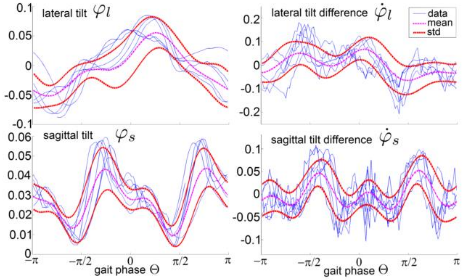
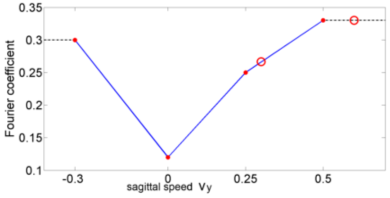
Literature Review on Fall Detection

- **Instability Detection and Fall Avoidance for a Humanoid using Attitude Sensors and Reflexes (Renner 2006)**





$$\mu(v_u) = \frac{1}{\sum_{i=1}^{N}\rho_i}\sum_{i=1}^{N}\rho_i s_i$$

$$\sigma(v_u) = \sqrt{\frac{N}{N-1}\frac{1}{\sum_{i=1}^{N}\rho_i}\sum_{i=1}^{N}\rho_i(s_i-\tilde{\mu}(v_u))^2}$$

$$\rho_i = 1-\frac{|v-v_u|}{v_k-v_j},$$

- **Offline model based abnormality evaluation approach**

- **For a database of support gaits, the periodic trajectories of the robot roll and pitch are recorded. A Fourier transform is conducted to compute the Fourier coefficients of the low frequency components.**

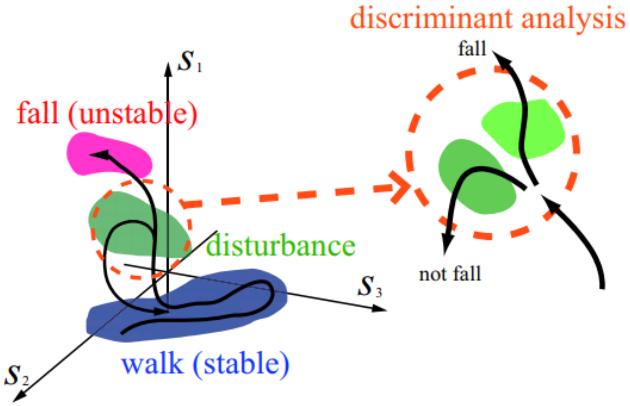
- **Then for another walking gait, a linear interpolation is conducted to get the Fourier coefficient. The corresponding mean and standard deviation is calculated to provide the normalized standard deviation.**

$$\tau_{l/s}^t = 1-\varsigma_{a,b}(|\delta_{l/s}^t|) = 1-\frac{1}{1+exp\left(\frac{|\delta_{l/s}^t|-a}{b}\right)},$$

Literature Review on Fall Detection

- Falling Motion Control for Humanoid Robots While Walking (Ogata 2007)

- Two fall detection approach have been proposed.

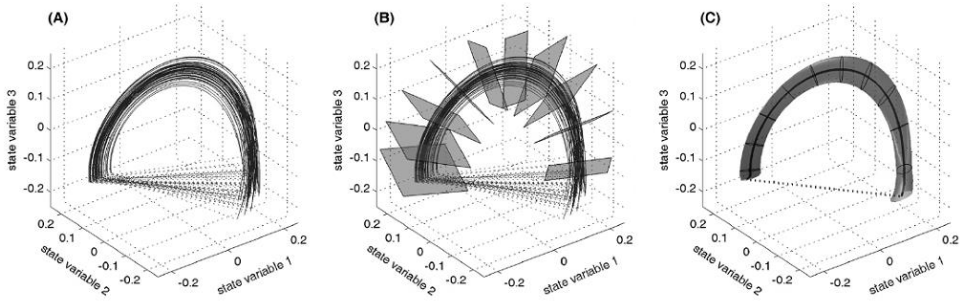


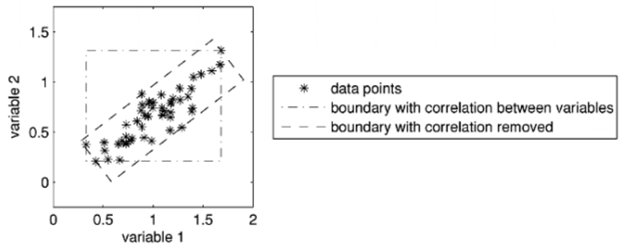
- Method 1: Abnormality detection method with heuristic threshold
- For certain "Normal" gait, a statistic average gait is calculate from many gait data. During its period, taking an evenly distributed N data as references. Then compare the actual robot state at the matching time to these references.

- Method 2:
- An categorization of the disturbed state is studied. There are two kinds of disturbed states: unstabilizable or stabilizable. Then a discriminant analysis is conducted to learn the data in order to classify the type of the unstable states.

Literature Review on Fall Detection

- **Fall detection in walking robots by multi-way principal component analysis (Karssen 2008)**





- **Offline model based abnormality evaluation approach**
- **For *I* points on (*J*-1)-dimensional plane, select *K* slices. On each slice, for all *I* points, data points on each dimension are analyzed through the PCA approach.**

$$\bar{\mathbf{X}}_j^k = \frac{1}{I}\sum_{i=1}^{I}\mathbf{X}_{ij}^k \qquad \text{for } j = 1, \ldots, (J-1), \quad \sigma_j^k = \frac{1}{I-1}\sum_{i=1}^{I}\mathbf{T}_{ij}^k \qquad \text{for } j = 1, \ldots, (J-1).$$

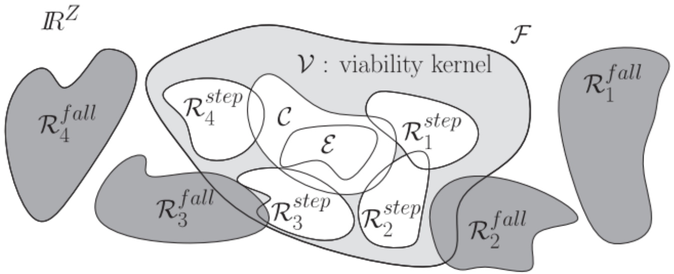
- **Then normal distributions for each dimension are computed.**

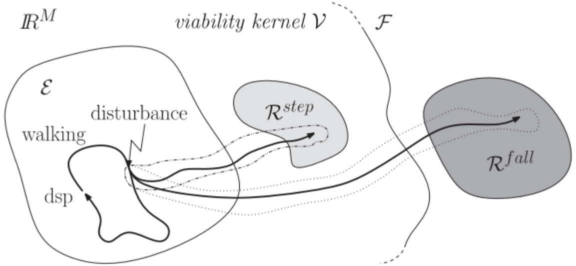
$$e_j = x_j - \bar{\mathbf{X}}_j^k \qquad \text{for } j = 1, \ldots, (J-1), \qquad \check{e}_j = \frac{\tilde{e}_j}{\sigma_j^k} \qquad \text{for } j = 1, \ldots, (J-1). \qquad D = \frac{1}{J-1}\sum_{j=1}^{J-1}\check{e}_j{}^2.$$

- **Cost is defined with the D-statistic for each slice. A heuristic value is then chosen for stability evaluation.**

Literature Review on Fall Detection

- **Probabilistic Balance Monitoring for Bipedal Robots (Hohn 2009)**





- **Offline model based probability evaluation method**

- **Instead of a comparison using the robot state vector, the method proposes a sensor-based feature measurement for comparison. Features contain force sensors, robot roll/pitch angle and angular velocities, MicroIMU at feet.**
- **GMM: single time sample comparison**

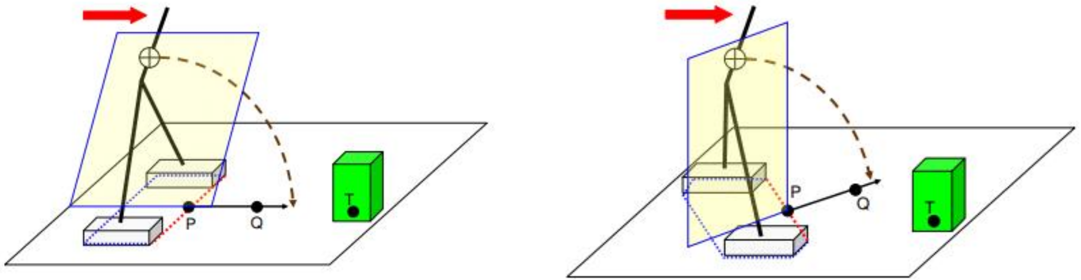
$$P(\boldsymbol{x}|\Omega_k) = \sum_{i=1}^{C_k} w_{ki}\,\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_{ki}, \Sigma_{ki})$$

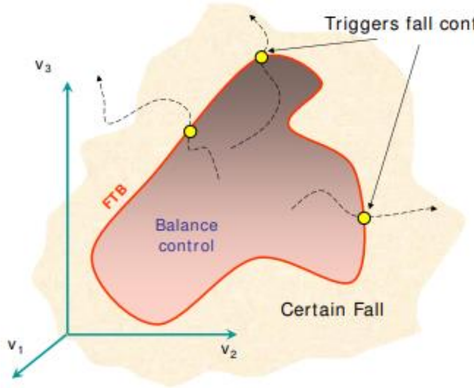
$$\Omega^* = \underset{\Omega_k}{\operatorname{argmax}}\, P(\Omega_k|\boldsymbol{x}). \qquad P(\Omega_k|\boldsymbol{x}) = \frac{P(\boldsymbol{x}|\Omega_k)P(\Omega_k)}{P(\boldsymbol{x})}. \qquad \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_k, \Sigma_k) = \frac{1}{\sqrt{(2\pi)^M|\Sigma_k|}} e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_k)^T\Sigma_k^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_k)}.$$

- **HMM: sample duration sequence comparison – kernel transition sequence to determine the set of the final robot state**

• Literature Review on Fall Detection

- **Safe Fall: Humanoid robot fall direction change through intelligent stepping and inertia shaping (Yun 2009)**





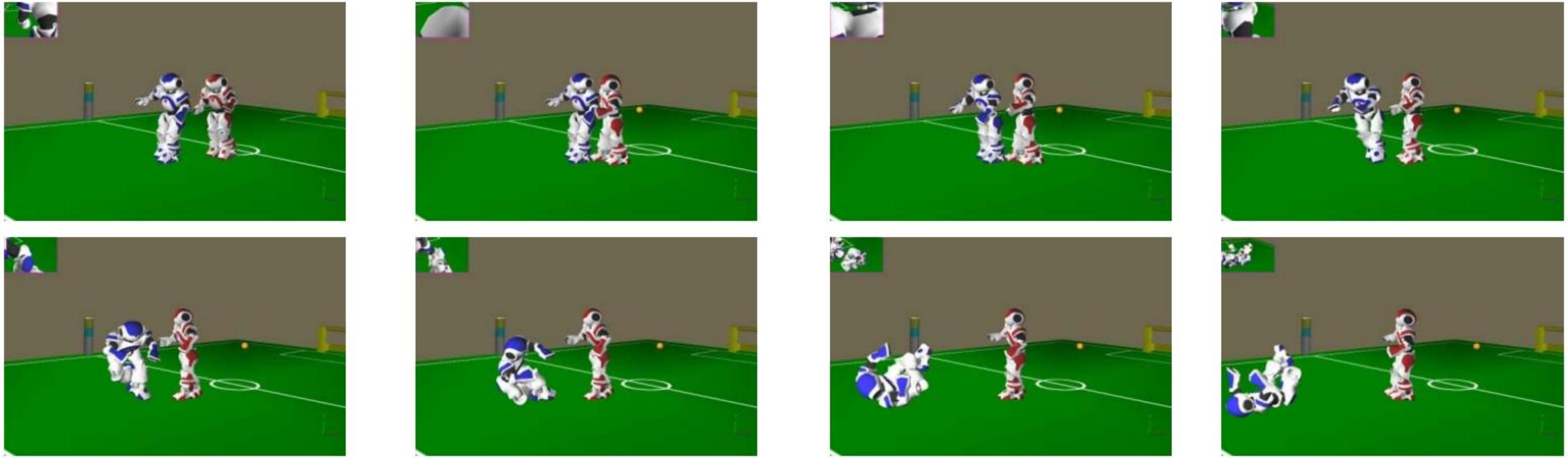
- **ZMP/COP with respect to the support polygon**

Literature Review on Fall Detection

- **Fall Detection and Management in Biped Humanoid Robots (Ruiz-del-Solar 2010)**

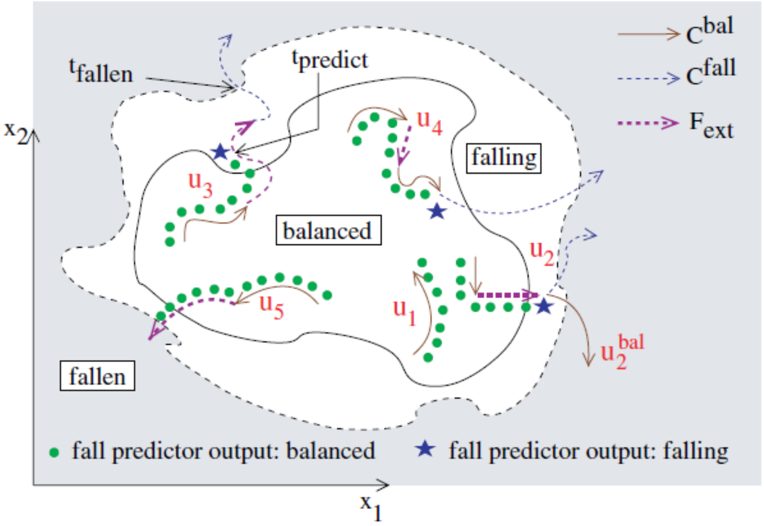
- **Online heuristic comparison approach: A comparison between the body's roll/pitch angle and angular velocities with respect to some heuristic thresholds.**

- **Body's roll/pitch angle and angular velocities are accessible through a Kalman filtered sensor data.**



Literature Review on Fall Detection

- **Learning to Predict Humanoid Fall (Kalyanakrishnan 2011)**

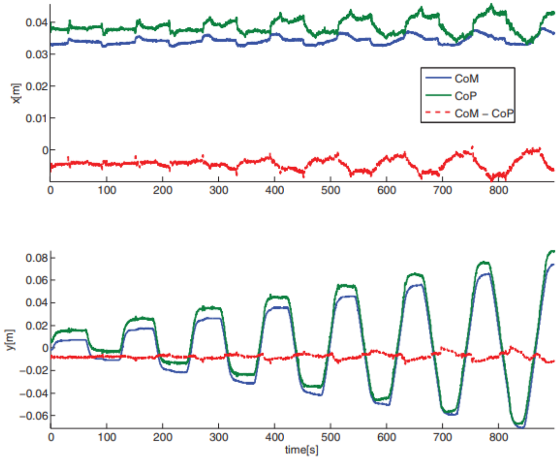


- **Supervised learning app** … **e**
- **Training is conducted in the simulation software. Training trajectories are gathered by imposing impulse to the representative robot configuration in different contact states.**

Physical Quantity	Type	#Features
CoM displacement	Real-valued	3
Linear momentum	Real-valued	3
Angular momentum about CoM	Real-valued	3
Rate of change of linear momentum	Real-valued	3
Rate of change of angular momentum about CoM	Real-valued	3
Foot contact mode	Discrete, 16 values	1

Literature Review on Fall Detection

- **Center of Mass Estimator for Humanoids and its Application in Modelling Error Compensation, Fall Detection and Prevention (Xinjilefu 2015)**



$$\hat{x} = [x_{com}, \dot{x}_{com}, x_{offset}]^T$$

$$x_{k+1} = f(x_k, u_k) = Ax_k + Bu_k$$
$$y_k = h(x_k) = Cx_k + Du_k$$

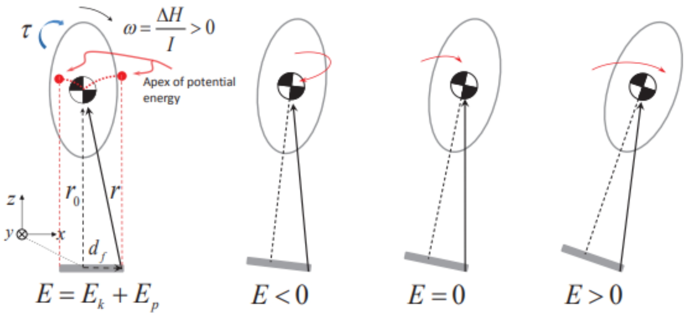
$$A = \begin{bmatrix} \mathbb{I} + \frac{1}{2}\omega^2\Delta t^2\mathbb{I} & \Delta t\mathbb{I} & \frac{1}{2}\omega^2\Delta t^2\mathbb{I} \\ \omega^2\Delta t\mathbb{I} & \mathbb{I} & \omega^2\Delta t\mathbb{I} \\ 0 & 0 & \mathbb{I} \end{bmatrix}, \quad C = \begin{bmatrix} \mathbb{I} & 0 & 0 \\ \omega^2\mathbb{I} & 0 & \omega^2\mathbb{I} \end{bmatrix}, D = \begin{bmatrix} 0 \\ -\omega^2\mathbb{I} \end{bmatrix}$$

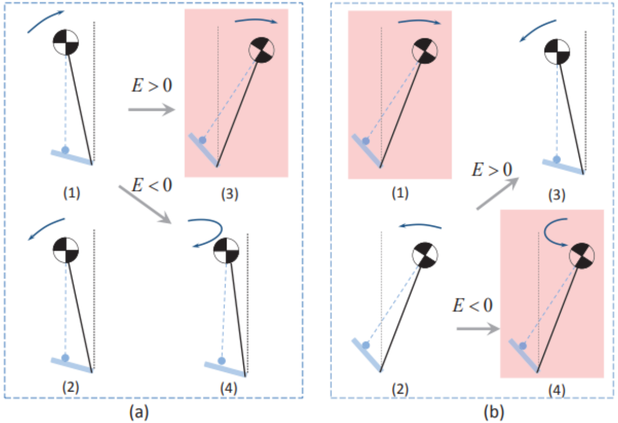
$$B = \begin{bmatrix} -\frac{1}{2}\omega^2\Delta t^2\mathbb{I} \\ -\omega^2\Delta t\mathbb{I} \\ 0 \end{bmatrix}$$

- **An COM estimation correction with an offset value by Kalman filter algorithm.**
- **Comparison of Capture Point within the Support Polygon for the fall indicator.**
- **Upper and Low bounds of force/torque sensor to determine the support polygon.**

• Literature Review on Fall Detection

- **Fall Prediction of Legged Robots Based on Energy State and Its Implication of Balance Augmentation: A Study on the Humanoid (Li 2015)**



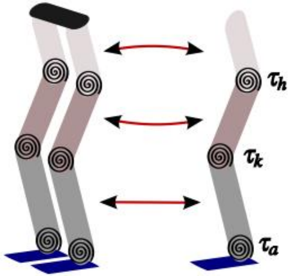


- **Inverted Pendulum Orbital Energy as the fall indicator**

- **Our previous work (ICRA 2018) coincidentally shares the same idea with this method.**

Literature Review on Fall Detection

- **Lyapunov Stability Margins for Humanoid Robot Balancing (Tsagarakis 2016)**



- **Model-based Lyapunov function to monitor the robot motion stability**

- **In addition to a simplified robot model, the motor dynamics is also considered in the formulation of the Lyapunov function.**

$$M_J(q)\ddot{q} + N\dot{q} + C(q,\dot{q})\dot{q} + S_m^T P(S_m q - \theta) = \tau_g(q)$$

$$J\ddot{\theta} + D\dot{\theta} - PS_m q + P\theta = V_{TG}V_m$$

- **Energy threshold can be chosen heuristically to determine the stability status of the given robot state.**