray'. During 'Patient entry', PoseBYTE outperforms Human-ReID with up to 19 pp. OpenPose yields worse HOTA and speed than AlphaPose152 with PoseBYTE on all workflow phases except 'Ultrasound' and 'X-Ray'. For OpenPifPaf16C there are no such exceptions. AlphaPose152+ PoseBYTE yields HOTA differences per phase of up to 10 pp, which is 23 pp with Human-ReID. This difference is present but less pronounced for OpenPifPaf30 with 9 pp and 13 pp.

Fig. 2 shows qualitative results of Human-ReID and PoseBYTE during 'Patient entry' with the AlphaPose152 detector. Each row shows a different frame in chronological order, where the top row comes first and bottom row last. Time intervals between rows are not constant. Each pose shows an integer tracking ID and the detection confidence score between 0 and 1.

At the start of the procedure, Human-ReID sees all persons earlier than PoseBYTE. Shortly after, PoseBYTE catches up and sees the same persons. Between the second and third timesteps, one assistant passes in front of the patient and another walks behind the infusion bags. Here, an identity swap occurs with Human-ReID between the patient and the first assistant, but not with PoseBYTE. Both trackers lose the second assistant, after which Human-ReID wrongly assigns a previously seen ID and PoseBYTE assigns a new one. Only Human-ReID sees the third assistant in the lower-left corner. In the fourth row, Human-ReID reassigned the first two assistants their initial IDs. A duplicate pose can be seen in the patient, which is now assigned both their original ID and that of the third assistant in the corner. PoseBYTE is still tracking the two assistants, but has assigned a new ID to the patient after an assistant passed them in front. In the last row, neither Human-ReID nor PoseBYTE has lost or swapped any IDs. Here, Human-ReID sees a pose in the reflection of the monitor, which PoseBYTE ignores because of the tracklet confirmation step inherent to BYTE.

4. Discussion

In this work we adapted BYTE for pose tracking in the Cath Lab and compared the resulting method to pose trackers from literature.

During the 'Patient entry' phase, Human-ReID and TrackingPose yield HOTA scores of up to 0.60. We observe in videos that Human-ReID is prone to identity swaps, which could be due to it relying on visual clues of similarly-dressed personnel. It occasionally detects duplicate poses, which slip past the non-maximum suppression designed to solve this very problem [13]. TrackingPose sometimes merges poses that are close to each other, possibly because of its multi-frame pose construction creating more opportunity to do so. It also tends to miss visible keypoints in partially occluded poses, which Human-ReID solves possibly by imposing a prior through bounding box detection. The many (4 to 6) visible persons during 'Patient entry' could amplify these issues. PoseBYTE uses no visual features and does tracking and detection separately, which could contribute to it performing up to 18 pp HOTA better on this phase. However, this tracker does tend to lose persons quickly during occlusions of more than a few frames.

With the 'Ultrasound' phase containing 4 to 5 people, one could expect the same problems to occur. Although TrackingPose performs similarly here, Human-ReID does better with up to 0.76 HOTA—1 pp higher than PoseBYTE. A difference between this phase and 'Patient entry' is that, although people occlude each other in both, they walk a lot during 'Patient entry' and stay in place during 'Ultrasound'. Their close vicinity causes TrackingPose the same problems as before, whilst their stillness could be allowing Human-ReID to track more accurately based on position. The same is visible in the other low-movement phases 'Wrist access' and 'X-Ray', which both have only 2 to 3 persons in the room beside the—rarely visible—patient, simplifying the tracking problem.

Modelling acceleration and jerk yielded little HOTA improvement for any tested detector. It yielded its largest HOTA improvement of 5 pp when using the AlphaPose50 detector in the 'Wound closure' phase. Different combinations of detector and phase yield different optimal memory factors. All in all, the benefit of including higher-order movement seems negligible.

We aim to provide a tracker that works reliably throughout a procedure. Even though PoseBYTE does not always perform better than the benchmark set by the state-of-the-art, its HOTA varies much less between workflow phases. For workflow analysis the most important phases to analyse through poses are 'Patient entry' and 'Wound closure', as during other phases the system logs provide an alternative source of workflow information. During these phases, PoseBYTE delivers a HOTA improvement of 12 pp to 19 pp with respect to the benchmark.

PoseBYTE speed roughly decreases with the number of people in the room when the AlphaPose detector is used. This effect is much less pronounced, if at all, with the OpenPifPaf and OpenPose detectors. The same can be observed with Human-ReID and TrackingPose, suggesting that the slowdown occurs in the detection- and not the tracking stage. In either case, PoseBYTE achieves higher speeds than the benchmark trackers in all situations, making real-time applications more viable.

For the purpose of workflow analysis, it is important to have a reliable information source throughout a procedure. PoseBYTE fits this description well, as it achieved the lowest HOTA spread of all tested trackers over the tested CAG workflow phases. Especially during 'Patient entry', where few other sources of workflow information are available, PoseBYTE improves on the state-of-the-art. Whether its overall HOTA score of 0.71 is high enough will depend on what information one wants to obtain. It will suffice for measuring estimate positions and short-term motion of people in the Cath Lab, which can already be indicative of workflow. However, the results might not be good enough for analysis of fine-grained long-term movement and gestures. Here, the tendency of PoseBYTE to lose tracklets after occlusion could interfere. One can mitigate the effect of inaccurate motion model predictions by excluding keypoints with a confidence below γ_{kp}.

We did not test for optimal values of PoseBYTE parameters other than α_a and α_j, and even those latter two were tested only over a limited set of values. For reference, a memory factor of 0.9 per frame amounts to a memory of only $0.9^{25} = 0.072$ per second. The used movement model assumes keypoints to move independently of each other, causing anatomically unrealistic predictions over time. This could explain why PoseBYTE still has trouble re-identifying poses after occlusions of some frames.

In future work more memory factors in the range $[0.9, 1)$ could be tested, in addition to finding optimal values for other parameters. A memory factor for velocity could be included, as we observe movements in the Cath Lab to often span short distances. Alternatively a different model could be used, built specifically for human motion prediction [31]. Finally, the integration of multiple camera views could be investigated as in [32–34].

PoseBYTE yields higher HOTA and speed in the Cath Lab with greater stability between different situations than the tested pose trackers from literature. With a HOTA score of 0.71 at 22.5 fps, it is a suitable method for short-term real-time pose tracking for workflow analysis in the Cath Lab.

5. Conclusion

We adapted BYTE for pose tracking in the Cath Lab without relying on visual clues. The algorithm was evaluated in terms of HOTA and speed on five annotated CAG workflow phases before, during, and after procedures. PoseBYTE shows stable performance across workflow

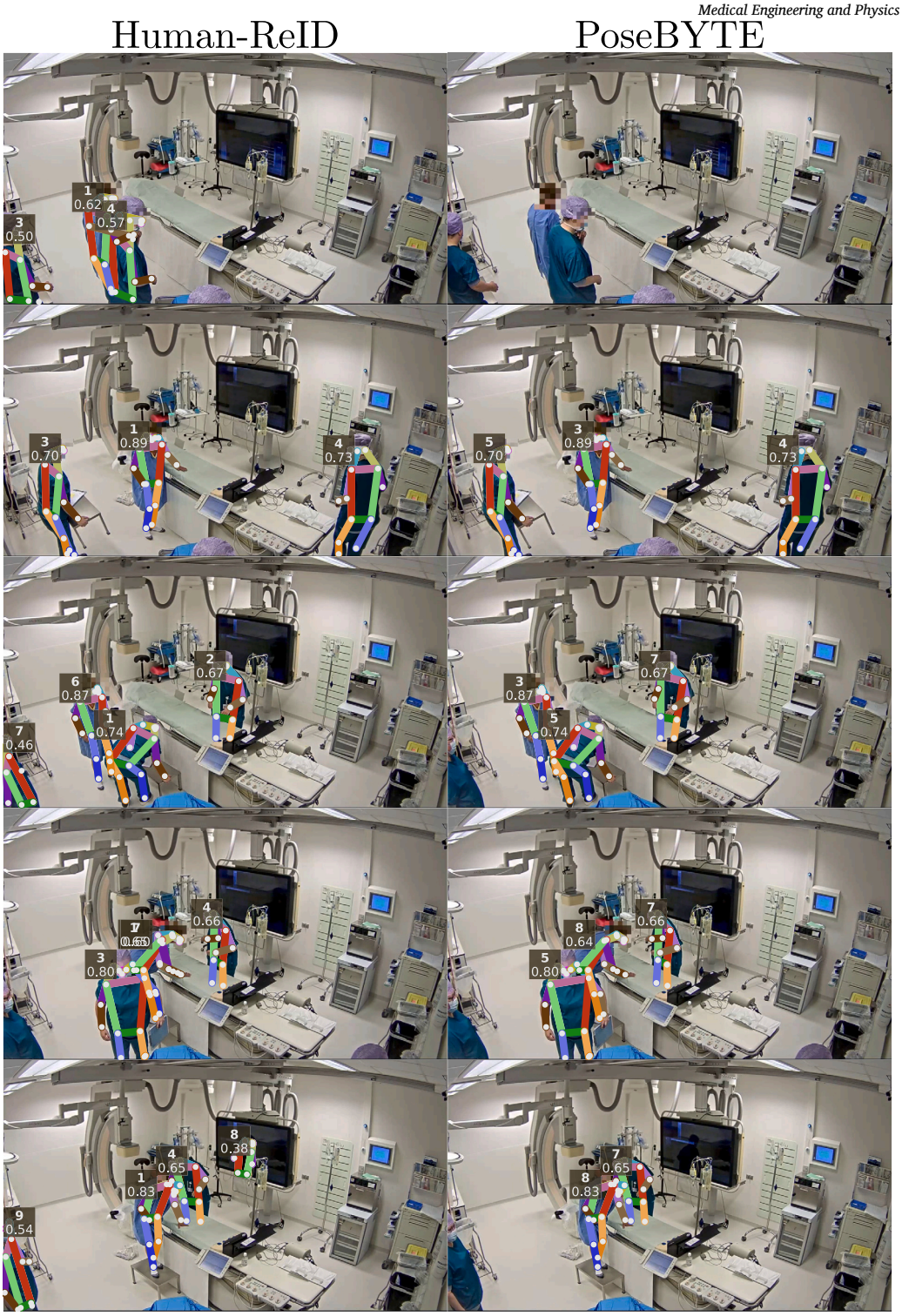

Fig. 2. Qualitative results of Human-ReID and PoseBYTE during the 'Patient entry' phase, where poses are detected with AlphaPose152. Rows show different timeframes in chronological order from top to bottom, with varying intervals.

phases and outperforms the current state of the art in terms of HOTA and speed. The improvement is most apparent when the patient enters the room, which is also the least trivial situation for tracking.

Declaration of competing interest

The authors who are affiliated with Philips (S. v R., C.B., B.H.W.H.) have financial interests in the subject matter, materials, and equipment, in the sense that they are employees of Philips. None of the other authors have any financial relationship or competing interests.

Acknowledgements

This research was funded by Rijksdienst voor Ondernemend Nederland, grant number AI212005, and was sponsored in part by Philips Healthcare, study protocol NL71861.058.19.

The study was conducted in accordance with the Declaration of Helsinki, and approved by the Medical Ethics Committee Leiden The Hague Delft (protocol code Z19.057, 30-10-2019) and by the board of the hospital where it was conducted. Informed consent was obtained from all subjects involved in the study.

Data availability

PoseBYTE was implemented by modifying the public source code of BYTE in Python, which is available at https://github.com/ifzhang/ByteTrack.

References

[1] Timoh KN, Huaulme A, Cleary K, Zaheer MA, Lavoué V, Donoho D, et al. A systematic review of annotation for surgical process model analysis in minimally invasive surgery based on video. Surg Endosc 2023;37:4298–314. https://doi.org/10.1007/s00464-023-10041-w.

[2] Schouten AM, Flipse SM, van Nieuwenhuizen KE, Jansen FW, van der Eijk AC, van den Dobbelsteen JJ. Operating room performance optimization metrics: a systematic review. J Med Syst 2023;47:19. https://doi.org/10.1007/s10916-023-01912-9.

[3] Lalys F, Jannin P. Surgical process modelling: a review. Int J Comput Assisted Radiol Surg 2014;9:495–511. https://doi.org/10.1007/s11548-013-0940-5.

[4] Mayo Clinic Staff. Coronary angiogram. https://www.mayoclinic.org/tests-procedures/coronary-angiogram/about/pac-20384904, 12 2021.

[5] Reed GW, Hantz S, Cunningham R, Krishnaswamy A, Ellis SG, Khot U, et al. Operational efficiency and productivity improvement initiatives in a large cardiac catheterization laboratory. JACC: Cardiovasc Interv 2018;11(4):329–38. https://doi.org/10.1016/j.jcin.2017.09.025.

[6] Garrow CR, Kowalewski K-F, Li L, Wagner M, Schmidt MW, Engelhardt S, et al. Machine learning for surgical phase recognition: a systematic review. Ann Surg 2021;273(4):684–93. https://doi.org/10.1097/SLA.0000000000004425.

[7] Berlet M, Vogel T, Ostler D, Czempiel T, Kähler M, Brunner S, et al. Surgical reporting for laparoscopic cholecystectomy based on phase annotation by a convolutional neural network (CNN) and the phenomenon of phase flickering: a proof of concept. Int J Comput Assisted Radiol Surg 2022;17:1991–9. https://doi.org/10.1007/s11548-022-02680-6.

[8] Aksamentov I, Twinanda AP, Mutter D, Marescaux J, Padoy N. Deep neural networks predict remaining surgery duration from cholecystectomy videos. In: Med. image comput. comput.-assist. interv. - MICCAI 2017. Cham, Switzerland: Springer; 2017. p. 586–93.

[9] Maier-Hein L, Vedula SS, Speidel S, Navab N, Kikinis R, Park A, et al. Surgical data science for next-generation interventions. Nat Biomed Eng 2017;1:691–6. https://doi.org/10.1038/s41551-017-0132-7.

[10] Saleem G, Bajwa UI, Raza RH. Toward human activity recognition: a survey. Neural Comput Appl 2023;35:4145–82. https://doi.org/10.1007/s00521-022-07937-4.

[11] Nguyen H-C, Nguyen T-H, Scherer R, Le V-H. Deep learning for human activity recognition on 3D human skeleton: survey and comparative study. Sens 2023;23(11):5121. https://doi.org/10.3390/s23115121.

[12] Wang C, Yan J. A comprehensive survey of RGB-based and skeleton-based human action recognition. IEEE Access 2023;11:53880–98. https://doi.org/10.1109/ACCESS.2023.3282311.

[13] Fang H-S, Li J, Tang H, Xu C, Zhu H, Xiu Y, et al. AlphaPose: whole-body regional multi-person pose estimation and tracking in real-time. IEEE Trans Pattern Anal Mach Intell 2023;45(6):7157–73. https://doi.org/10.1109/TPAMI.2022.3222784.

[14] Kreiss S, Bertoni L, Alahi A. OpenPifPaf: composite fields for semantic keypoint detection and spatio-temporal association. IEEE Trans Intell Transp Syst 2022;23(8):13498–511. https://doi.org/10.1109/TITS.2021.3124981.

[15] Wang M, Tighe J, Modolo D. Combining detection and tracking for human pose estimation in videos. In: 2020 IEEE/CVF conf. comput. vis. pattern recognit. (CVPR). New York, USA: IEEE; 2020. p. 11085–93.

[16] Wang Z, Zhao H, Li Y-L, Wang S, Torr PH, Bertinetto L. Do different tracking tasks require different appearance models? In: Adv. neural inf. process. syst. 34 (NeurIPS 2021). New York, USA: Curran Associates, Inc.; 2021. p. 726–38.

[17] Andriluka M, Iqbal U, Insafutdinov E, Pishchulin L, Milan A, Gall J, et al. Posetrack: a benchmark for human pose estimation and tracking. In: 2018 IEEE/CVF conf. comput. vis. pattern recognit. (CVPR). New York, USA: IEEE; 2018. p. 5167–76.

[18] Zhang Y, Sun P, Jiang Y, Yu D, Weng F, Yuan Z, et al. ByteTrack: multi-object tracking by associating every detection box. In: Eur. conf. on comput. vis. Cham, Switzerland: Springer; 2022. p. 1–21.

[19] CVAT.ai Corporation. Computer vision annotation tool (CVAT). https://www.cvat.ai, 06 2023.

[20] Kalman RE. A new approach to linear filtering and prediction problems. J Basic Eng 1960;82(1):35–45. https://doi.org/10.1115/1.3662552.

[21] Zou Z, Chen K, Shi Z, Guo Y, Ye J. Object detection in 20 years: a survey. Proc IEEE 2023;111(3):257–76. https://doi.org/10.1109/JPROC.2023.3238524.

[22] Padilla R, Passos WL, Dias TLB, Netto SL, da Silva EAB. A comparative analysis of object detection metrics with a companion open-source toolkit. Electron 2021;10(3):279. https://doi.org/10.3390/electronics10030279.

[23] Kuhn HW. Variants of the Hungarian method for assignment problems. Nav Res Logist Q 1956;03(04):253–8. https://doi.org/10.1002/nav.3800030404.

[24] Lin T-Y, Patterson G, Ronchi MR, Cui Y, Maire M, Belongie S, et al. Common objects in context (COCO). https://cocodataset.org, 08 2021.

[25] Sers R, Forrester S, Zecca M, Ward S, Moss E. Objective assessment of surgeon kinematics during simulated laparoscopic surgery: a preliminary evaluation of the effect of high body mass index models. Int J Comput Assisted Radiol Surg 2022;17(1):75–83. https://doi.org/10.1007/s11548-021-02455-5.

[26] Luiten J, Ošep A, Dendorfer P, Torr P, Geiger A, Leal-Taixé L, et al. HOTA: a higher order metric for evaluating multi-object tracking. Int J Comput Vis 2021;129(2):548–78. https://doi.org/10.1007/s11263-020-01375-2.

[27] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proc. 29th IEEE conf. comput. vis. pattern recognit. New York, USA: IEEE; 2016. p. 770–8.

[28] Redmon J, Farhadi A. YOLOv3: an incremental improvement. arXiv:1804.02767v1. https://doi.org/10.48550/arXiv.1804.02767, 04 2018.

[29] He K, Zhang X, Ren S, Sun J. Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans Pattern Anal Mach Intell 2015;37(9):1904–16. https://doi.org/10.1109/TPAMI.2015.2389824.

[30] Ma N, Zhang X, Zheng H-T, Sun J. ShuffleNet V2: practical guidelines for efficient CNN architecture design. In: Proc. 15th eur. conf. comput. vis. Cham, Switzerland: Springer; 2018. p. 122–38.

[31] Martinez J, Black MJ, Romero J. On human motion prediction using recurrent neural networks. In: 2017 IEEE conference on computer vision and pattern recognition (CVPR). New York, USA: IEEE; 2017. p. 4674–83.

[32] Kadkhodamohammadi A, Gangi A, de Mathelin M, Padoy N. Articulated clinician detection using 3D pictorial structures on RGB-D data. Med Image Anal 2017;35:215–24. https://doi.org/10.1016/j.media.2016.07.001.

[33] Kadkhodamohammadi A, Gangi A, de Mathelin M, Padoy N. A multi-view RGB-D approach for human pose estimation in operating rooms. In: 2017 IEEE winter conf. appl. comput. vis. (WACV). New York, USA: IEEE; 2017. p. 363–72.

[34] Kadkhodamohammadi A, Padoy N. A generalizable approach for multi-view 3D human pose regression. Mach Vis Appl 2021;32:6. https://doi.org/10.1007/s00138-020-01120-2.