Poor Posture Detection Algorithm

Step1. Landmarks

Extract 2D coordinates of the key landmarks:

$$S_L = (x_{SL}, y_{SL}), \quad S_R = (x_{SR}, y_{SR}), \quad E_L = (x_{EL}, y_{EL}),$$

and define the shoulder midpoint

$$M = \tfrac{1}{2}(S_L + S_R).$$

Step2. Angle Definition

For any three points A, B, C, define the angle at B as

$$\angle ABC = \cos^{-1}\left(\frac{(A-B)\cdot(C-B)}{\|A-B\|\,\|C-B\|}\right).$$

Step3. Posture Angles

$$\theta_{\text{sh}} = \angle\big(S_L,\ S_R,\ (x_{SR}, 0)\big), \qquad \theta_{\text{neck}} = \angle\big(E_L,\ S_L,\ (x_{SL}, 0)\big).$$

Step4. Calibration

During the first N frames, compute mean angles and subtract a margin δ (e.g. 10°):

$$\tau_{\text{sh}} = \overline{\theta_{\text{sh}}} - \delta, \qquad \tau_{\text{neck}} = \overline{\theta_{\text{neck}}} - \delta.$$

Step5. Decision Rule

$$\text{Poor posture} \iff \theta_{\text{sh}} < \tau_{\text{sh}} \ \ \vee \ \ \theta_{\text{neck}} < \tau_{\text{neck}}.$$

Step6. Stability Check

Estimate midpoint speed:

$$v = \frac{\sum_i \|M_i - M_{i-1}\|}{t_k - t_0}.$$

If $v < v_{\text{still}}$, trust the angle classification; otherwise ignore (movement or standing up).