Robotics and Animatronics in Disney

Lecture 4: Controlling Humanoid Robots with Motion Capture Data

Katsu Yamane
kyamane@disneyresearch.com
Disney Research Pittsburgh

Goals

- Introduce two approaches to using human motion data to control humanoid robots
- Discuss whether using human motion is essential in realizing natural motions

Disney Research, Pittsburgh

Motion Capture Data

- Stylistic, natural and expressive
- Provide good reference for
  - Humanoid robot control
  - Character animation


Vicon


MetaMotion


Organic Motion

Disney Research, Pittsburgh

Robotics

Learning/control for humanoid robots

- Imitation [Ude et al. 2000] [Jenkins et al. 2002] …
- Humanoid robot control [Safonova et al. 2003] [Nakaoka et al. 2003] [Ott et al. 2008]


[Nakaoka 2003]

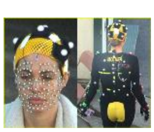

[Ott 2008]

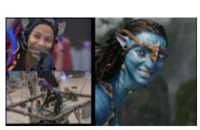
Disney Research, Pittsburgh

Production Usage

- Video games
- Films: The Polar Express (2004), Beowulf (2007), Avatar (2009), A Christmas Carol (2009) …
- Characters matching the actor, or extensive manual retargeting


The Polar Express


Beowulf


Avatar

Disney Research, Pittsburgh

Limitations

Ratatouille (2007)

Motion capture is not shortcut (at least for now)
- Subject and environment specific
- Mostly human subjects
- Limited to human capability

Disney Research, Pittsburgh

Extending the Horizon

- Subject and environment specific
  - Characters/robots with different mass properties?
- Mostly human subjects
  - Non-humanoid characters? (Lecture 5)
- Specific to environment
  - Different environment? (Lecture 5)

Online Balancing and Tracking by Force Control

[Yamane, Hodgins 2009]
[Yamane, Anderson, Hodgins 2010]

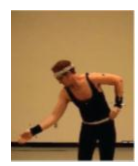
Disney Research Pittsburgh

Goal

Track human motion capture data while balancing

- No task-specific knowledge
- Force (torque) control
- Online (e.g., teleoperation)

Issues

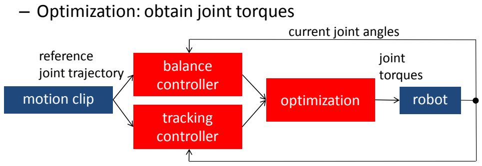


≠
link length
mass/inertia
motion range
torque range

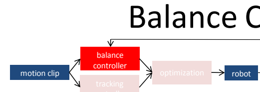
tracking | balancing

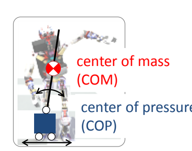
Managing Tracking and Balancing

- Main components
  - Balancing: optimal control for simplified robot model
  - Tracking: follow reference motion
  - Optimization: obtain joint torques

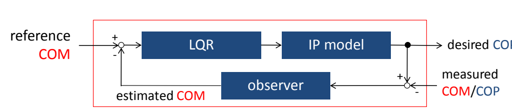


Balance Controller

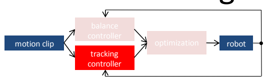




- Based on a simplified robot model
- Typical choice
  - Linear inverted pendulum (IP) model
  - Linear quadratic regulator (LQR)



Tracking Controller

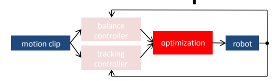


- Resolved acceleration control [Luh et al. 1980]
  - Desired joint accelerations

    $$\hat{\ddot{\theta}} = \ddot{\theta}_{ref} + k_d(\dot{\theta}_{ref} - \dot{\theta}) + k_p(\theta_{ref} - \theta)$$
  - Desired contact accelerations (linear/angular)

    $$\hat{\ddot{r}} = \ddot{r}_{ref} + K_d(\dot{r}_{ref} - \dot{r}) + K_p(r_{ref} - r)$$

Optimization



Quadratic cost function with an analytical solution

Obtain joint torque τ and contact force f_c that minimize

COP error + joint acc error + foot acc error + contact force magnitude

subject to complete robot dynamics $M\ddot{q} + c + g = S^T\tau + J_c^T f_c$

Contact Force and Hardware Limits

- Friction/COP constraints
  - Inequality constraints increase calculation time
  - → Use larger weights for contact friction and moment
- Joint limits
  - Angle: considered in inverse kinematics
  - Velocity: large damping near limits
  - Torque: if a joint torque exceeds the limit, in the next frame
    - Set its reference torque to the limit
    - Add difference from reference torque to the cost function

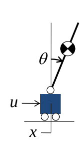
Simulation

- Sarcos robot model
  - 25 joints (neck and wrist joints fixed)
  - Inertia parameters from the CAD data
  - Joint motion ranges from experiments
  - Joint velocity/torque limits from design specifications
  - Assume trunk position/orientation are available

Simulation

- Parameters
  - LQR

    $$Q = diag\{1 \times 10^7, 1 \times 10^8, 1 \times 10^2, 1 \times 10^3\}, R = 1$$
  - Feedback gains: $k_p = 4, k_d = 4$ in most examples
  - Optimization: all weights 1

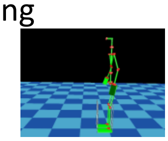


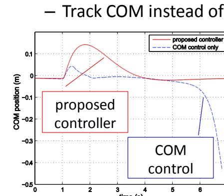
$$x = (x \quad \theta \quad \dot{x} \quad \dot{\theta})^T$$

$$J = \int_0^\infty (x^T Q x + u^T R u) dt$$

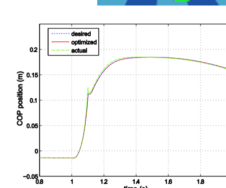
Simple Balancing

- Push 250N for 0.1s from back
- Comparison with COM control
  - Track COM instead of COP





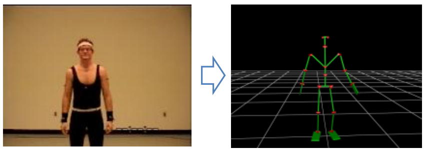
COM position



Desired/optimized/actual COP positions

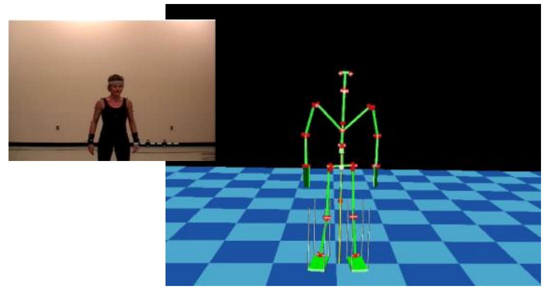
Motion Capture Data

- Inverse kinematics [Yamane and Nakamura 2003]
  - Marker positions (including head markers)
  - Joint angle limits
  - Flat contact



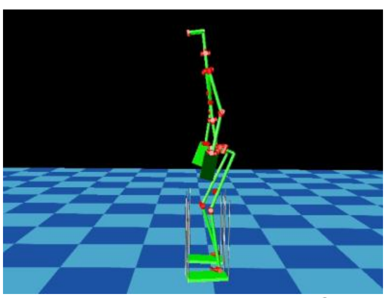
Disney Research, Pittsburgh

Results: "I'm a little teapot"



Disney Research, Pittsburgh

Disturbances



Disney Research, Pittsburgh

Hardware Experiment Issues

The robot is initially stationary

But the optimized torques do not match the actual torques!

- Model error
  - Assume constant error
  - Initialize from measured joint torques and contact forces
- Contact force distribution
  - Keep the initial COP at each foot
  - Update from optimized and actual contact forces

Disney Research, Pittsburgh

Model Error

$$M\ddot{\theta} + c + g + h = S^T\tau + J_c^T f_c$$

- Initialize at first frame
  - Measurements: θ_0, τ_0, f_{c0}
  - Stationary: $\dot{\theta}_0 = 0, \ddot{\theta}_0 = 0$
    - → $h = S^T\tau_0 + J_c^T f_{c0} - g(\theta_0)$

Disney Research, Pittsburgh

Contact Force Distribution

- Keep the initial COP at each foot
  - New cost function term: moment around initial COP
- Feed back the contact force difference to bias the optimized contact forces

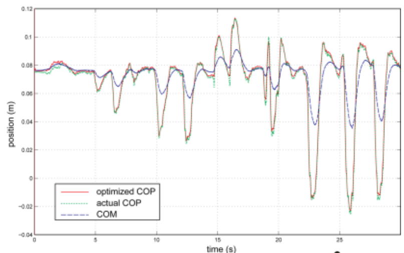
$$M\ddot{\theta} + c + g + h + J_c^T \Delta f_c = S^T\tau + J_c^T f_c$$

$$\Delta f_c = K_I \int \left(f_c^{act}(s) - f_c^{opt}(s) \right) ds$$

Disney Research, Pittsburgh

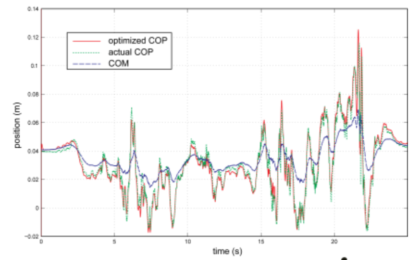
Hardware Experiment Results

COP tracking in balancing (push)

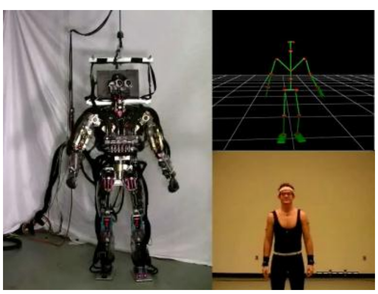


Hardware Experiment Results

COP tracking in "teapot" tracking



Hardware Experiment



Extension to Contact State Changes

[Yamane, Hodgins 2011]

Disney Research Pittsburgh

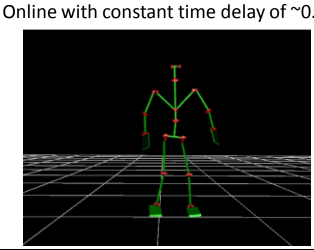
Contact State Change Issues

- Liftoff: if the leg still supports some weight, robot falls
- Touchdown: if the other foot still fully supports the weight, touchdown may not occur at the specified time

COP needs to be at desired location to realize timely contact state change

Mapping Algorithm Features

- Step time/location as in reference motion
- Generic motions: no segmentation/parameterization
- Online with constant time delay of ~0.5s



Example: Tai-Chi from CMU Graphics Lab Motion Capture Library http://mocap.cs.cmu.edu/

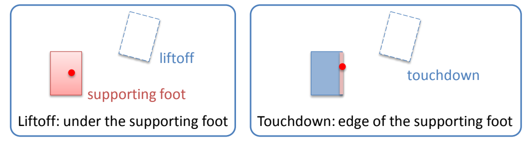
Mapping Algorithm Features

Assumptions on the reference motion

- At the first frame, both feet are in flat contact
- At every frame, the set of links in contact is known (don't have to be flat contact)
- At every frame, at least one foot is in contact (no flight phase)

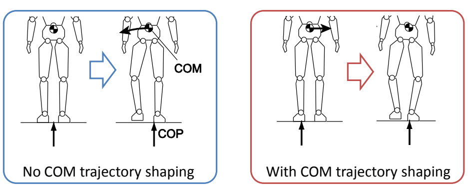
Extension to Stepping

Desirable COP locations at liftoff and touchdown



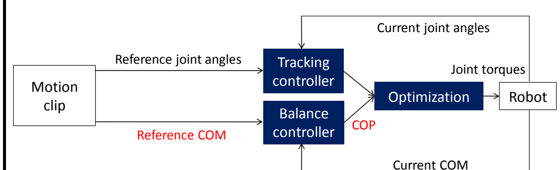
Extension to Stepping

How? Shaping COM trajectory



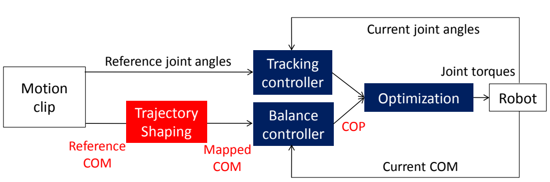
Previous Controller

Reference COM determines the COP



New Controller

Trajectory shaping: modify the reference COM to bring future COP into the desired region



$t_0 \le t \le t_0 + T$

Computing the New COM

Discretized state equation of the balance controller

$$\begin{cases} x_{i+1} = Ax_i + Bu_i \\ y_i = Cx_i \end{cases}$$

- x_i: state of the robot
- u_i: reference COM
- y_i: COP to track reference COM

Predict COP k frames later

$$y_k = C\left(A^k x_0 + M\hat{u}\right)$$

- $M = (A^{k-1}B \quad A^{k-2}B \quad \dots \quad B)$
- x_0: current state
- $\hat{u} = (\hat{u}_0 \quad \hat{u}_1 \quad \dots \quad \hat{u}_{k-1})$: original COM trajectory

Computing the New COM

If y_k is not in the desired region at $k = n \ (n < N)$

1. Choose a desired COP y_{ref}
2. Obtain new COM trajectory u that brings COP closer to y_{ref}

   Minimize $Z = \frac{1}{2}(y_{ref} - y)^T(y_{ref} - y) + \frac{1}{2}(\hat{u} - u)^T W(\hat{u} - u)$

   subject to $y = C(A^n x_0 + Mu)$

3. Use the first element of u as the new reference COM

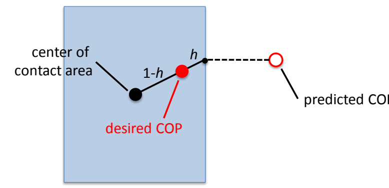
$W = diag\{wi^2\} \ (i = 1,2, \ldots, n)$

To smoothly connect with the rest of the motion

Disney Research, Pittsburgh

Choosing the Desired COP

1) if predicted COP is out of contact area

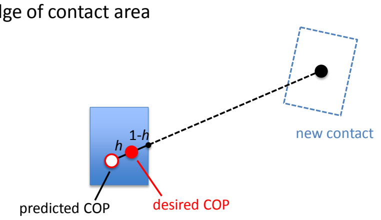


$h = \frac{h_{max} n}{N}$ to prevent large COP change in short time

Disney Research, Pittsburgh

Choosing the Desired COP

2) If a foot should touch down but the COP is not at the edge of contact area



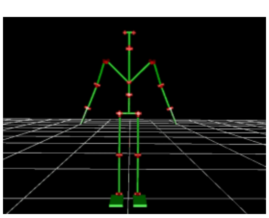
Disney Research, Pittsburgh

Simulation

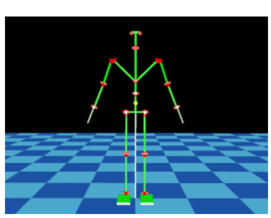
- Main control timestep: 1ms
- Discretization timestep: 5ms
- COP prediction up to 100 frames (0.5s)
- Desired COP parameters:
  $h_{max} = 0.8, w = 0.01$
- Robot model: Carnegie Mellon/Sarcos humanoid robot

Disney Research, Pittsburgh

Simple Example



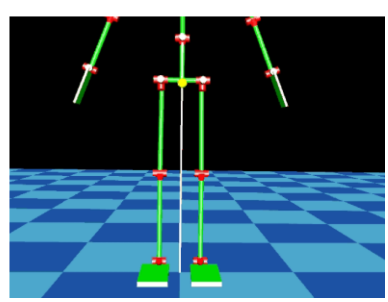
reference motion (created manually) 0.2x



without COM Trajectory shaping 0.2x

Disney Research, Pittsburgh

Simple Example

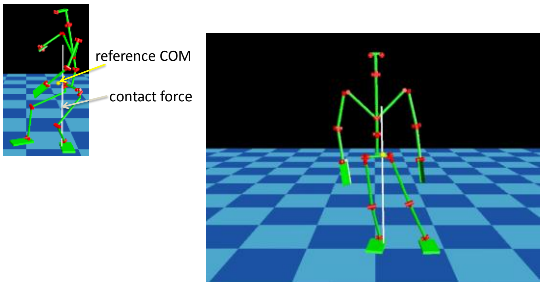


0.2x

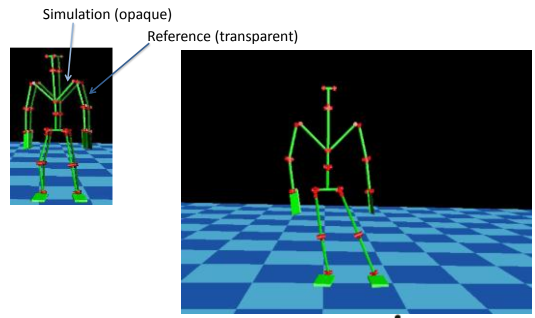
with COM trajectory shaping

Disney Research, Pittsburgh

Tai-Chi



Tai-Chi: Simulation Result

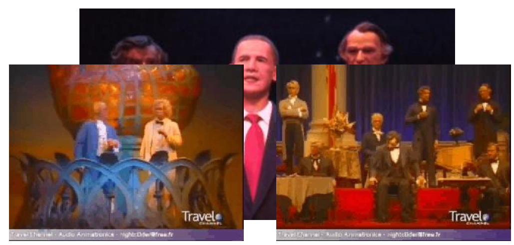


Indirect Control of Contact Forces

[Mistry, Murai, Yamane, Hodgins 2010]

Disney Research Pittsburgh

Contacts and Forces in Robot Control



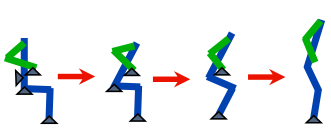
Real physical interaction is a key component currently missing from Disney animatronics

Goal: Improve realism through natural physical interaction and contacts

Case Study: Standing up from a chair

Challenges:

- Stability and robustness
- Large, quick, center-of-mass shift
- Contact state changes
- Contact force control



Carnegie Mellon/Sarcos Humanoid Robot

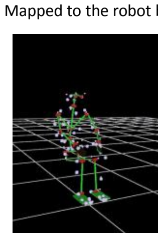
Natural and stylized motions from motion capture

Natural:

Stylized:

Elderly Style

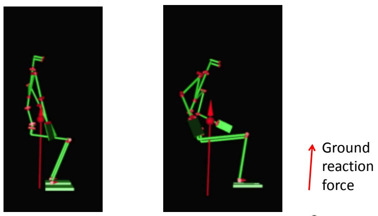
Mapped to the robot body



Also record contact forces and muscle activities

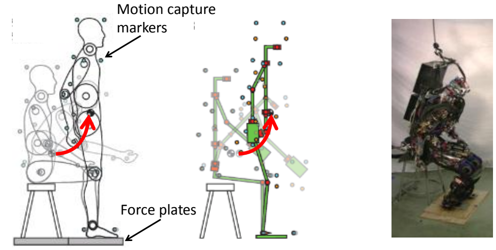
Force control for stability and robustness

Simple motion playback will not work
Often more robust to control robots when contacts are allowed
(e.g. touching the armrests vs. floating closely above them)



Natural   Elderly Style

Indirect Control



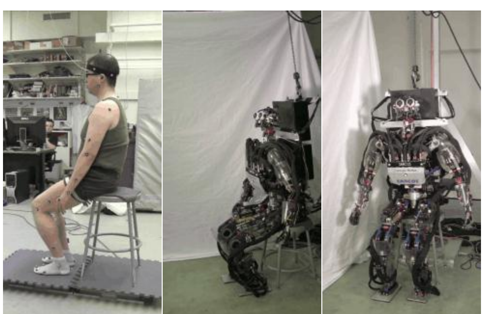
Capture motion
Estimate COM trajectory

Inverse kinematics with
COM and marker constraints

Joint position control

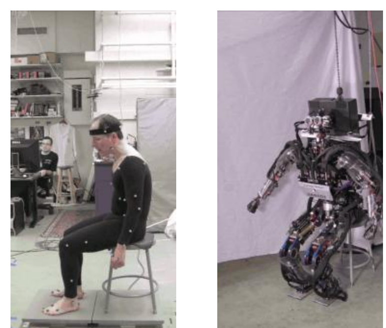
Result 1: Natural Motion

First step- position control



Result 2: Stylized Elderly Motion

First step- position control



Summary

- Two approaches to using human motion data for controlling humanoid robots
  - Blending tracking and balancing controllers on force-controlled humanoid robot
  - Indirectly controlling contact forces by matching the COM motion

Discussion

- Is human motion source necessary for synthesizing human-like motions?
- Is human-like motion enough for making robots look alive?