

Multi-person Physics-based Pose Estimation for Combat Sports

Hossein Feiz¹ David Labb  ¹ Sheldon Andrews¹

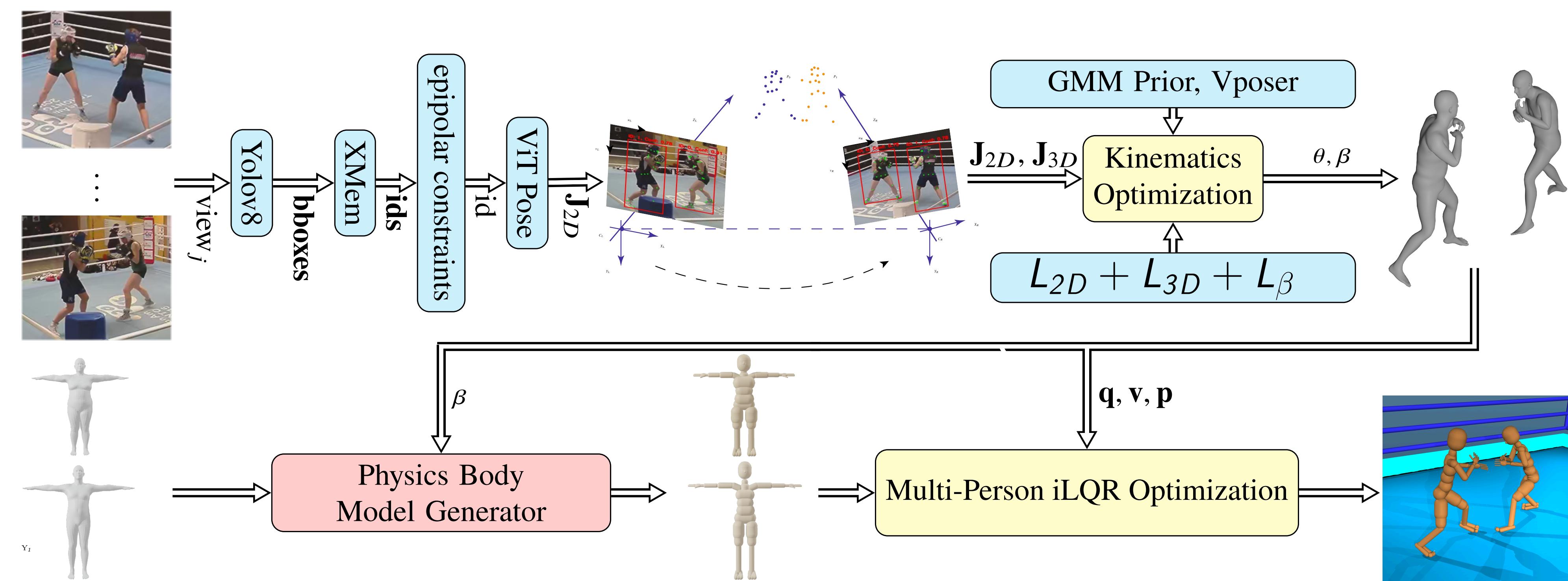
¹  cole de technologie sup  rieure, Montr  al, Qu  bec, Canada



Introduction

Motion capture from only RGB cameras in a typical sports scene (e.g., boxing), with the presence of coaches, viewers, and close interactions between athletes, brings many challenges, such as heavy occlusion, background crowding, and fast, complicated movements. To address this problematic, a multi-view configuration was used. Our system only required the visibility of an athlete from two cameras. We also created a pipeline using machine and deep learning techniques that automated the process of creating animations.

Pipeline



Multi-frame Multi-view Tracking IDs

- 1st item on the right
- 2nd item on the right
- 3rd item on the right

Weighted Triangulation and Filtering

The weighted triangulation estimates the 3D positions of the keypoints from their corresponding 2D keypoints from all N cameras. The 3D positions of each joint are thus determined by solving the linear system:

Weighted Triangulation

$$\begin{bmatrix} \mu_1(\mathbf{P}_{11} - u_1\mathbf{P}_{31}) \\ \mu_1(\mathbf{P}_{21} - v_1\mathbf{P}_{31}) \\ \vdots \\ \mu_N(\mathbf{P}_{1N} - u_N\mathbf{P}_{3N}) \\ \mu_N(\mathbf{P}_{2N} - v_N\mathbf{P}_{3N}) \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix},$$

\mathbf{P}_{ij} extracts the i -th row of the projection matrix \mathbf{P}_j . μ_j is the average confidence of a 2D point for the current frame in camera j . Higher confidence values give more weight to certain cameras. We use SVD to solve Eq. 1.

After triangulation, outliers are handled by interpolating and smoothing using cubic spline for each joint's trajectory. If there is no solution for triangulation, we utilize an extended Kalman filter for dynamic state estimation based on velocity, acceleration of the keypoint, and position constraint to estimate the 3D keypoints. This results in robust filtering and smoothing while preserving trajectory integrity, enabling dependable 3D reconstructions from sparse and noisy 2D keypoints.

Kinematics Optimization

Our kinematics optimization refines the SMPL model to minimize the difference between the provided 2D poses (\mathbf{J}_{2D}) from multiple views and 3D keypoints (\mathbf{J}_{3D}) obtained from triangulation. The process ensures temporal coherency and natural motion by incorporating smoothness and human motion priors. An LBFGS optimizer is used to solve the minimization problem, considering various loss terms, including 2D re-projection, 3D alignment, smoothness, and prior losses.

Kinematics Optimization Formulas

$$\min_{\theta} w_1 L_{2D} + w_2 L_{3D} + w_3 L_{reg} + w_4 L_{smooth} + w_5 L_{GMM} + w_6 L_{Vposer}. \quad (2)$$

$$L_{2D} = \sum_{j \in \mathcal{V}} \sum_{i \in \mathcal{J}_{2D}} c_{j,i} \rho(J_{proj_{j,i}} - J_{2D,i}), \quad (3) \quad L_{smooth} = \sum_t \|\theta^t - \theta^{t-1}\|^2 + \|\mathcal{M}(\theta^t, \beta) - \mathcal{M}(\theta^{t-1}, \beta)\|^2. \quad (5)$$

$$L_{3D} = \sum_{i \in \mathcal{J}_{3D}} c_i \|J_i(\theta, \beta) - J_{3D,i}\|^2, \quad (4) \quad L_{GMM} = \frac{1}{N} \sum_{i=1}^N GMM(\theta_i, \beta), \quad L_{Vposer} = \frac{1}{N} \sum_{i=1}^N (z(\theta_i))^2. \quad (6)$$

Physics Body Model Generator

Since our goal is to reconstruct the motion of K physics-based humanoids at each time step t , we define the combined state of all humanoids as $\mathbf{x}_t = (\mathbf{q}_t, \mathbf{v}_t)$, where $\mathbf{q}_t \in \mathbb{R}^{63K}$ and $\mathbf{v}_t \in \mathbb{R}^{62K}$ represent the rotation and velocity of the joints. Whereas \mathbf{q}_t is a minimal coordinate representation of position-level coordinates, we also compute $\mathbf{p}_t \in \mathbb{R}^{21K}$ which are the 3D position of the SMPL joints relative to the humanoid's body frame of reference. This vector is used for the optimization process. Note that the root orientation is represented as a quaternion. Further details about the humanoid body model can be found in the supplementary material.

Multi-person Dynamics Optimization

The iLQR algorithm optimizes a control trajectory $\mathbf{u}_{0:T}$ by minimizing the cost function:

$$\min_{\mathbf{u}_{0:T}} \sum_{t=0}^T w_1 L_{reg,t} + w_2 L_{p,t} + w_3 L_{v,t} + w_4 L_{collision,t}. \quad (7)$$

The reference positions \mathbf{p}_t and velocities \mathbf{v}_t are extracted from the kinematics optimization at each frame t , and the trajectory optimization is “warm started” using control parameters \mathbf{u}_{t-1} from the previous frame.

The control is updated iteratively as $\Delta \mathbf{u}_t = \mathbf{K}_t \Delta \mathbf{x}_t + \alpha \mathbf{k}_t$, with \mathbf{K}_t and \mathbf{k}_t refined until convergence.

Block 1

Content for the first block.

Block 2