Human Motion Capture

Instructor Dr. Anup Basu

Notes prepared in part by:
Dr. Amirhossein Firouzmanesh, currently in Microsoft

Outline

- What is Motion Capture (MoCap)?
- MoCap Technology
- Data Representation
- Using MoCap (Mesh Skinning)
- MoCap Compression

What is MoCap

- **MoCap** is the process of recording the movement of objects or people
- Used in: Military, Entertainment, Sports, and medical applications, computer vision and robotics
- Also, filmmaking and video game development:
  - Recording actions of human actors
  - Using that information to animate digital character models in 2D or 3D computer animation

Example of Motion Capture (Avatar)

- http://www.youtube.com/watch?v=P2_vB7zx_SQ#!

Representation of Motion

1- 3D position of markers (keypoints)

2- Using a hierarchical structure (skeleton)

- Relative rotation angles of each of the joints

- Rotational angles to marker position -> Forward kinematics
- Marker position to rotation angles: Inverse kinematics

How to Capture Motion (Challenges)

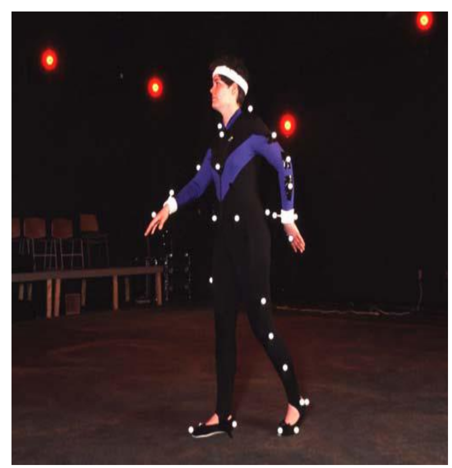
- Accurate Measurement
- Comfort of the actor
- Processing time
- Occlusion
- Correspondence
- Noise

How to Capture Motion (Technology)

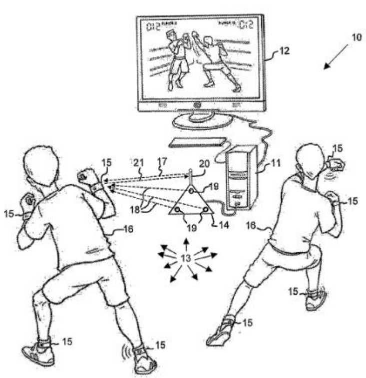
- Prosthetic(Electromechanical)

- Electromagnetic

- Optical



- Acoustic



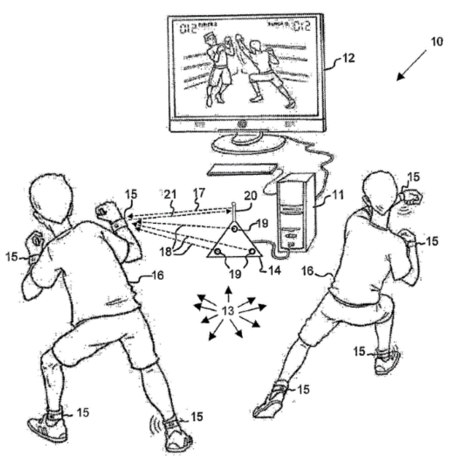
Prosthetic (Electromechanical) MoCap

- Uses Potentiometers on a plastic exoskeleton
- The actor "wears" the exoskeleton, and then act out his or her movements
- Pros:
  - Accurate data
  - Real time (fast)
  - Wide range
- Cons:

Weight & discomfort for the actress

Acoustic sensing technology

- Parts
  - 3 audio receivers
  - Array of audio transmitters on various parts of the performer's body
- Mechanism
  - Transmitters sequentially triggered to output a "click"
  - Each receiver measures the time it takes for the sound to travel from each transmitter
  - The calculated distance of the three receivers is triangulated to estimate a point in 3D space
- Disadvantage:
  - Sequential data (not a "snap shot" of the performer’s skeletal position)
  - Limited Range
  - Limited number of sensors

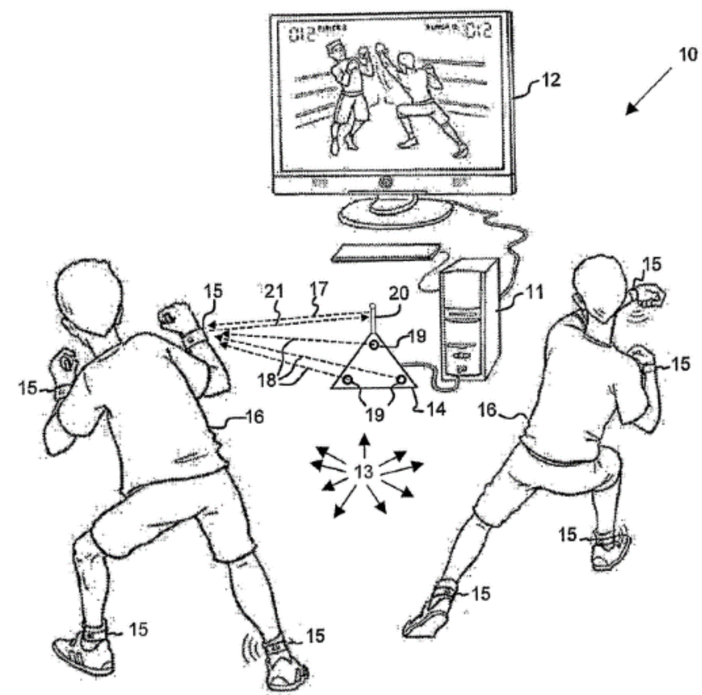


Calculating 3D Position of a Transmitter

- Given the Distance between 2 Receivers & the Distances between Each Receiver and the Transmitter, we can One Traingle in 3D

- With a 3rd Receiver, we can determine a Second Triangle in 3D

- These two 3D triangles can Uniquely determine the 3D location of the transmitter!



Electro-magnetic Sensing Technology

- Parts:
  - A centrally located transmitter
  - A set of receivers strapped on to various parts of the performer's body
- Mechanism
  - Receivers measure their spatial relationship to the transmitter
  - Each receiver is connected to an interface that can be synchronized to prevent data skew
  - Data stream: 3D positions and orientations for each receiver
  - Inverse kinematics system

- Benefits
  - No occlusion problem

- Disadvantages
  - Hindrance of cables
  - Lack of sufficient receivers
  - Limited capture area
  - Affected by any sizable areas of metal in the vicinity of the capture area, such as girders, posts, and so on

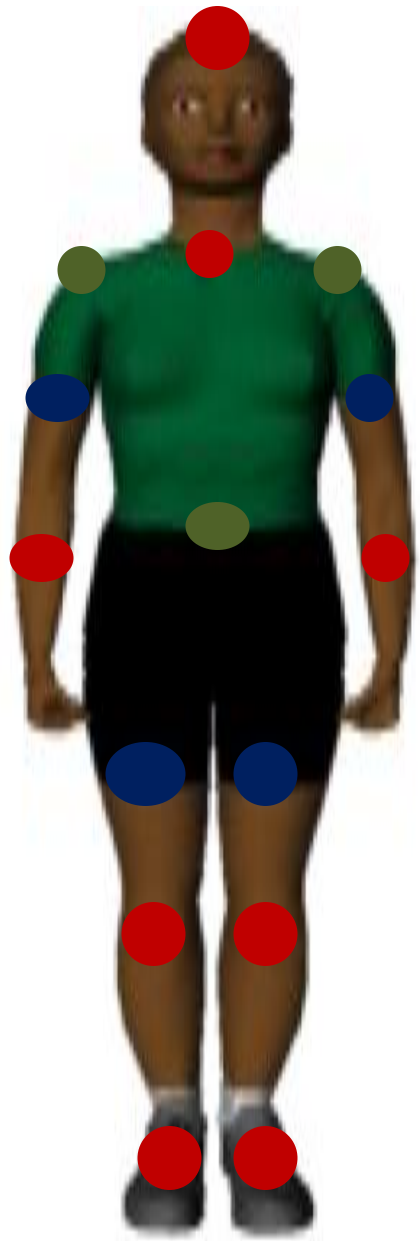

Optical sensing technology

- Quite popular over the last decade.
- Offers the performer the most freedom of movement
  - No cables required
- Mechanism:
  - Markers: directionally reflective balls
  - At least three video cameras
  - Each camera equipped with a light source
  - Computer: Calculates a 3D position of each marker from the camera views
  - This data is typically applied to an inverse kinematics system, to animate a skeleton

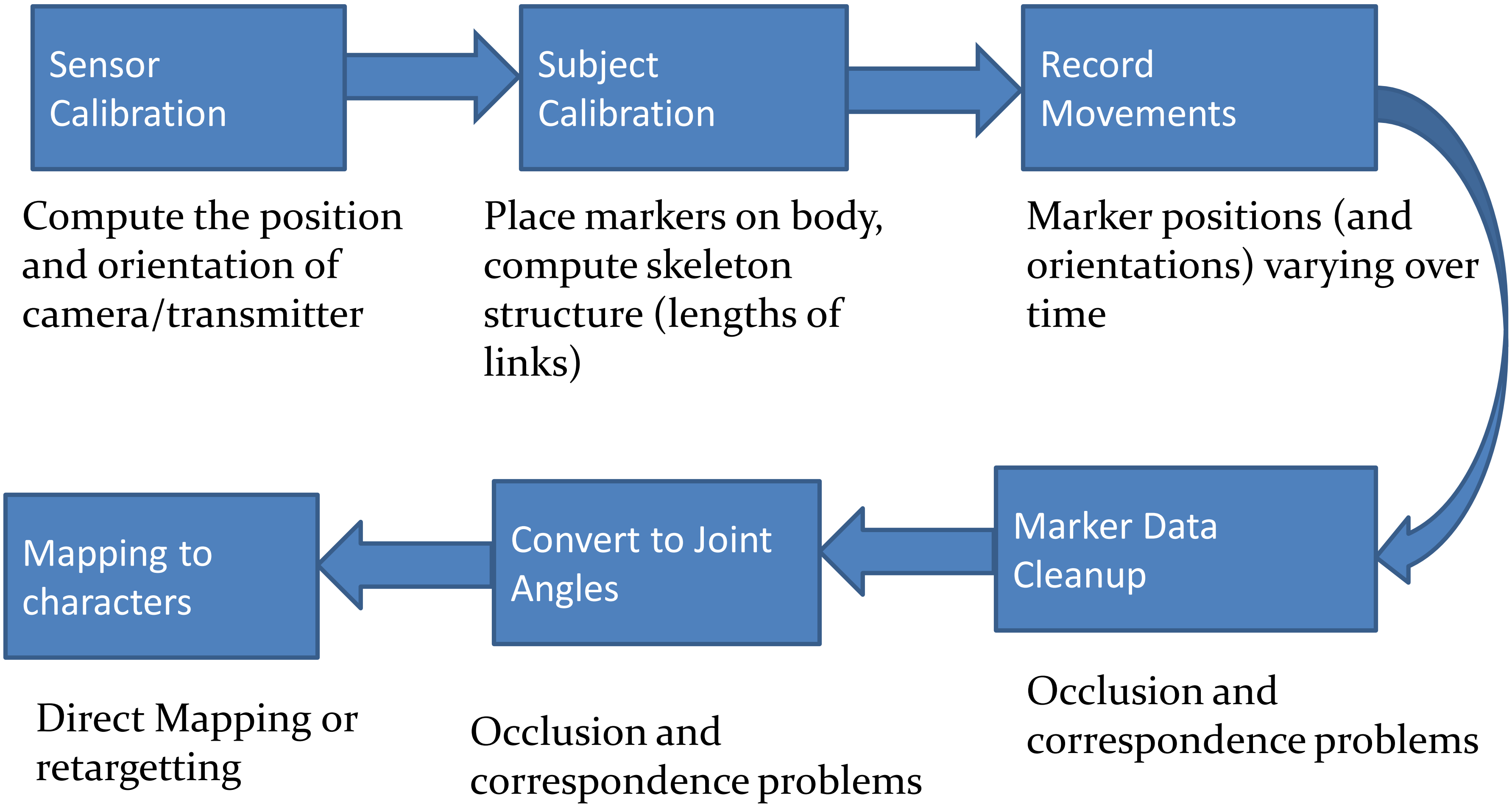
Marker Placements

- Location should move rigidly with joints

- Stay away from bulging muscles



Production Pipeline

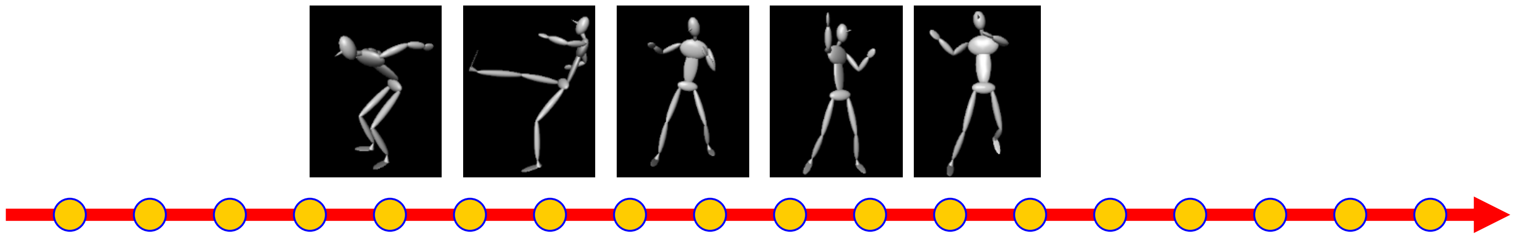


Outline

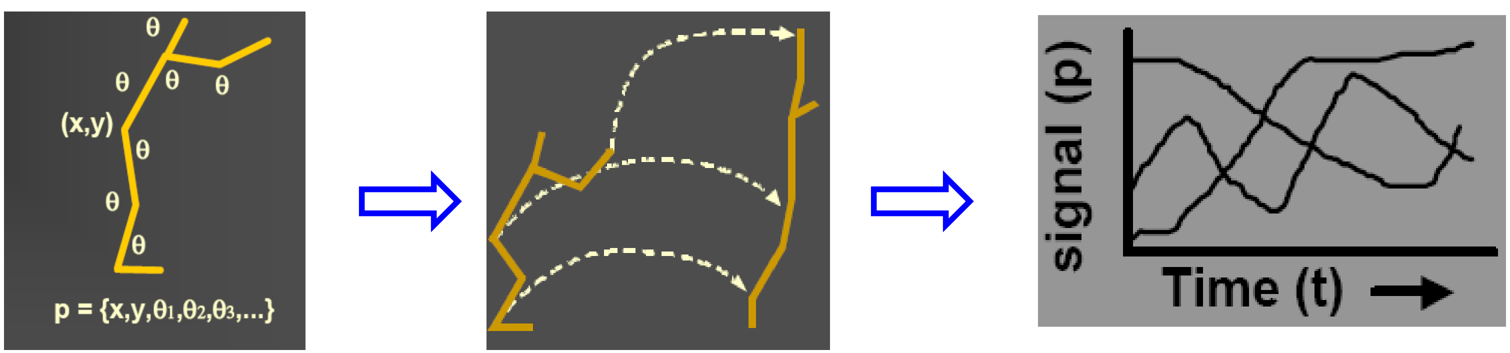
- What is Motion Capture (MoCap)?
- MoCap Technology
- Data Representation
- Using MoCap (Mesh Skinning)
- MoCap Synthesis
- MoCap Compression

Human motion representation

A sequence of poses: $q_1, q_2, \ldots q_T$

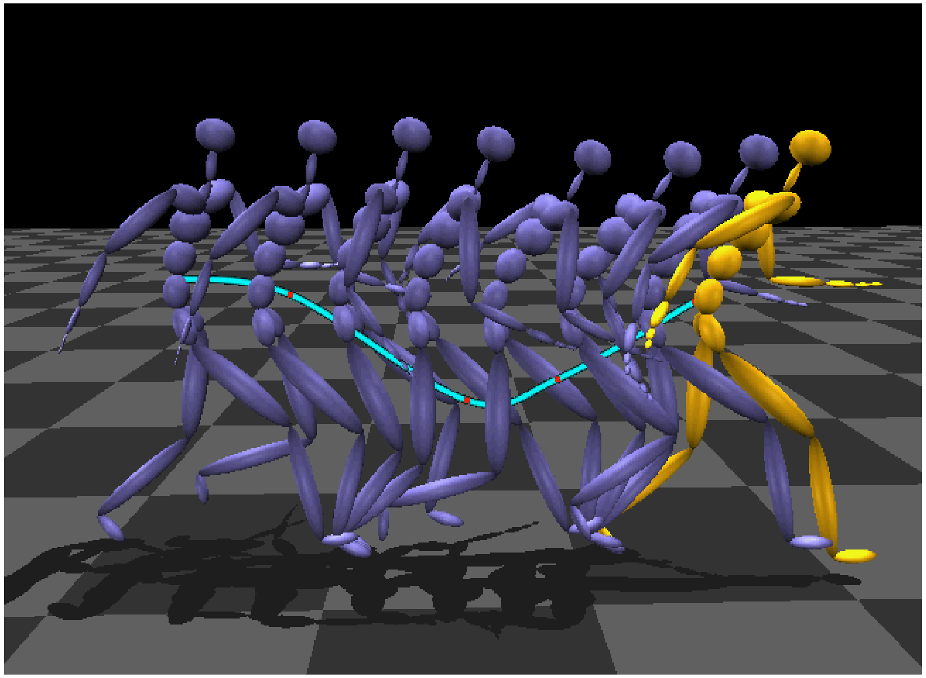


Each pose is represented as a high-dimensional vector q_t: R^n



Pose q_t | Motion $q_1, \ldots q_T$ | Motion trajectories

How to represent human motion?



-the body proportion and size of characters

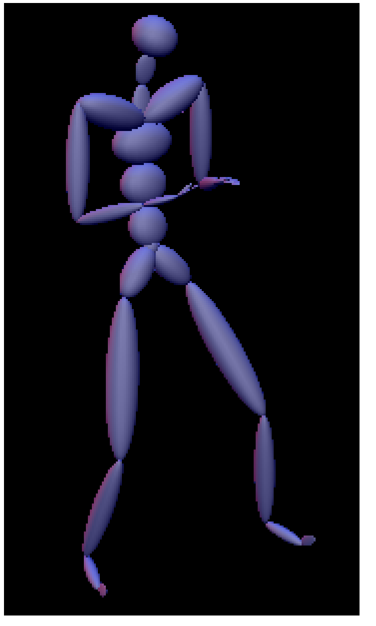
- the joint angle values across the entire sequence

- the 3D position of ONE JOINT relative to which all other joints can be calculated

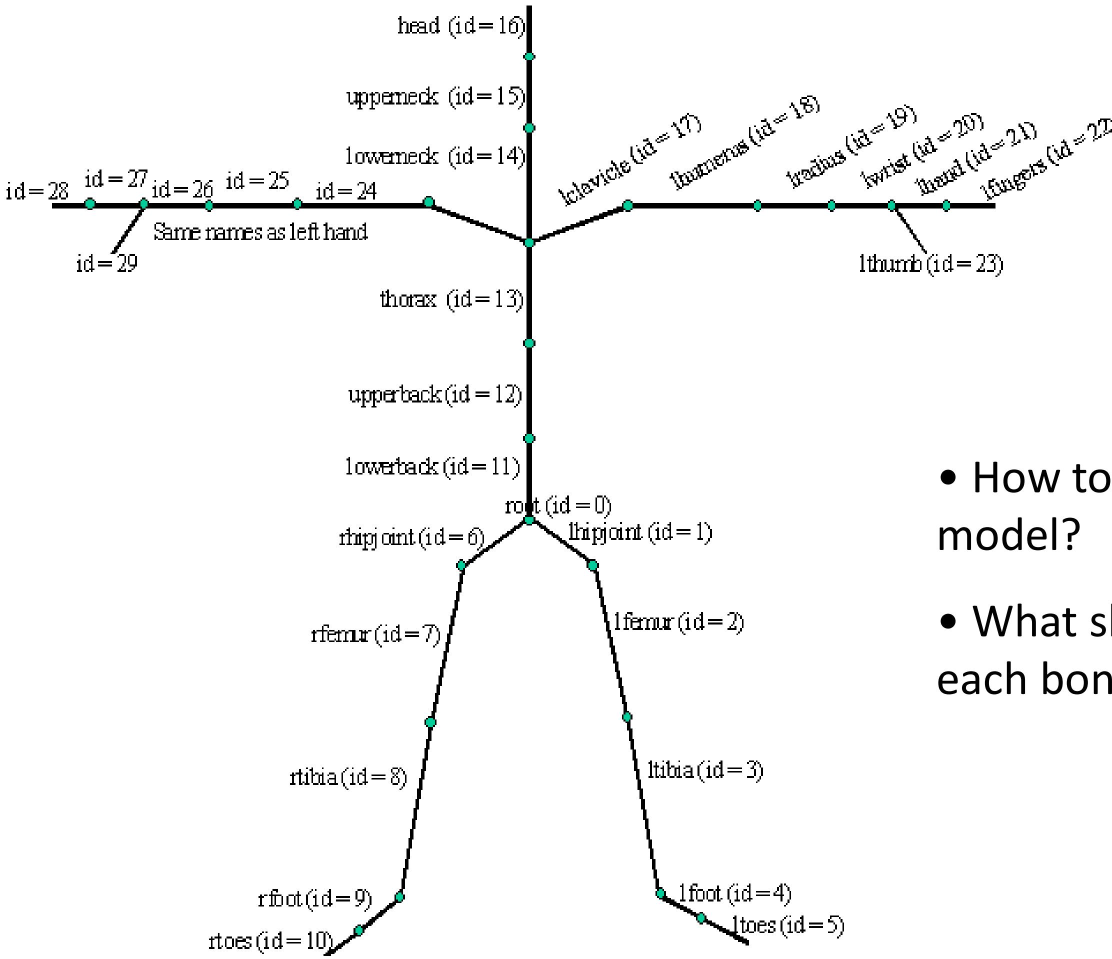
Motion Capture Data Files (Example)

Each sequence of human motion data contains two files:

- Skeleton file (.asf): Specifies the skeleton model of a character
- Motion data file (.amc): Specifies the joint angle values over the frame/time



Human skeletal model

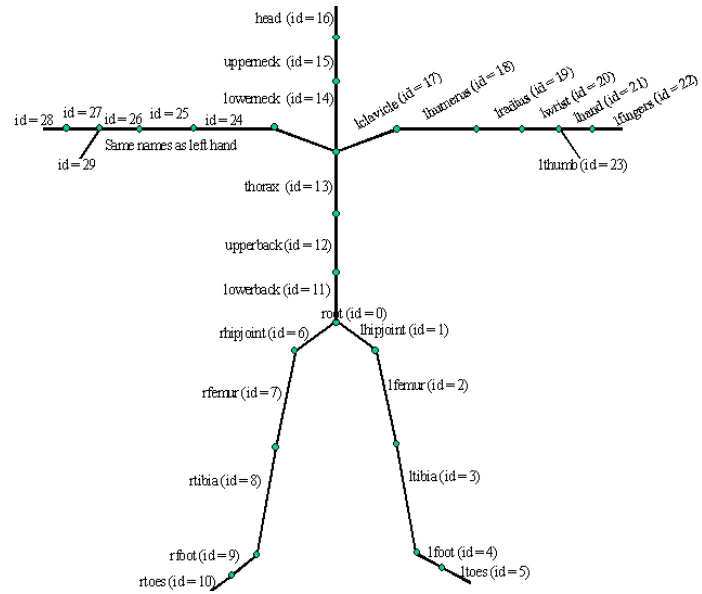


- How to describe the skeletal model?
- What should you know about each bone?

This is still a tree!

Human skeletal file (.asf)

- individual bone information
  - length of the bone
  - direction of the bone
  - local coordinate frame
  - number of Dofs
  - joint limits



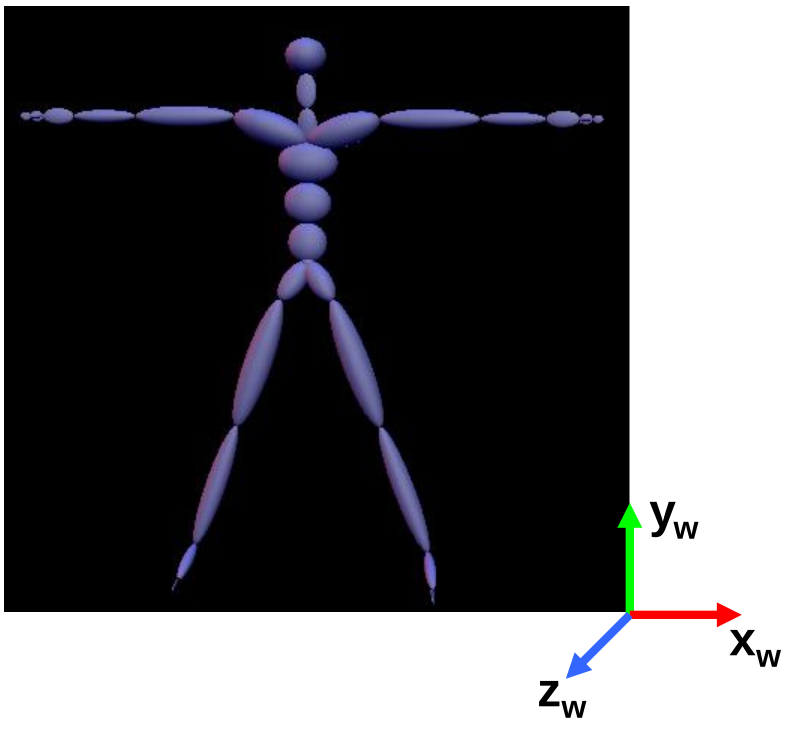
- bone hierarchy/connections

Individual bone information

```
begin
    id bone_id               /* Unique id for each bone */
    name bone_name           /* Unique name for each bone */
    direction dX dY dZ       /* Vector describing direction of the bone in world */ coor.
    system
    length 7.01722           /* Length of the bone*/
    axis 0 0 20 XYZ          /* Rotation of local coordinate system for
                             this bone relative to the world coordinate
                             system. In .AMC file the rotation angles
                             for this bone for each time frame will be
                             defined relative to this local coordinate
                             system**/
    dof rx ry rz             /* Degrees of freedom for this bone.
    limits (-160.0 20.0)     /* joint limits*/
           (-70.0 70.0)
           (-60.0 70.0)
    end
```

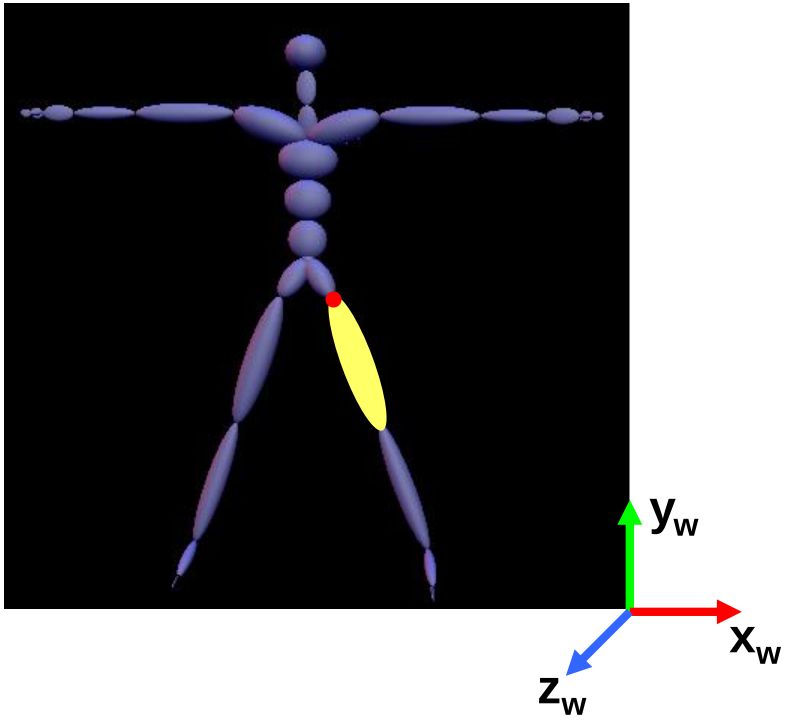
Individual bone information

begin
id 2
name lfemur
direction 0.34 -0.93 0
length 7.01722
axis 0 0 20 XYZ
dof rx ry rz
limits (-160.0 20.0)
(-70.0 70.0)
(-60.0 70.0)
end



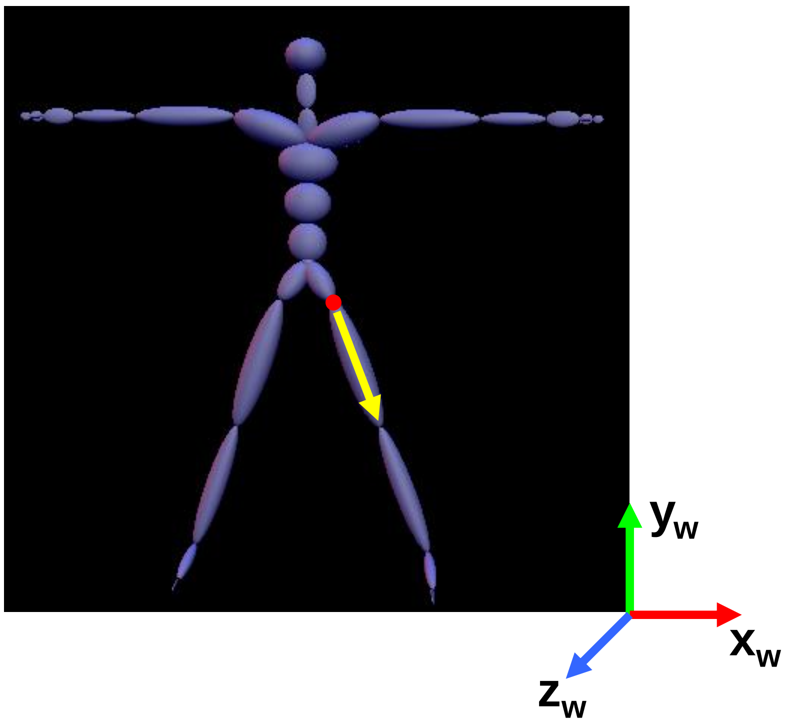
Individual bone information

begin
id 2
name lfemur
direction 0.34 -0.93 0
length 7.01722
axis 0 0 20 XYZ
dof rx ry rz
limits (-160.0 20.0)
(-70.0 70.0)
(-60.0 70.0)
end



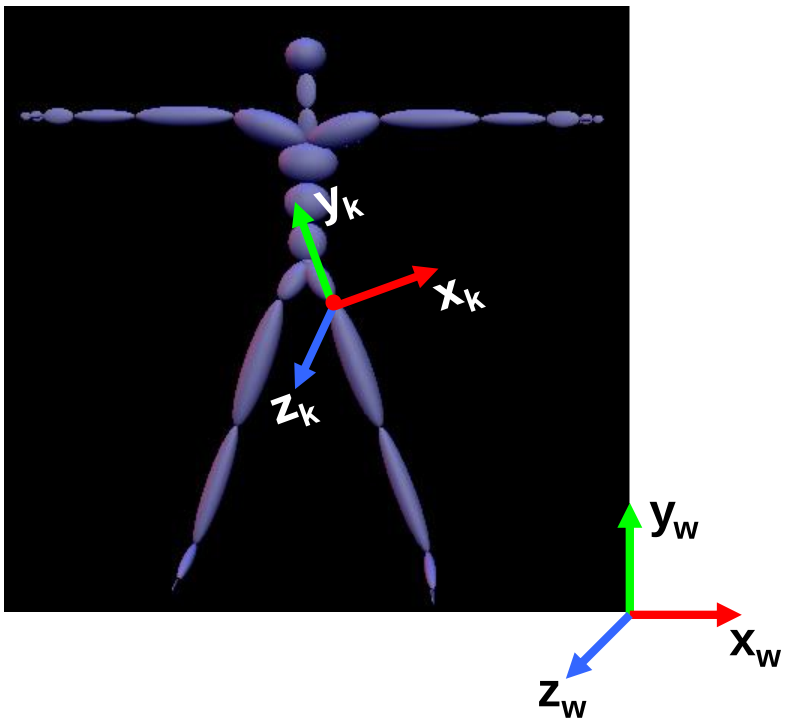
Individual bone information

begin
id 2
name lfemur
direction 0.34 -0.93 0
length 7.01722
axis 0 0 20 XYZ
dof rx ry rz
limits (-160.0 20.0)
(-70.0 70.0)
(-60.0 70.0)
end



Individual bone information

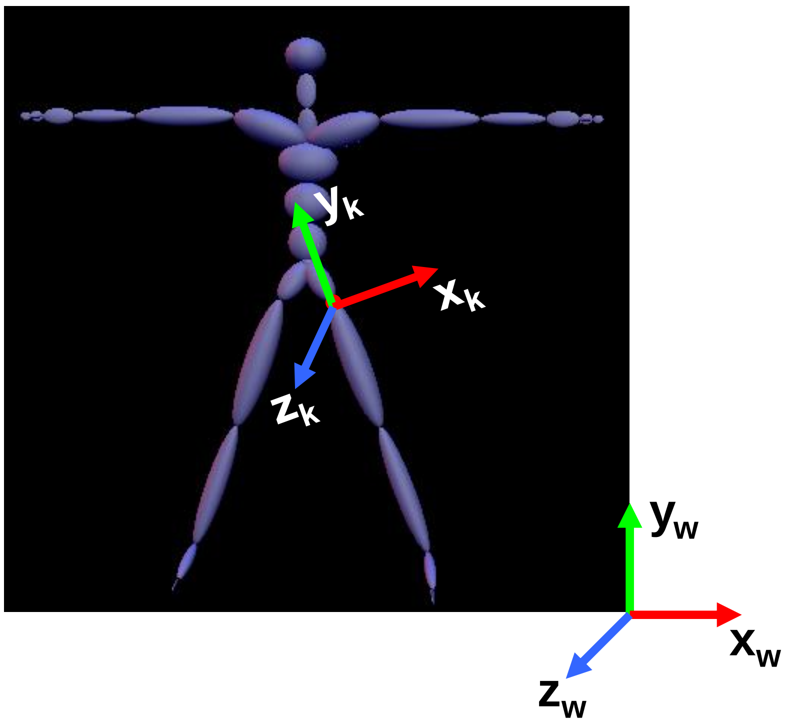
begin
id 2
name lfemur
direction 0.34 -0.93 0
length 7.01722
axis 0 0 20 XYZ
dof rx ry rz
limits (-160.0 20.0)
(-70.0 70.0)
(-60.0 70.0)
end



Euler angle representation: $R_k = R_z(\gamma)R_y(\beta)R_x(\alpha)$

Individual bone information

begin
id 2
name lfemur
direction 0.34 -0.93 0
length 7.01722
axis 0 0 20 XYZ
dof rx ry rz
limits (-160.0 20.0)
(-70.0 70.0)
(-60.0 70.0)
end

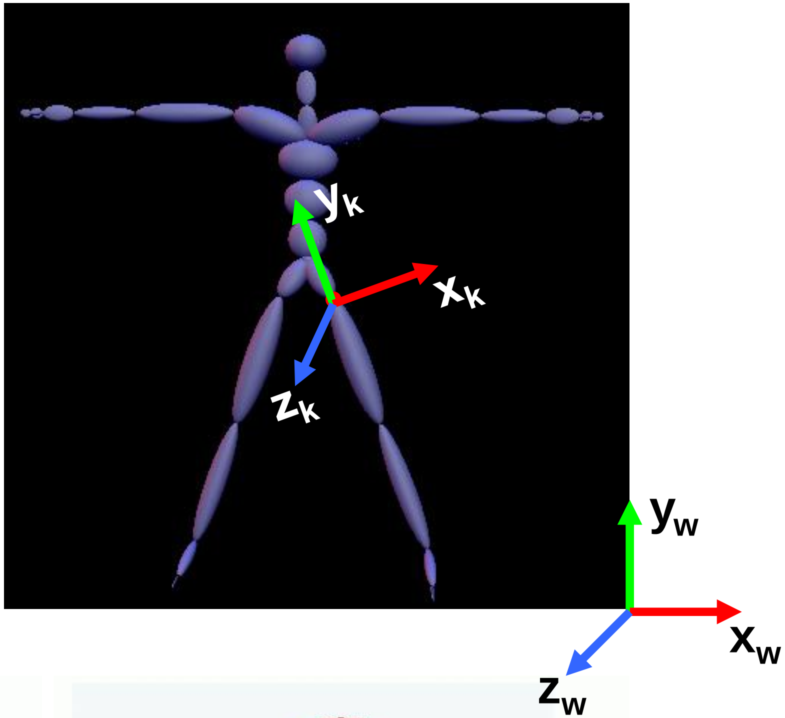


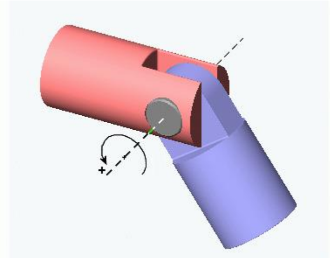
- The number of dof for this joint

- The minimal and maximum joint angle for each dof

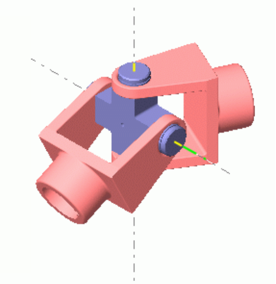
Individual bone information

begin
id 2
name lfemur
direction 0.34 -0.93 0
length 7.01722
axis 0 0 20 XYZ
dof rx ry rz
limits (-160.0 20.0)
(-70.0 70.0)
(-60.0 70.0)
end

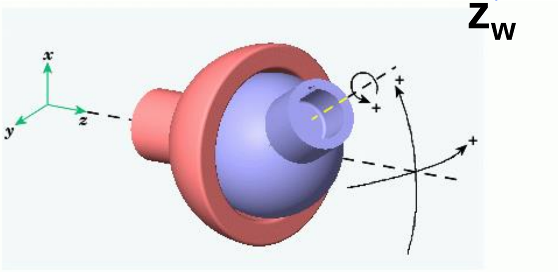




1-dof joint



2-dof joint

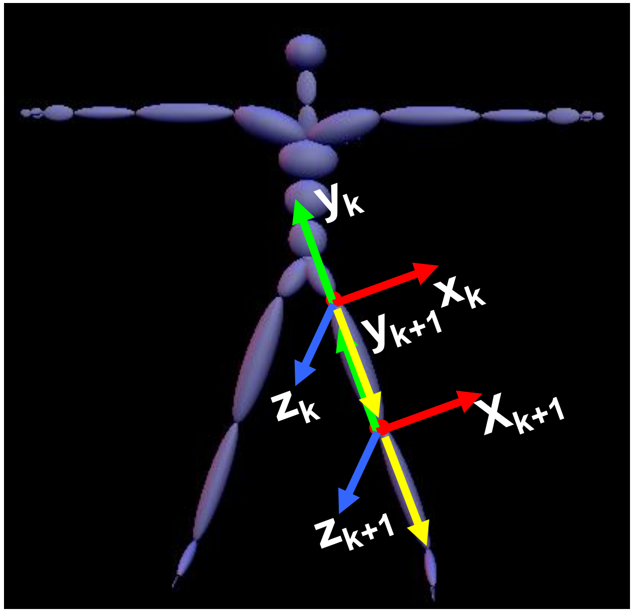


3-dof joint

Individual bone information

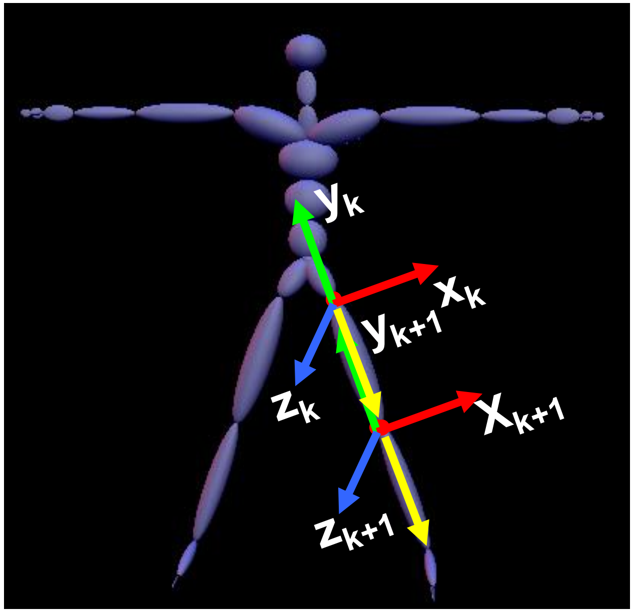
begin
id 2
name lfemur
direction 0.34 -0.93 0
length 7.01722
axis 0 0 20 XYZ
dof rx ry rz
limits (-160.0 20.0)
(-70.0 70.0)
(-60.0 70.0)
end

begin
id 3
name ltibia
direction 0.34 -0.93 0
length 7.2138
axis 0 0 20 XYZ
dof rx
limits (-10.0 170.0)
end



Individual bone information

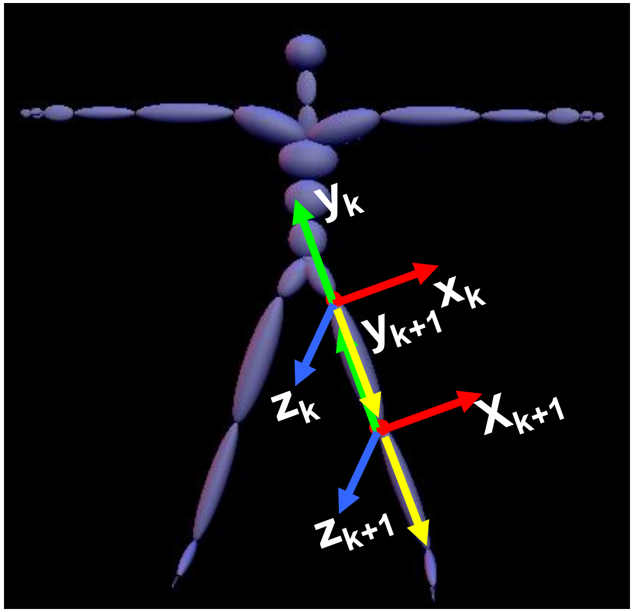
begin
id 2
name lfemur
direction 0.34 -0.93 0
length 7.01722
axis 0 0 20 XYZ
dof rx ry rz
limits (-160.0 20.0)
(-70.0 70.0)
(-60.0 70.0)
end
begin
id 3
name ltibia
direction 0.34 -0.93 0
length 7.2138
axis 0 0 20 XYZ
dof rx
limits (-10.0 170.0)
end



What do we miss?

Individual bone information

```
begin
id 2
name lfemur
direction 0.34 -0.93 0
length 7.01722
axis 0 0 20 XYZ
dof rx ry rz
limits (-160.0 20.0)
       (-70.0 70.0)
       (-60.0 70.0)
end
begin
id 3
name ltibia
direction 0.34 -0.93 0
length 7.2138
axis 0 0 20 XYZ
dof rx
limits (-10.0 170.0)
end
```



What did we miss?

- global position
- global orientation

Root representation

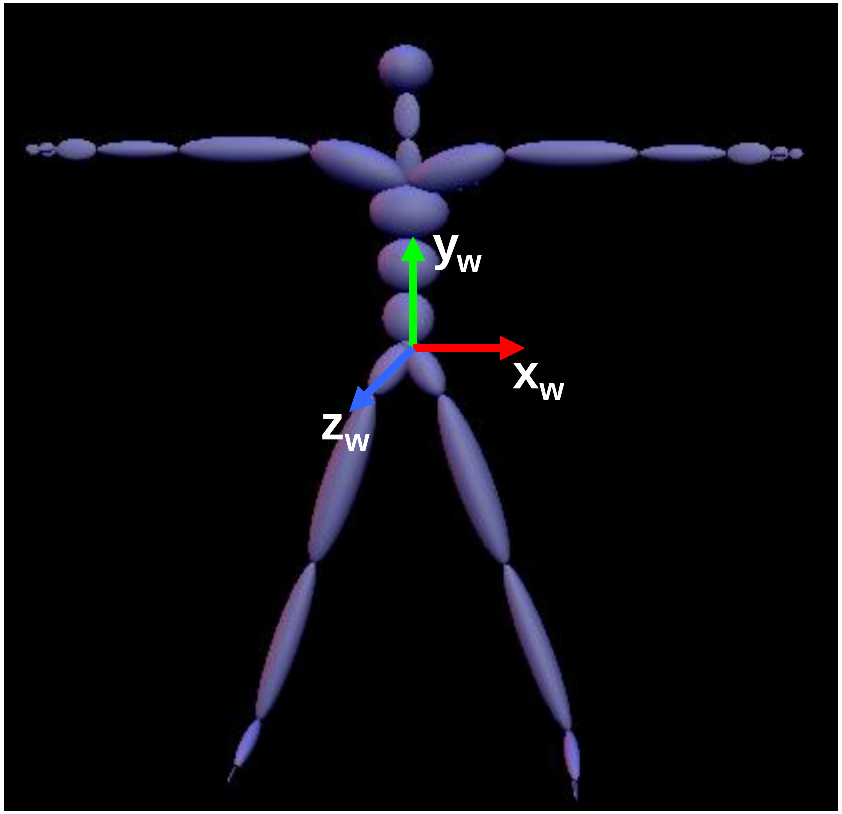
:root

order TX TY TZ RX RY RZ

axis XYZ

position 0 0 0

orientation 0 0 0



Root representation

:root

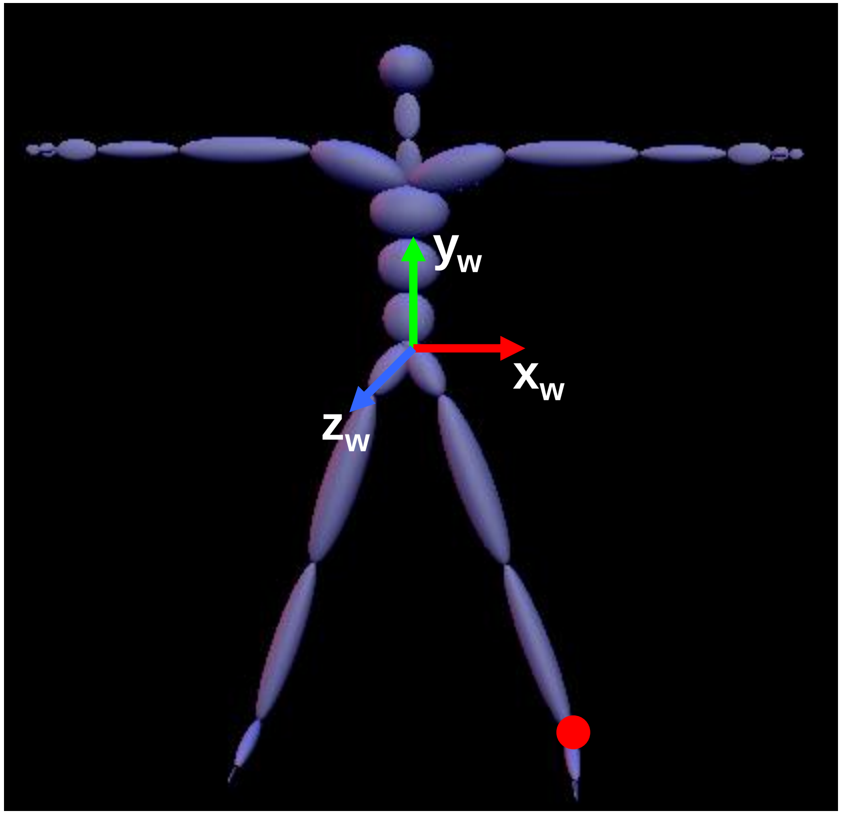
order TX TY TZ RX RY RZ

axis XYZ

position 0 0 0

orientation 0 0 0

How to compute the coordinate of a joint in the world coordinate frame?



Root representation

:root

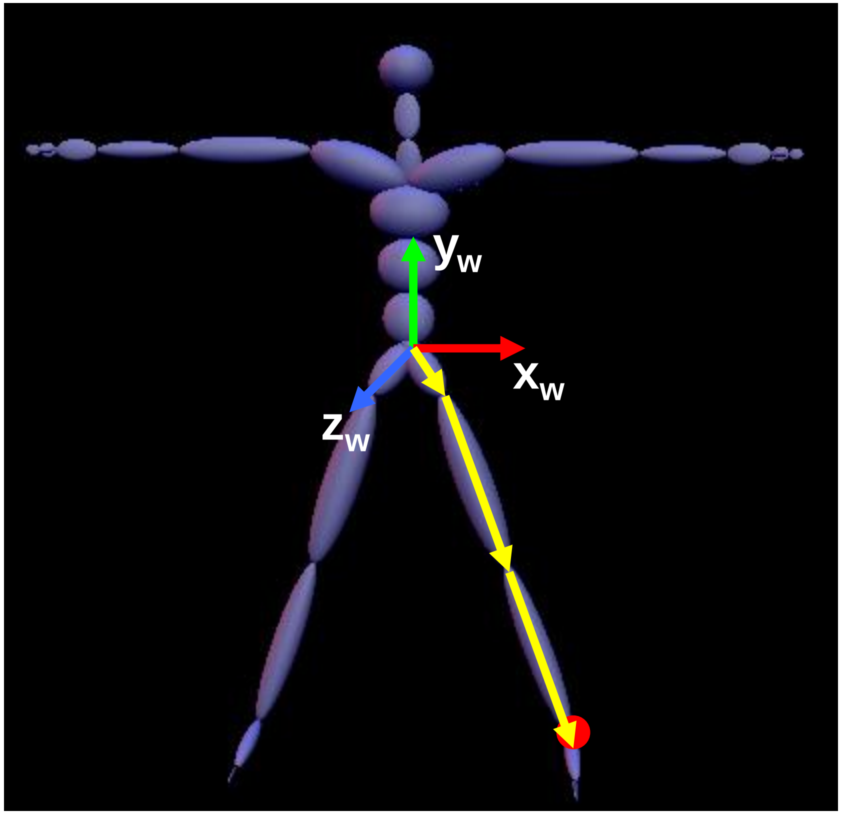
order TX TY TZ RX RY RZ

axis XYZ

position 0 0 0

orientation 0 0 0

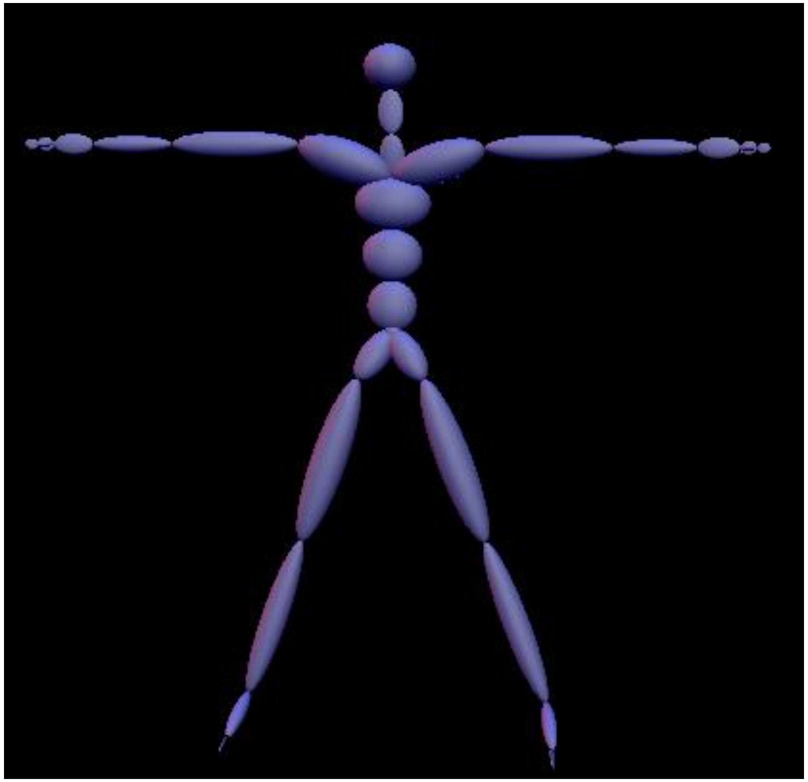
How to compute the coordinate of a joint in the world coordinate frame?



Hierarchy/Bone Connections

:hierarchy

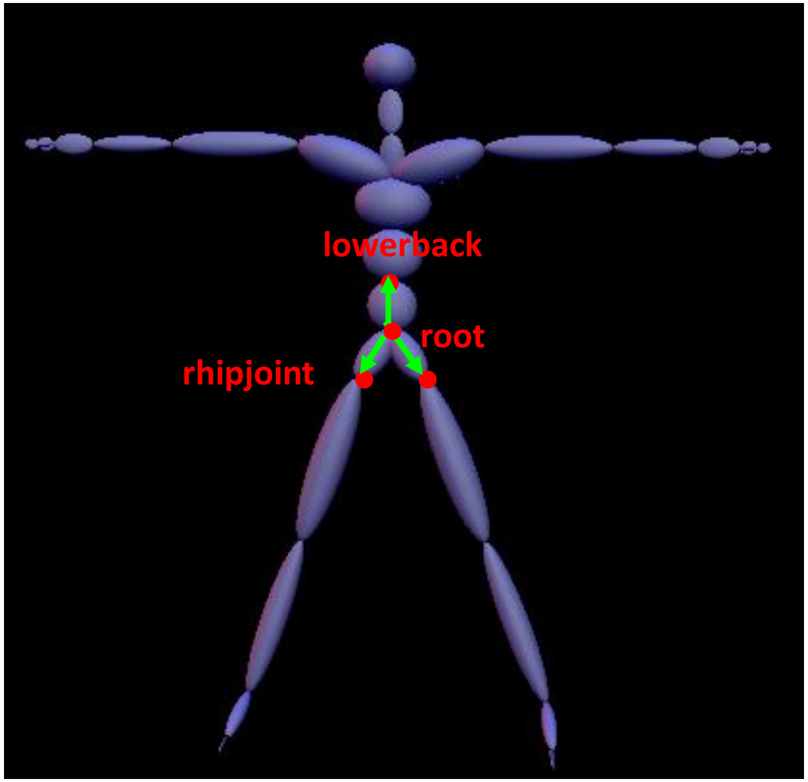
```
begin
 root lhipjoint rhipjoint lowerback
 lhipjoint lfemur
 lfemur ltibia
 ltibia lfoot
 lfoot ltoes
 rhipjoint rfemur
 rfemur rtibia
 rtibia rfoot
 rfoot rtoes
 lowerback upperback
 upperback thorax
 thorax lowerneck lclavicle rclavicle
 …
end
```



Hierarchy/Bone Connections

:hierarchy

begin
root lhipjoint rhipjoint lowerback
lhipjoint lfemur
lfemur ltibia
ltibia lfoot
lfoot ltoes
rhipjoint rfemur
rfemur rtibia
rtibia rfoot
rfoot rtoes
lowerback upperback
upperback thorax
thorax lowerneck lclavicle rclavicle
…
end



Hierarchy/Bone Connections

:hierarchy

begin

root lhipjoint rhipjoint lowerback

lhipjoint lfemur

lfemur ltibia

ltibia lfoot

lfoot ltoes

rhipjoint rfemur

rfemur rtibia

rtibia rfoot

rfoot rtoes

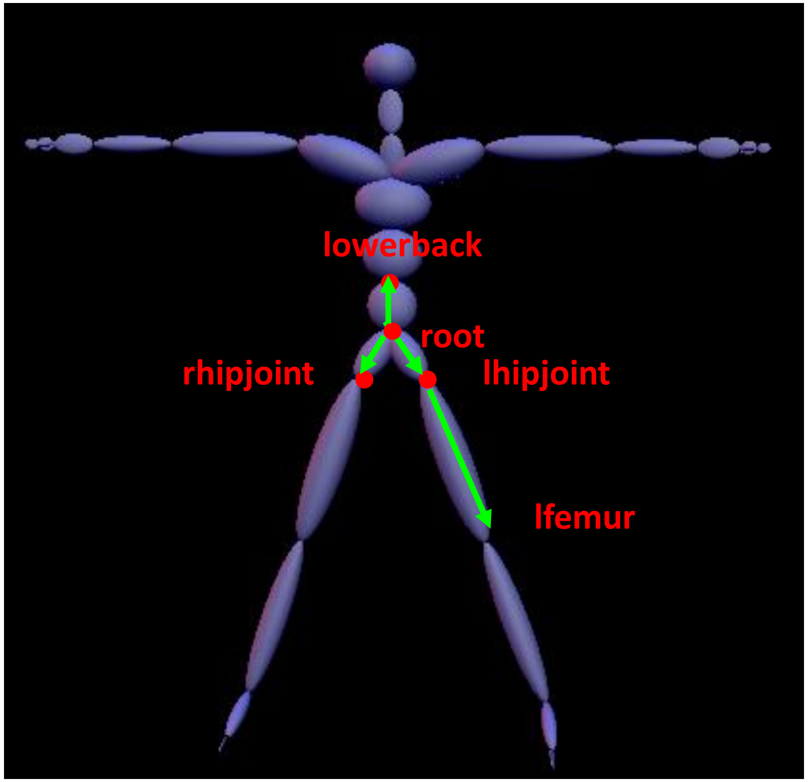
lowerback upperback

upperback thorax

thorax lowerneck lclavicle rclavicle

…

end



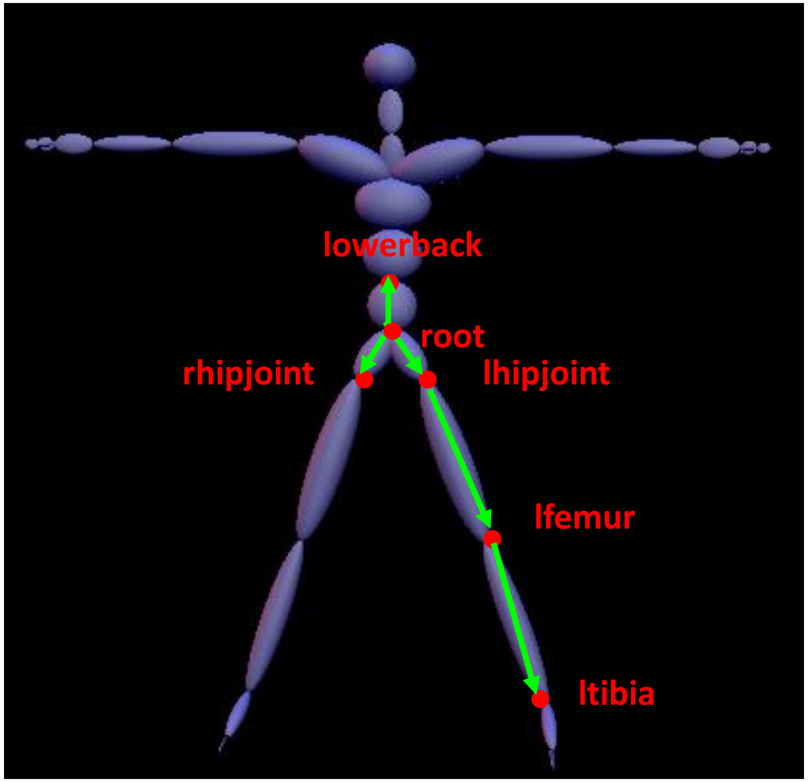
Hierarchy/Bone Connections

:hierarchy

begin

root lhipjoint rhipjoint lowerback
lhipjoint lfemur
lfemur ltibia
ltibia lfoot
lfoot ltoes
rhipjoint rfemur
rfemur rtibia
rtibia rfoot
rfoot rtoes
lowerback upperback
upperback thorax
thorax lowerneck lclavicle rclavicle
…

end



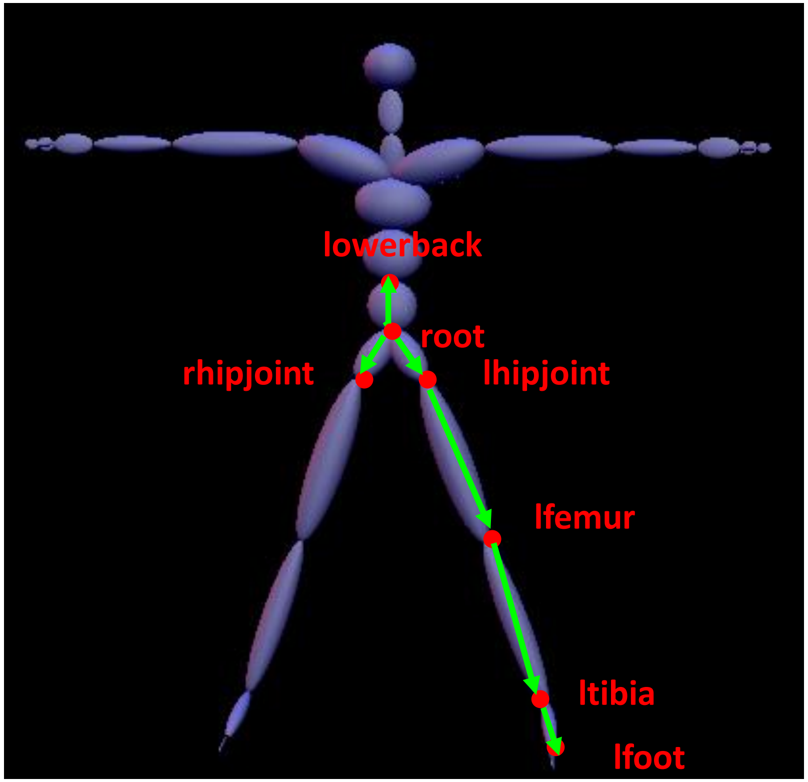
Hierarchy/Bone Connections

:hierarchy

begin

root lhipjoint rhipjoint lowerback
lhipjoint lfemur
lfemur ltibia
ltibia lfoot
lfoot ltoes
rhipjoint rfemur
rfemur rtibia
rtibia rfoot
rfoot rtoes
lowerback upperback
upperback thorax
thorax lowerneck lclavicle rclavicle
…

end



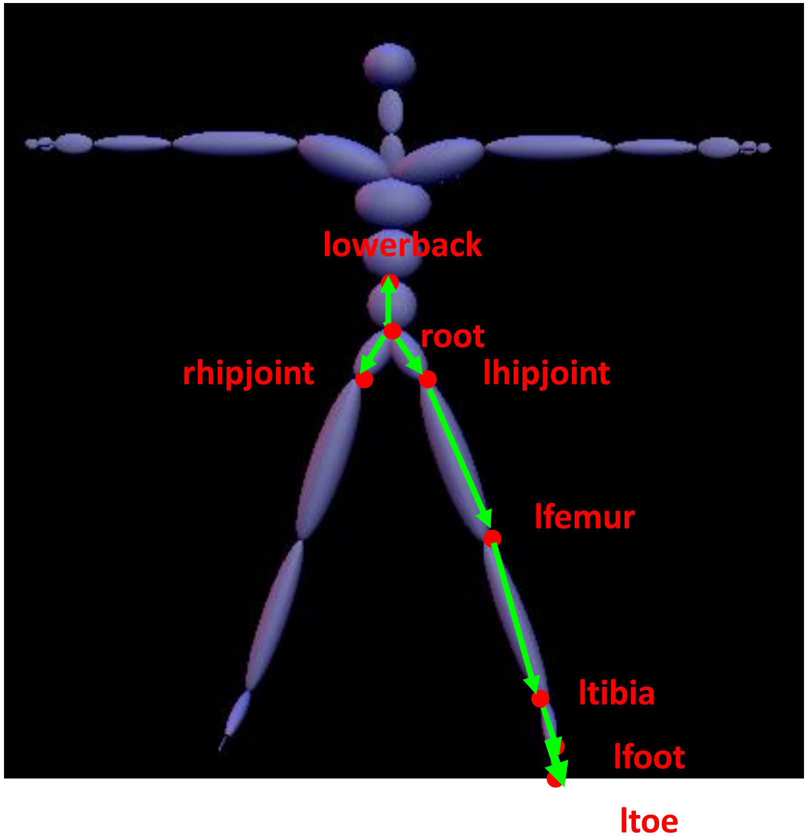
Hierarchy/Bone Connections

:hierarchy

begin

root lhipjoint rhipjoint lowerback
lhipjoint lfemur
lfemur ltibia
ltibia lfoot
lfoot ltoes
rhipjoint rfemur
rfemur rtibia
rtibia rfoot
rfoot rtoes
lowerback upperback
upperback thorax
thorax lowerneck lclavicle rclavicle
…

end



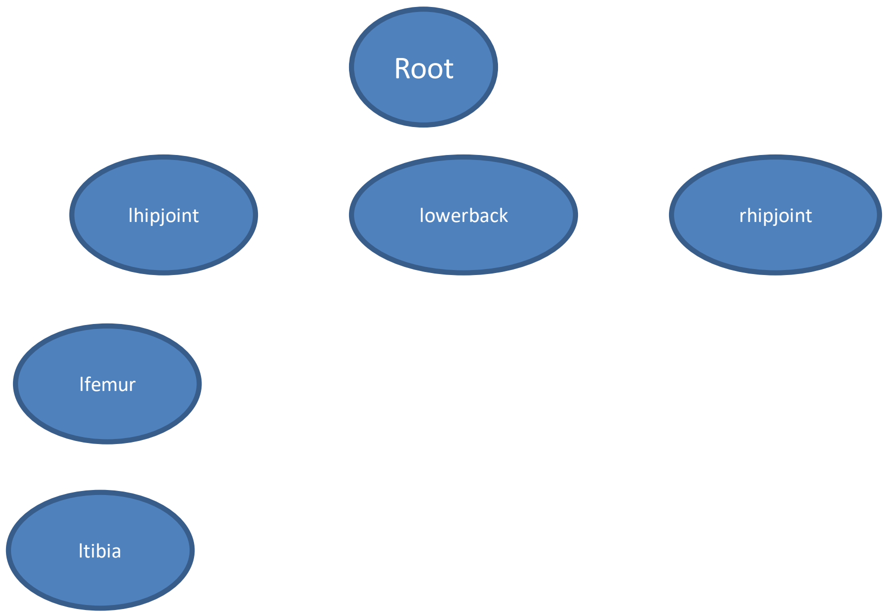
Tree Corresponding to Skeleton File (.asf)

:hierarchy

begin

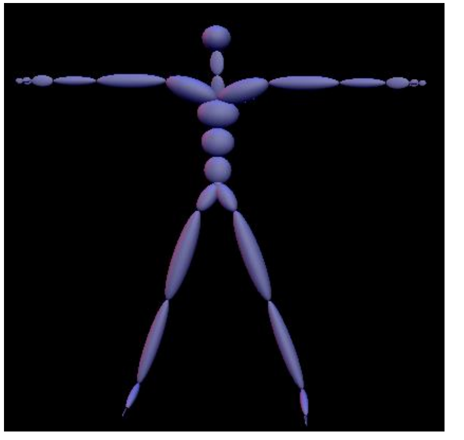
root lhipjoint rhipjoint lowerback
lhipjoint lfemur
lfemur ltibia
ltibia lfoot
lfoot ltoes
rhipjoint rfemur
rfemur rtibia
rtibia rfoot
rfoot rtoes
lowerback upperback
upperback thorax
thorax lowerneck lclavicle rclavicle
…

end



What can we do with .asf file?

We can visualize the default pose

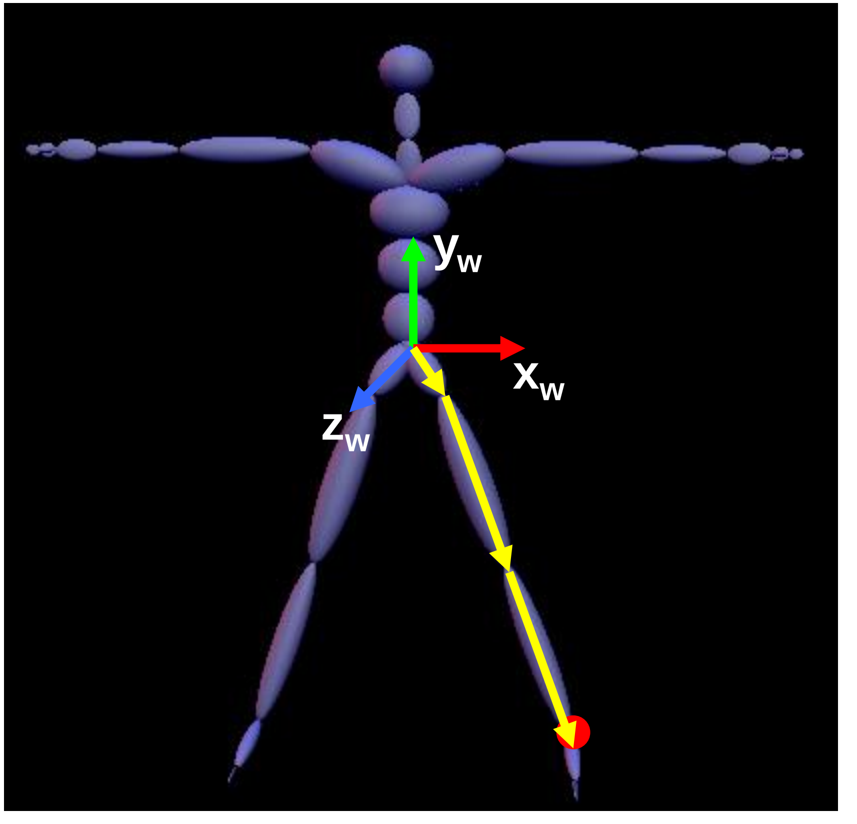


We can compute various transforms in the default pose

- between world coordinate frame and local coordinate
- between parent coordinate frame and child coordinate frame

Forward kinematics

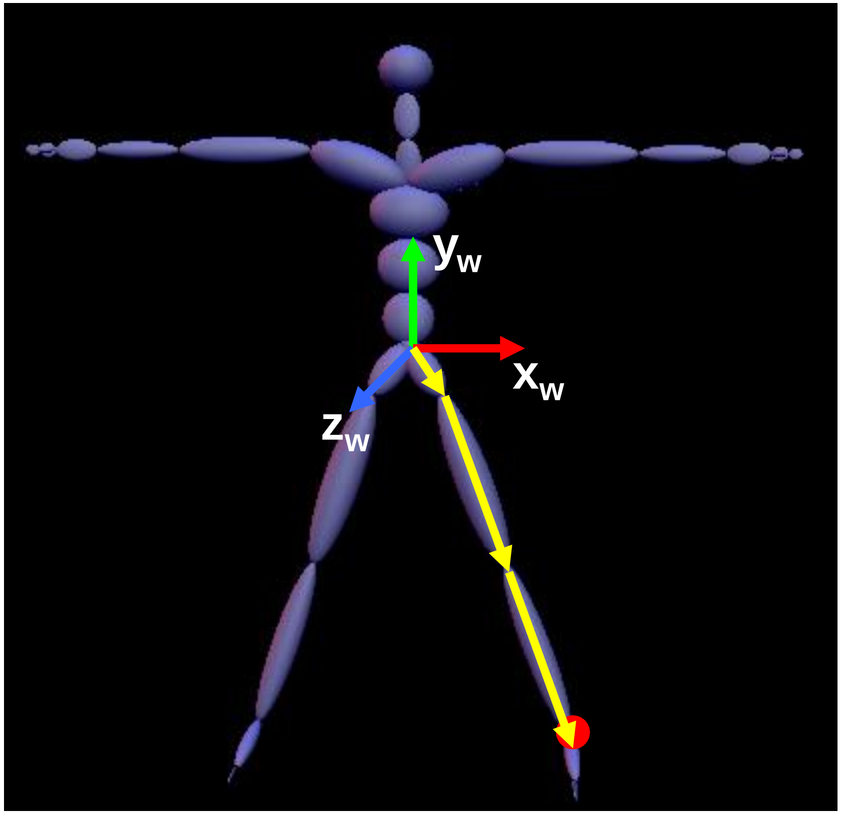
How to compute the coordinate of a joint in the world coordinate frame?



Forward kinematics

How to compute the coordinate of a joint in the world coordinate frame?

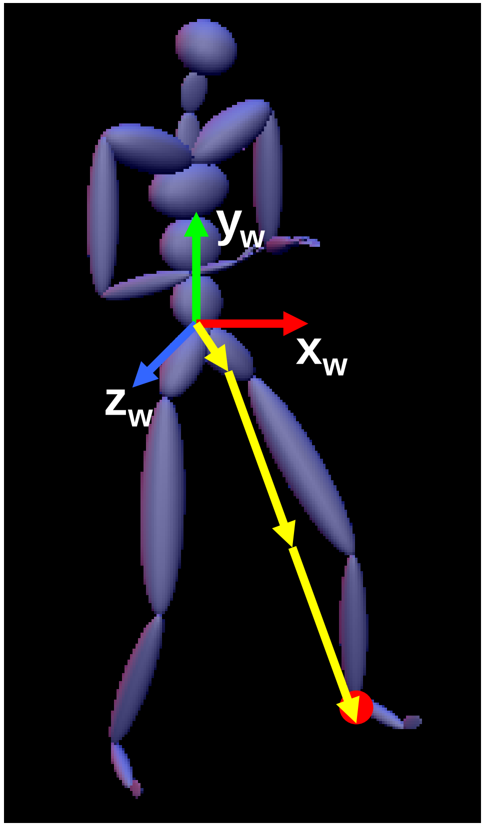
$$X_w = T_1^{w} T_2^{1} \cdots T_k^{k-1} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$



Forward kinematics

How to compute the coordinate of a joint in the world coordinate frame?

$$X_w = T_1^w T_2^1 \cdots T_k^{k-1} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

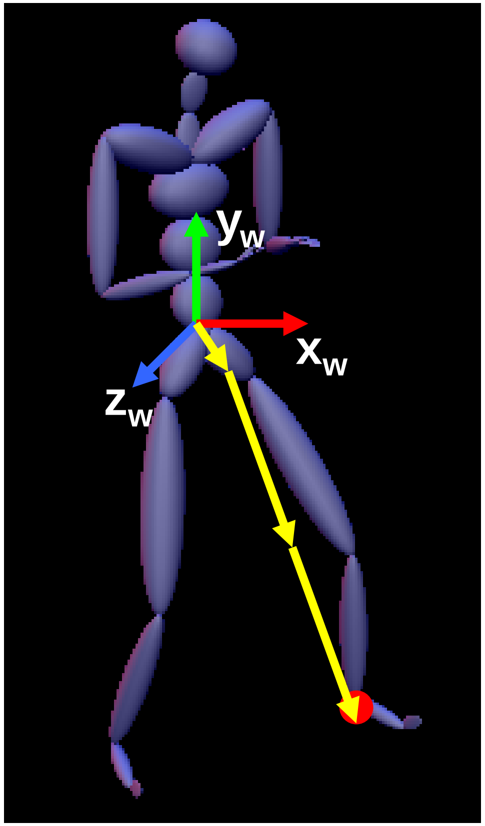


Forward kinematics

How to compute the coordinate of a joint in the world coordinate frame?

$$X_w = T_1^w T_2^1 \cdots T_k^{k-1} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

We need to consider joint angle values!



Motion data file (.amc)

```
i                                                   // frame number
root 2.36756 16.4521 12.3335 -165.118 31.188 -179.889 // root position and orientation
lowerback -17.2981 -0.243065 -1.41128               // joint angles for lowerback joint
upperback 0.421503 -0.161394 2.20925                // joint angles for thorax joint
thorax 10.2185 -0.176777 3.1832
lowerneck -15.0172 -5.84786 -7.55529
upperneck 30.0554 -3.19622 -4.68899
head 12.6247 -2.35554 -0.876544
rclavicle 4.77083e-014 -3.02153e-014
rhumerus -23.3927 30.8588 -91.7324
rradius 108.098
rwrist -35.4375
rhand -5.30059 11.2226
rfingers 7.12502
rthumb 20.5046 -17.7147
lclavicle 4.77083e-014 -3.02153e-014
lhumerus -35.2156 -19.5059 100.612
```

Motion data file (.amc)

i
frame number //

root 2.36756 16.4521 12.3335 -165.118 31.188 -179.889 // root position and orientation

lowerback -17.2981 -0.243065 -1.41128 // joint angles for lowerback joint

upperback 0.421503 -0.161394 2.20925 // joint angles for thorax joint

thorax 10.2185 -0.176777 3.1832

lowerneck -15.0172 -5.84786 -7.55529

upperneck 30.0554 -3.19622 -4.68899

head 12.6247 -2.35554 -0.876544

rclavicle 4.77083e-014 -3.02153e-014

rhumerus -23.3927 30.8588 -91.7324

rradius 108.098

rwrist -35.4375

rhand -5.30059 11.2226

rfingers 7.12502

rthumb 20.5046 -17.7147

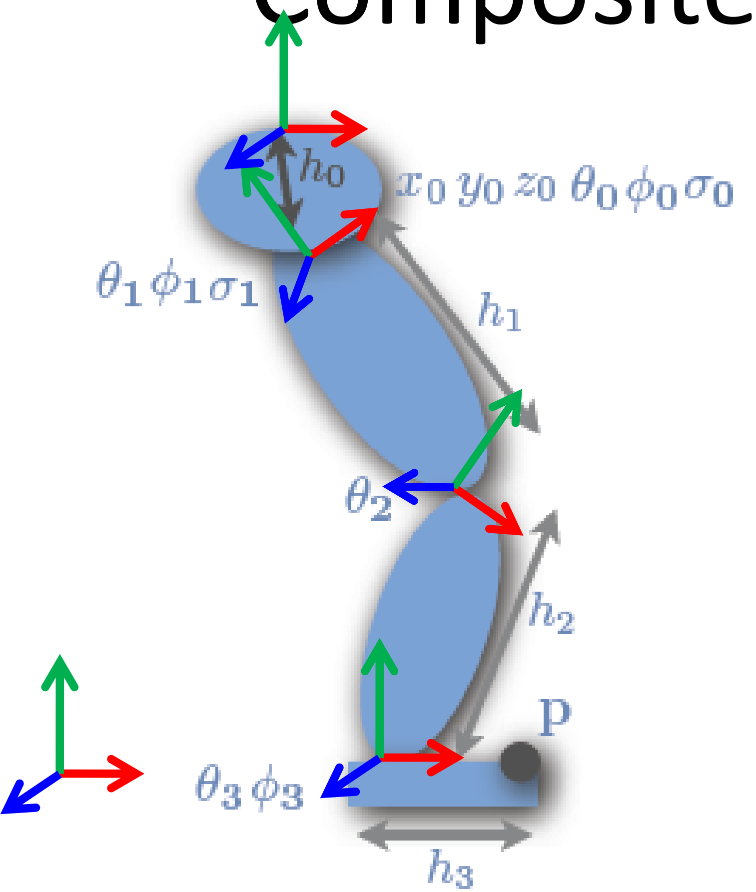
lclavicle 4.77083e-014 -3.02153e-014

lhumerus -35.2156 -19.5059 100.612

- Rotation described in local coordinate frame
- Euler angle representation x-y-z

$$R_z(\delta)R_y(\phi)R_x(\theta)$$

Composite 3D Transformation



A series of transformations on an object can be applied as a series of matrix multiplications

$\mathbf{p}$: position in the global coordinate

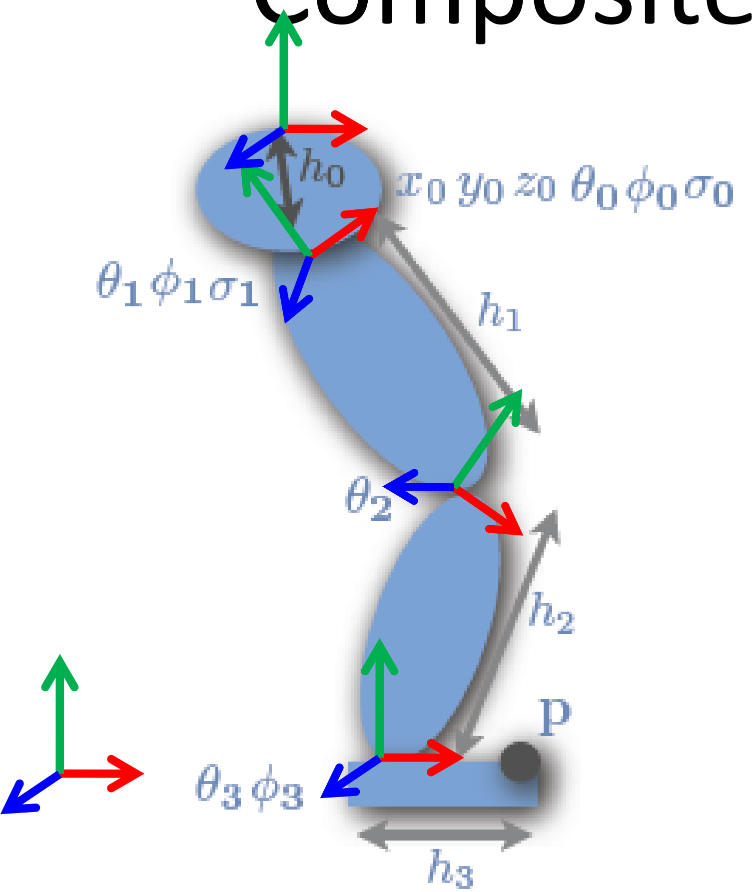
$\mathbf{x}$: position in the local coordinate

$(h_3, 0, 0)$

$$p = T(x_0, y_0, z_0)R(\theta_0)R(\varphi_0)R(\delta_0)\boxed{T_1^0}R(\theta_1)R(\varphi_1)R(\delta_1)\boxed{T_2^1}R(\theta_2)\boxed{T_3^2}R(\theta_3)R(\varphi_3)x$$

From .asf file

Composite 3D Transformation



A series of transformations on an object can be applied as a series of matrix multiplications

$\mathbf{p}$: position in the global coordinate

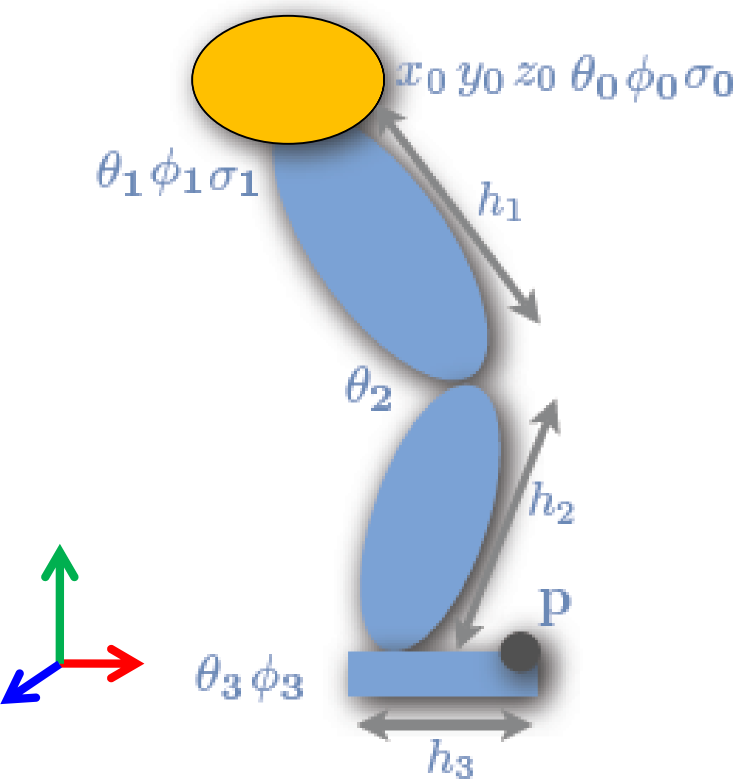
$\mathbf{x}$: position in the local coordinate

$(h_3, 0, 0)$

$$p = \boxed{T(x_0, y_0, z_0)R(\theta_0)R(\varphi_0)R(\delta_0)}\,T_1^0\,\boxed{R(\theta_1)R(\varphi_1)R(\delta_1)}\,T_2^1\,\boxed{R(\theta_2)}\,T_3^2\,\boxed{R(\theta_3)R(\varphi_3)}\,x$$

From .amc file

Composite 3D Transformation



A series of transformations on an object can be applied as a series of matrix multiplications

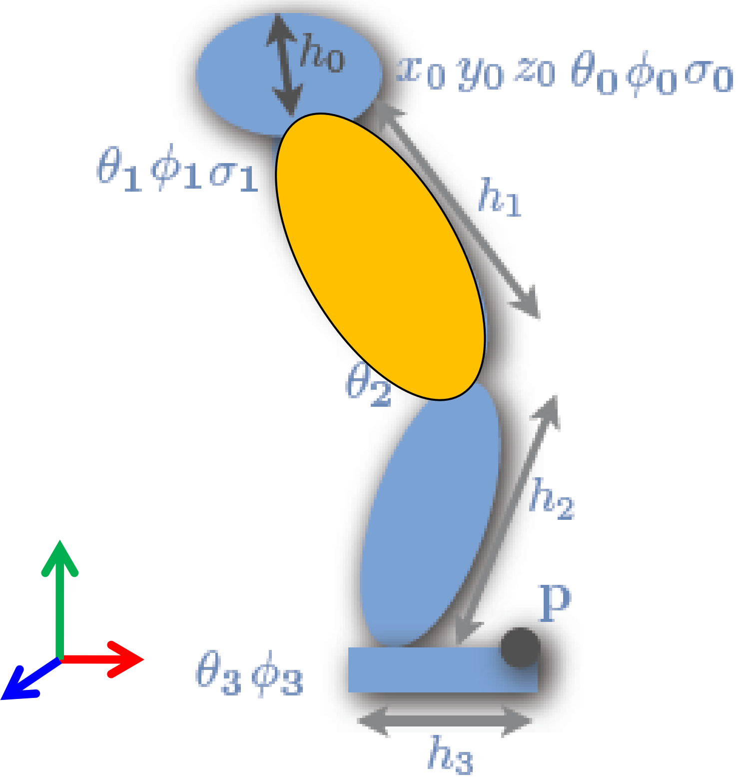
$\mathbf{p}$: position in the global coordinate

$\mathbf{x}$: position in the local coordinate

$(h_3, 0, 0)$

$$p = \boxed{T(x_0, y_0, z_0)R(\theta_0)R(\varphi_0)R(\delta_0)}T_1^0 R(\theta_1)R(\varphi_1)R(\delta_1)T_2^1 R(\theta_2)T_3^2 R(\theta_3)R(\varphi_3)x$$

Composite 3D Transformation



A series of transformations on an object can be applied as a series of matrix multiplications

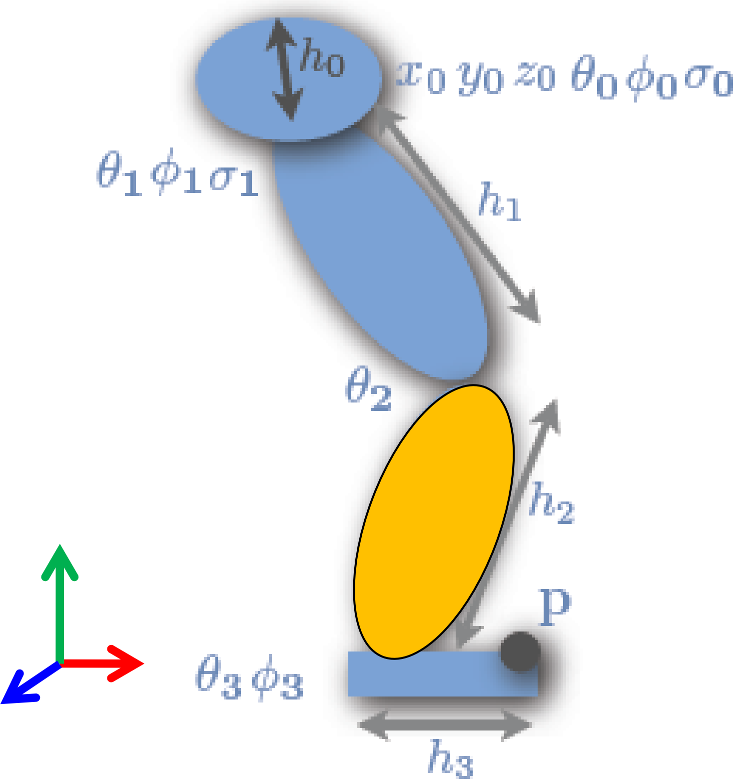
$\mathbf{p}$: position in the global coordinate

$\mathbf{x}$: position in the local coordinate

$(h_3, 0, 0)$

$$p = \boxed{T(x_0, y_0, z_0)R(\theta_0)R(\varphi_0)R(\delta_0)T_1^0 R(\theta_1)R(\varphi_1)R(\delta_1)} T_2^1 R(\theta_2) T_3^2 R(\theta_3) R(\varphi_3) x$$

Composite 3D Transformation



A series of transformations on an object can be applied as a series of matrix multiplications

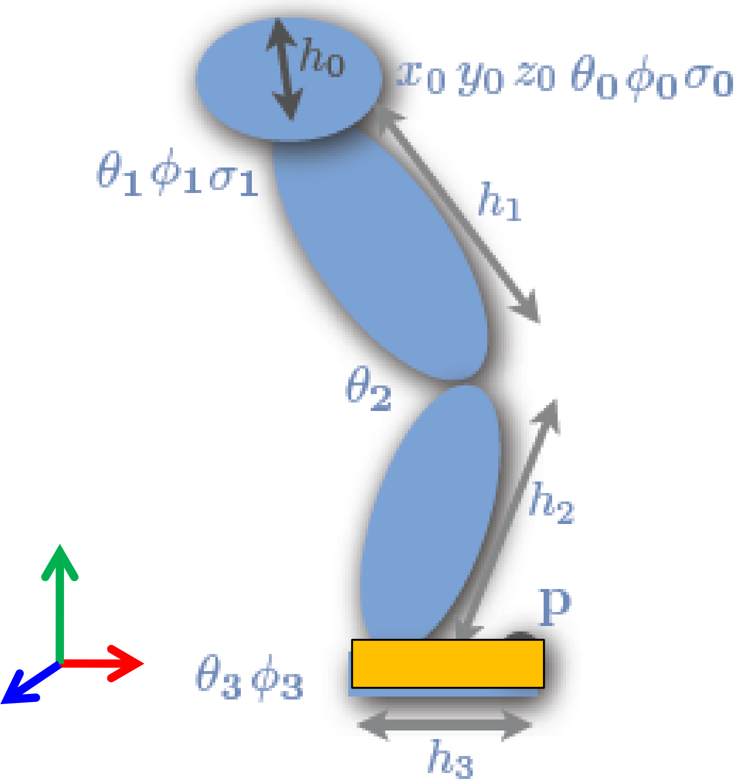
$\mathbf{p}$: position in the global coordinate

$\mathbf{x}$: position in the local coordinate

$(h_3, 0, 0)$

$$p = \boxed{T(x_0, y_0, z_0)R(\theta_0)R(\varphi_0)R(\delta_0)T_1^0 R(\theta_1)R(\varphi_1)R(\delta_1)T_2^1 R(\theta_2)} T_3^2 R(\theta_3)R(\varphi_3)x$$

Composite 3D Transformation



A series of transformations on an object can be applied as a series of matrix multiplications

$\mathbf{p}$: position in the global coordinate

$\mathbf{x}$: position in the local coordinate

$(h_3, 0, 0)$

$$p = \boxed{T(x_0, y_0, z_0)R(\theta_0)R(\varphi_0)R(\delta_0)T_1^0 R(\theta_1)R(\varphi_1)R(\delta_1)T_2^1 R(\theta_2)T_3^2 R(\theta_3)R(\varphi_3)}\, x$$

Outline

- What is Motion Capture (MoCap)?
- MoCap Technology
- Data Representation
- Using MoCap (Mesh Skinning)
- MoCap Compression

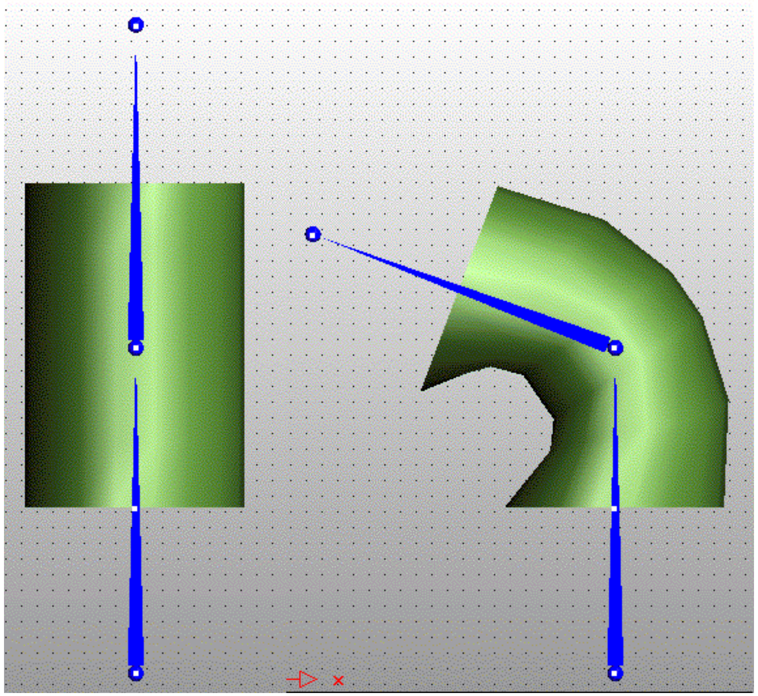
Some Character Models

Mesh skinning

- Skinning is the process of binding a skeleton to a single mesh object
- Skinning deformation is the process of deforming the mesh as the skeleton is animated or moved.

Mesh skinning

- Cylinder Being Deformed by Two Bones



Skinning basics

- For each vertex, compute the position by

$$v' = \sum_{i=1}^{N} w_i M_i v$$

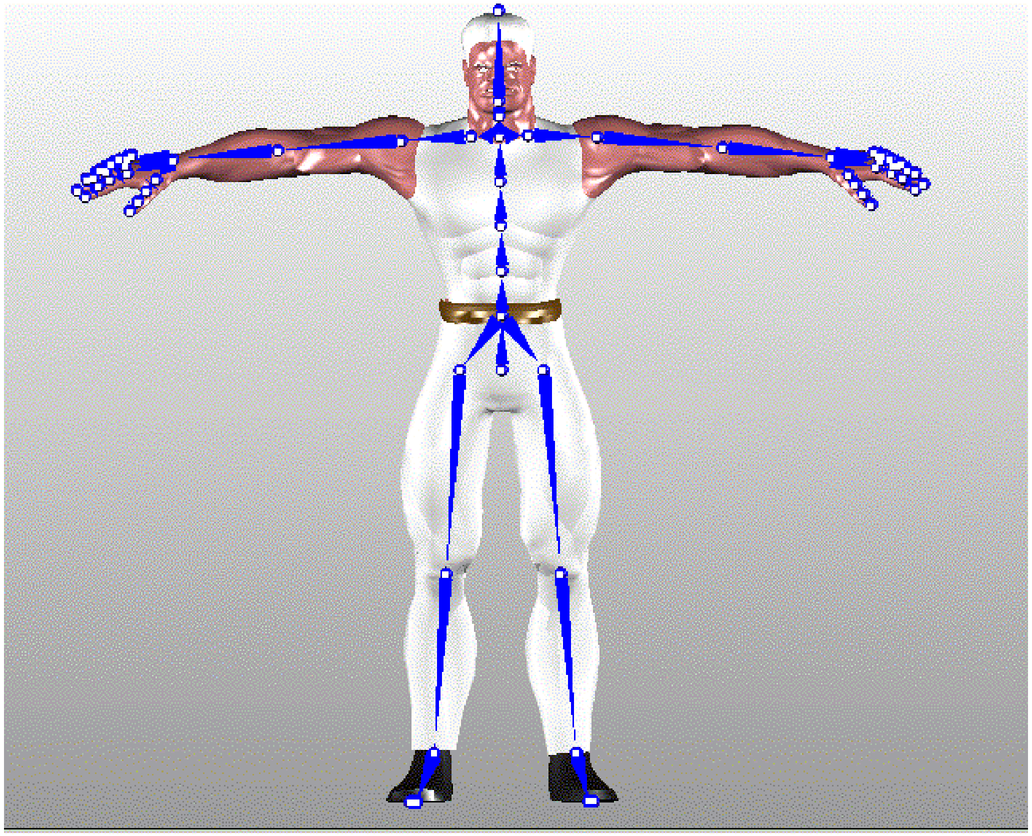
From mesh model → $\boxed{v:}$ undeformed vertex position

v': deformed vertex position

From mocap data → $\boxed{M_i:}$ articulated motion

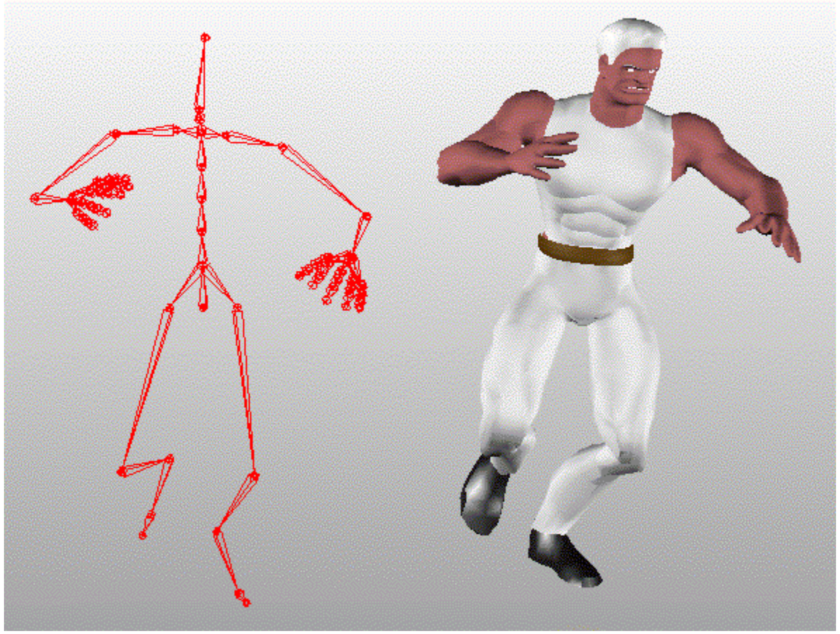
Specified by artists → $\boxed{w_i:}$ blending weight $\sum_{i=1}^{N} w_i = 1$

The "Bind Pose"



Mesh skinning

- Skeleton causing deformation of a single skin mesh



Pinnocchio: Automatically Embedding a Skeleton into a Mesh

There are various techniques that try to automatically place a skeleton inside a given mesh WITHOUT any Human Intervention.

All these techniques have their advantages and limitations. Thus, even now Human animation experts adjusting things following an initial automatic embedding process is the best alternative for creating visually pleasing animations.

https://www.youtube.com/watch?v=EklzamltEgM

Outline

- What is Motion Capture (MoCap)?
- MoCap Technology
- Data Representation
- Using MoCap (Mesh Skinning)
- MoCap Compression

Why and How

- Why: To save bandwidth
  - Especially when multiple characters present in the scene
- How: Exploiting redundancy
  - Temporal correlation
    - Example: Curve fitting (Bezier curves), wavelet coding, DCT
  - Precision-> Quantization
  - Entropy coding (e.g. Huffman)
  - Spatial correlation (relation between joints)-> Clustering, PCA

Important Factors

- Compression ratio
  - Example: A 1 MB motion file compressed 20 KB -> Compression ratio= 1000/20= 50:1
- Processing time
  - 60 to 120 frames per second (fps)
  - Compression and decompression time
- Delay
  - 16 frames at 60 fps -> about 0.25s delay

PCA Based Approach

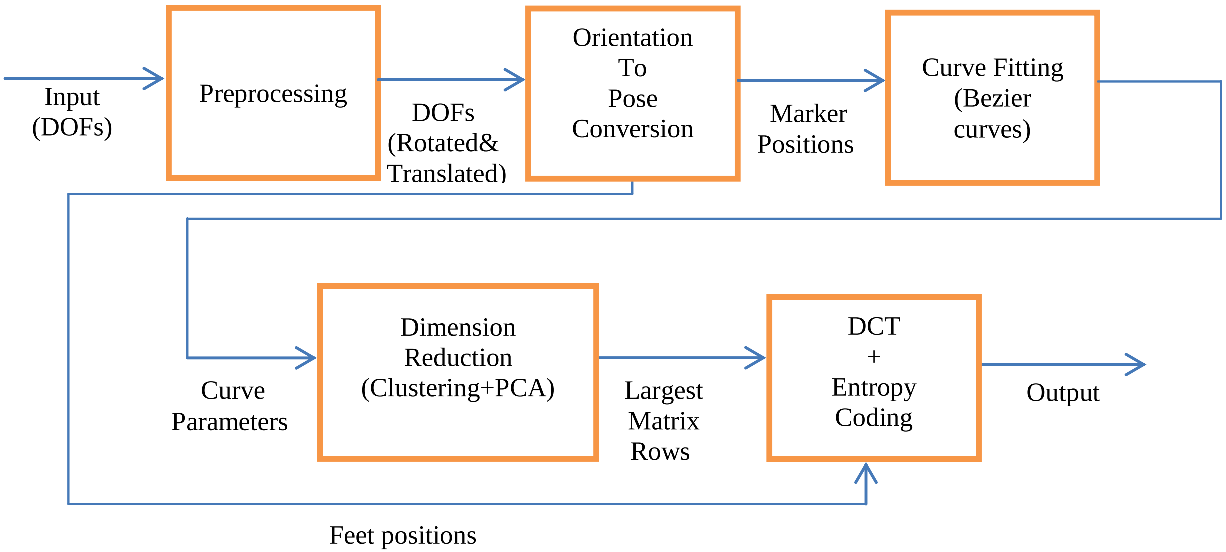


Fig. 2. PCA Based Approach.

PCA

- **Principal component analysis** (**PCA**) is a statistical procedure that uses orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called **principal components**.
- The number of principal components is less than or equal to the number of original variables.
- This transformation is defined in such a way that the first principal component has the largest possible variance (that is, accounts for as much of the variability in the data as possible), and each succeeding component in turn has the highest variance possible under the constraint that it be orthogonal to (i.e., uncorrelated with) the preceeding components.

PCA Based Approach

- Compression ratio of 25:1 to 30:1 without any noticeable visual degradation
- Cons
  - Overhead (convert joint position to joint angle and vice versa)
  - High compression ratios only when applied to a large motion database (not suitable for individual clips)

Wavelet Based Approach

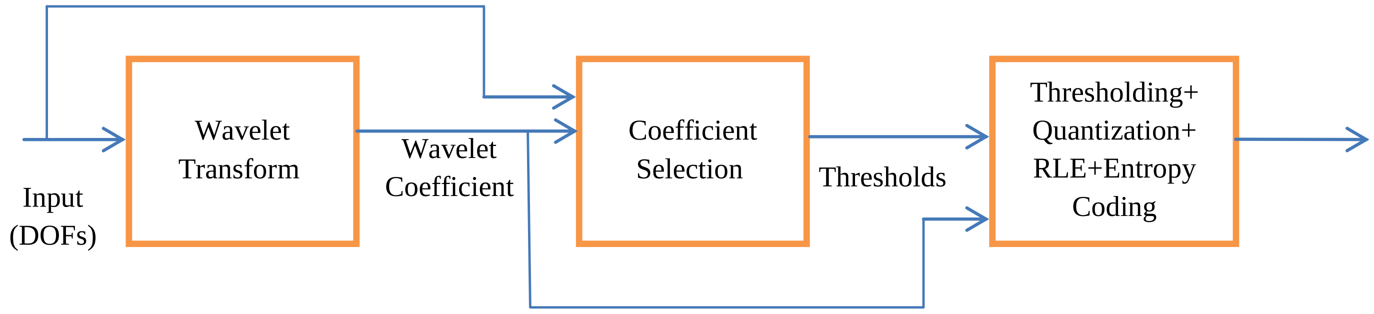


Fig. 3. Wavelet Based Approach.

Wavelet Based Approach

- Compression ratio of 25:1 (for short motions) to 35:1(for longer motions) can be achieved without any noticeable visual degradation
- Cons
  - Compression very time consuming (133-327 ms per frame on a 2GHz AMD Athlon 64 bits) because of global optimization

Perceptually Motivated Compression

- In image and video
  - People are sensitive to error in certain parts of the image
- In Motion
  - People might be sensitive to error in certain joints

Perceptually Motivated MoCap Compression

- Bone Length
  - Errors in Coding for Longer Bones have greater visual impact

- Joint in Contact with a Surface
  - Errors in location of the feet is more easily visible
  - Looks like “Skating” if there is foot slippage
  - Which joint is more important when writing on a white board?

Effect of Error on Bones

- Long Bone

  A small error in angular rotation is more visible

Effect of Error on Bones

- Short Bone

  A small error is angular rotation is less visible

MoCap Compression by Segment Replacement

- Approach: Using a motion primitive database
  - Facilitates the exploitation of temporal redundancies
  - Especially in repetitive and/or rhythmic motion

- Divide the compression process into offline and online phases
  - Offline phase: Create a reference primitive database based on a set of training motion data
  - Online phase: Compress the motion segment by segment.
    - For each segment transmit the reference to the closest primitive in the database

Our Previous Work in IEEE TMM

- IEEE TMM 2011: “Perceptually guided fast compression of 3-D motion capture data”
- Compression of motion capture data using perceptual factors
- Requirements:
  - (1) Encoding and decoding should take a small portion of the processing resources (not just real time),
  - (2) Incorporating human perceptual factors of animation into the technique.
- Based on wavelet coding
- Basic idea: select different numbers of coefficients for different channels of data based on the importance of each channel on the perceived quality of motion.
- Optimize the coefficient selection algorithm based on two factors.
  - The length of the bone connected to a joint.
  - Variations in each DOF during the whole animation

Proposed Method

1. Divide the animation into M slices, having W frames in each slice (with the exception in the last slice, which can contain less than W frames).
2. For each slice m_k $(1 \le k \le M)$ perform Steps 3 to 9:
3. Since the animated characters can be of different sizes, we use the normalized values (or relative measurement) to provide a comparable basis. Thus, for each channel we obtain the normalized bone length (l_i) by dividing the bone length attached to the corresponding joint by the maximum bone length in the skeleton.
4. Find the variation of each channel (c_i) using the following formula:

$$v_i = \frac{\sum_{p=m_i*W}^{m_i*W+W-1} \left| c_{i,p+1} - c_{i,p} \right|}{W} \tag{1}$$

5. Combine the effects of relative bone length and variation to obtain a weighted quantity (A_i) representing the relative importance (perceptual impact) of each channel.

$$A_i = l_i v_i \tag{2}$$

6. For each channel compute $K_i = \max(\frac{\alpha A_i NW}{\sum_{j=1}^{N} A_j}, C)$, the largest (in absolute value) wavelet coefficient, and set other coefficients to zero (where C is a constant representing the minimum number of wavelet coefficients kept for each channel).
7. Quantize the remaining wavelet coefficients linearly into QP bits.
8. Use run-length to encode the wavelet coefficients.
9. Use an entropy coding method to compress the data.

Results

- Carnegie Mellon University ASF/ASM viewer for user studies.
- Input format: IEEE 32 bit (4 bytes) floats precision, where every 4 bytes represent one DOF in a frame.
- 20 bones, 56 DOF.
- Motion Sequences:
  - 2_2: Simple walking motion
  - 49_14: taichi
- Three methods:
  - The proposed method with both bone lengths and variation in the degrees of freedom being taken into account;
  - The proposed method but only bone lengths being taken into account;
  - Standard wavelet compression (setting the same percentage of coefficients to zero for all the channels).

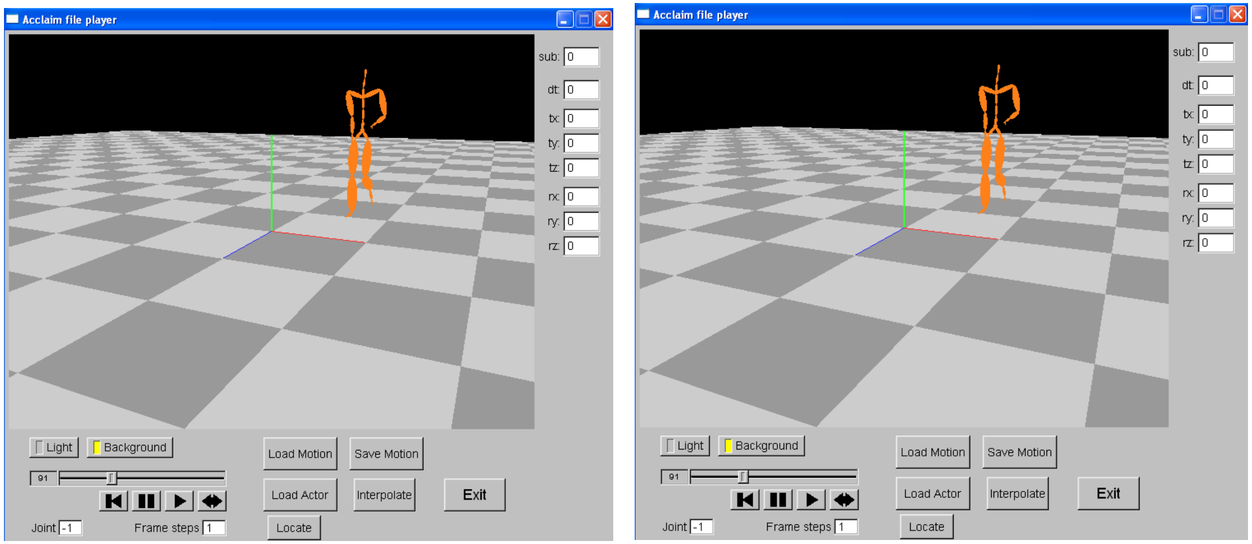


Fig. 4. Motion capture viewer environment.

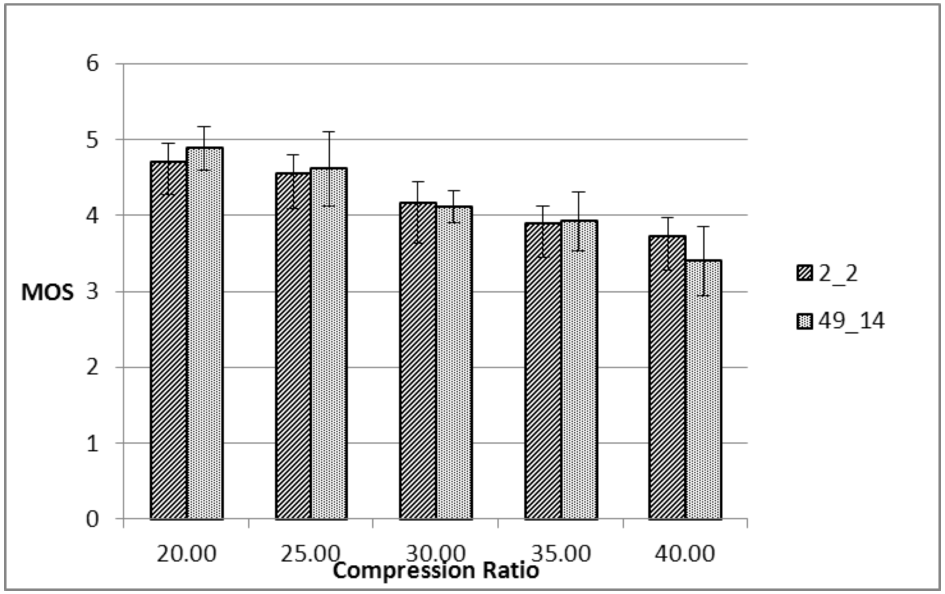


Fig. 5. Results of user study on two of CMU database motion files.

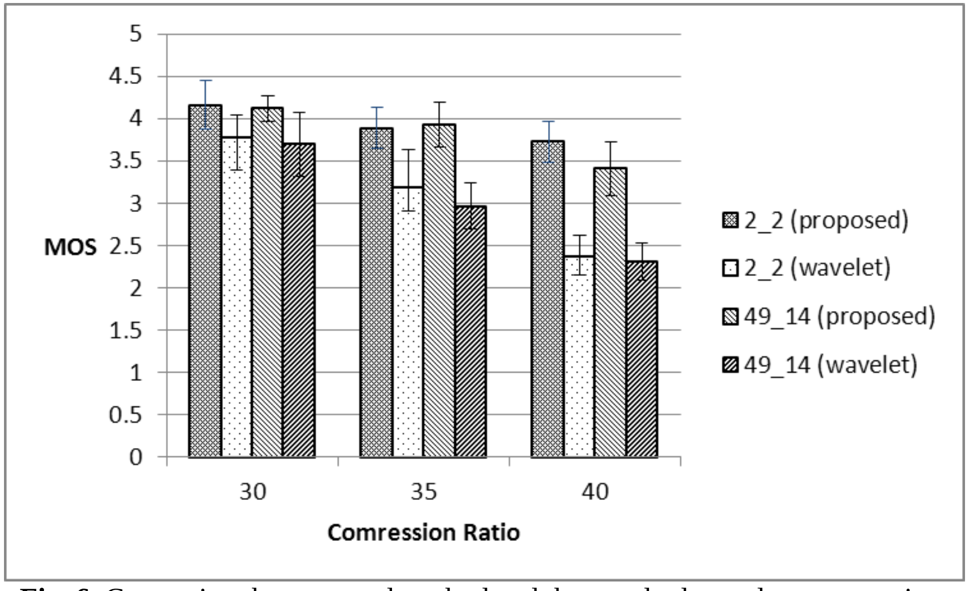


Fig. 6. Comparing the proposed method and the standard wavelet compression.

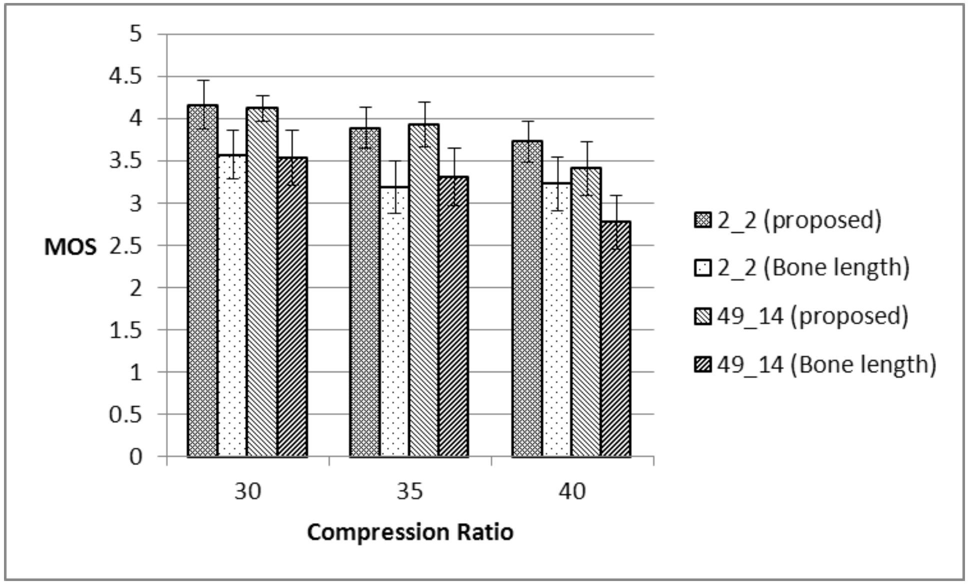


Fig. 7. Verifying the efficiency of selected perceptual factors.

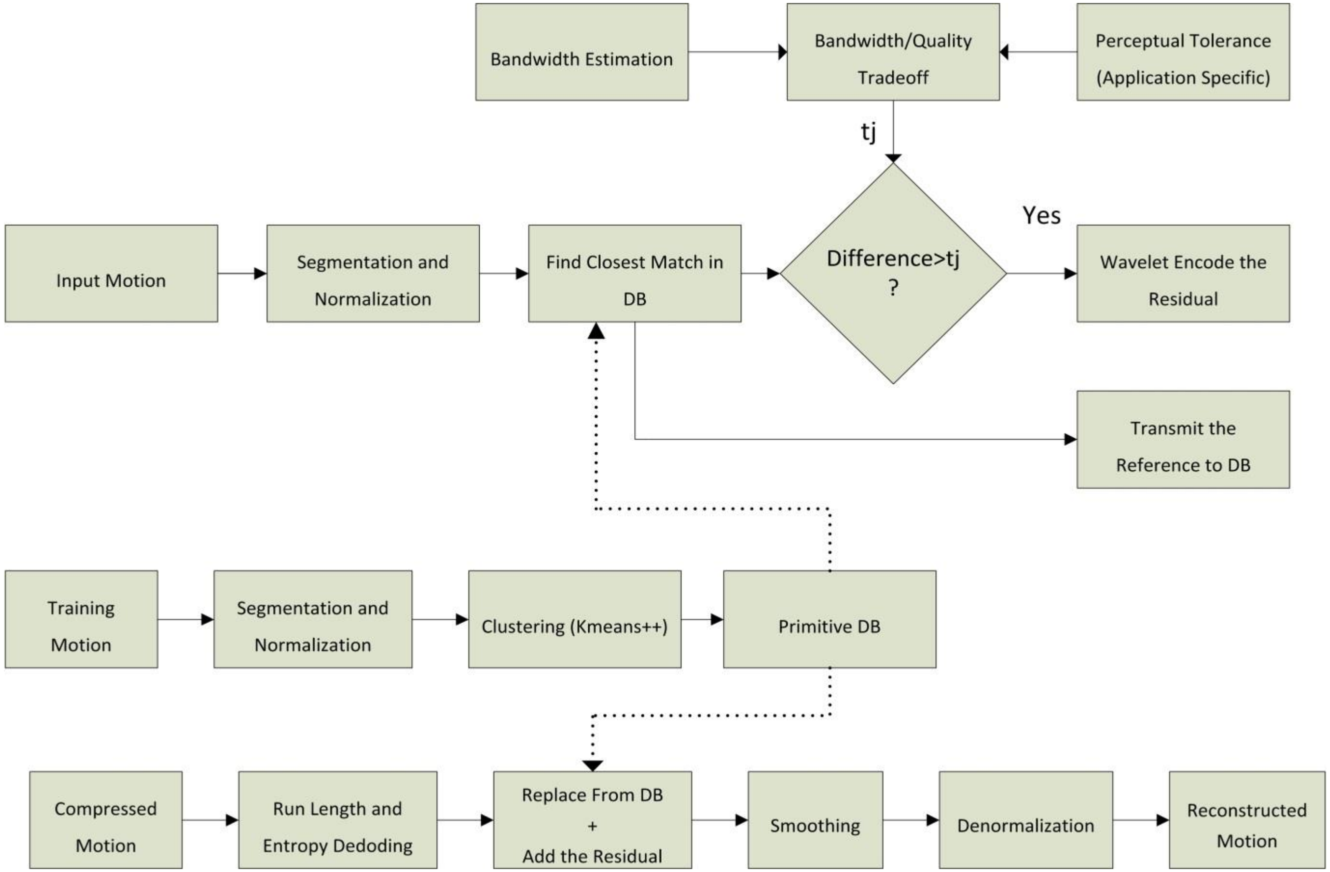
Table I: Compression Times (times are in μs).

Motion File	No. of frames	Total Time	Per Frame
14_9	3287	79687	24.2
1_4	4298	107500	25.01
25_9	5147	131250	25.5

Our Previous Work (Approach 2)

- Approach: Using a motion primitive database
  - Facilitates the exploitation of temporal redundancies
  - Especially in repetitive and/or rhythmic motion
- Divide the compression process into offline and online phases
  - Offline phase: Create a reference primitive database based on a set of training motion data
  - Online phase: Compress the motion segment by segment.
    - For each segment transmit the reference to the closest primitive in the database

Outline of the Method



Error Threshold

- Compression: Replace each segment with its closest database primitive
  - If the error is above a threshold (t_j) encode the difference between the primitive and current segments using wavelet transform.
  - Theoretically, this threshold (t_j) is defined separately for each joint
  - In practice, we focus on the joints which generate visually significant impacts
  - Define two thresholds:
    - t_L for lower body joints
    - t_U for upper body joints
    - Set t_U as 2 times t_L (based our observations)

User Study

- A series of comparisons where motions were displayed side by side
- started with a high distortion error value of t_L (0.024)
- decreased t_L by 0.002 whenever the participant chose the correct (original) motion twice in a row.
  - For a wrong answer, t_L was increased by 0.002.
- Converging point: Average of the last three reversals points
- Average converging threshold: $t_L = 0.012$
- Smallest: $t_L = 0.01$
- LargestL $t_L = 0.016$

Results

- Motions from CMU database (mocap.cs.cmu.edu)

94_2 61_4 79_17 79_44 135_4 138_2 90_2

- Compression Ratio

Motion	94_2	61_4	79_17	79_44	135_4	138_2	90_2
CR	62:1	51:1	52:1	50:1	40:1	44:1	38:1

Results (Cont.)

- *Compression ratio (CR) of the current methods reported in the literature*

Method	[Ari06]	[BPV07]	[GPD09]	gZip
CR	30:1	48:1	25:1	3.3:1

- *The Compression ratio can be increased or decreased by adjusting a single parameter value t_L.*