Appendix 3: Computation of True Axial Rotation

Computation of True Axial Rotation

In this appendix, the orientation of an object will be denoted using ${}^{T}\mathbf{R}_{H}(t_k)$ - indicating, in this instance, the orientation of the (H)umerus with respect to the (T)orso at time t_k. Angular velocity of the humerus expressed in the coordinate system of the torso, at time t_k, will be designated by ${}^{T}\omega(t_k)$. The longitudinal axis of the humerus expressed in the torso coordinate system, at time t_k will be denoted by ${}^{T}\mathbf{l}(t_k)$.

Miyazaki [1] defines true axial rotation at time t_k, $\theta(t_k)$, as the integration of the projection of the angular velocity vector onto the longitudinal axis of the humerus:

$$\theta(t_k) = \int_0^{t_k} {}^{T}\boldsymbol{\omega} \cdot {}^{T}\mathbf{l}\, dt \tag{1}$$

Both the angular velocity and the longitudinal axis of the humerus are expressed in the torso coordinate system in Equation (1), although this is not strictly necessary. They could both be expressed in the scapular coordinate system, although care must be taken that **both** quantities are expressed in the same coordinate system so the dot product between the two is valid.

Equation (1) assumes a continuous time domain, but in motion capture recordings the time domain is discretized according to the capture frequency. In practice, true axial rotation can be computed via the trapezoidal rule. Let $trapz([y(0), ..., y(t_k)], \Delta t)$ compute the integral of the equally spaced sequence of numbers in $[y(0), ..., y(t_k)]$ and let Δt denote the spacing between each number in the sequence. Then, true axial rotation can be computed numerically via:

$$\theta(t_k) = trapz([{}^{T}\omega(t_0) \cdot {}^{T}\mathbf{l}(t_0), ..., {}^{T}\omega(t_k) \cdot {}^{T}\mathbf{l}(t_k)], \Delta t) \tag{2}$$

Equation (2) implies that $\theta(t_0) = 0$.

True axial rotation can also be computed from the finite helical axes representation. Let the rotation of the humerus from time t_k to time t_{k+1} – ${}^{T}\mathbf{R}_{H}(t_{k+1}) \cdot ({}^{T}\mathbf{R}_{H}(t_k))^{T}$ – be represented by a rotation about the unit vector ${}^{T}\hat{\mathbf{n}}(t_k)$ by an angle $a(t_k)$. Scaling ${}^{T}\hat{\mathbf{n}}(t_k)$ by $a(t_k)$, we obtain ${}^{T}\boldsymbol{\alpha}(t_k) = {}^{T}\hat{\mathbf{n}}(t_k) * a(t_k)$. Because the time between motion capture frames is typically small (e.g. 10 ms for 100 Hz), $a(t_k)$ is typically very small allowing us to treat ${}^{T}\boldsymbol{\alpha}(t_k)$ as an infinitesimal rotation. Infinitesimal rotations form a vector space [2], which permits us the ability to project ${}^{T}\boldsymbol{\alpha}(t_k)$ onto another axis of rotation using the dot product. Note that without the infinitesimal rotation assumption, the helical axes of rotation cannot be projected onto another axis of rotation by utilizing the dot product because 3D rotations do not form a vector space. The infinitesimal rotation assumption allows us to re-write true axial rotation in terms of the finite helical axes representation:

$$\begin{aligned} &\theta(t_0) = 0 \\ &\theta(t_k) = \sum_{i=0}^{k-1} {}^{T}\boldsymbol{\alpha}(t_i) \cdot {}^{T}\mathbf{l}(t_i) \ \text{ for } k \neq 0 \end{aligned} \tag{3}$$

Our analysis demonstrated that for all trials and all timepoints the difference in true axial rotation as computed via angular velocity versus finite helical axes was at most 0.18°.

Appendix 3: Computation of True Axial Rotation

Given the high capture frequency of most motion capture studies (>100 Hz), the infinitesimal rotation assumption should be well-justified. Out of an abundance of caution, however, we computed true axial rotation without the infinitesimal rotation assumption. This technique relies on the Swing-Twist method [3], which decomposes a rotation into two rotations whose axes of rotation are orthogonal to each other. The physical interpretation of the Swing-Twist and Swing-Spin [4, 5] (referenced in the main manuscript) methods are identical, however the Swing-Twist method is much more computationally efficient. Let $\mathbf{R}(\hat{\boldsymbol{\nu}}, \psi)$ represent a rotation about axis $\hat{\boldsymbol{\nu}}$ by ψ. Then, given a desired axis of rotation $\hat{\boldsymbol{\kappa}}$, the Swing-Twist method decomposes $\mathbf{R}(\hat{\boldsymbol{\nu}}, \psi)$ as:

$$\mathbf{R}(\hat{\boldsymbol{\nu}}, \psi) = \mathbf{R}(\hat{\boldsymbol{\kappa}}, \beta) \cdot \mathbf{R}(\hat{\boldsymbol{\eta}}, \tau) \tag{4}$$

In this decomposition, $\hat{\boldsymbol{\kappa}} \cdot \hat{\boldsymbol{\eta}} = 0$ and $|\tau|$ is minimal. This decomposition is analogous to projecting a vector $\psi\hat{\boldsymbol{\nu}}$ onto a unit vector $\hat{\boldsymbol{\kappa}}$. For all trials and timeframes, there was no difference (to within 0.01 deg) between computing true axial rotation using finite helical axes and the Swing-Twist decomposition - indicating that the infinitesimal rotation assumption is well-justified. An implementation of each method of computing true axial rotation – and their comparison – can be found in the code repository associated with this manuscript.

[1] S. Miyazaki, A. Ishida, New mathematical definition and calculation of axial rotation of anatomical joints, Journal of biomechanical engineering 113(3) (1991) 270-5.

[2] F. Hamano, Derivative of rotation matrix direct matrix derivation of well known formula, arXiv preprint arXiv:1311.6010 (2013).

[3] P. Dobrowolski, Swing-twist decomposition in Clifford algebra, arXiv preprint arXiv:1506.05481 (2015).

[4] A. Campeau-Lecours, D.-S. Vu, F. Schweitzer, J.-S. Roy, Alternative Representation of the Shoulder Orientation Based on the Tilt-and-Torsion Angles, Journal of biomechanical engineering 142(7) (2020). https://doi.org/10.1115/1.4046203.

[5] P.L. Cheng, A.C. Nicol, J.P. Paul, Determination of axial rotation angles of limb segments — a new method, Journal of biomechanics 33(7) (2000) 837-843. http://www.sciencedirect.com/science/article/pii/S0021929000000324.