Human Motion Analysis
Lecture 7: Human pose estimation

Raquel Urtasun

TTI Chicago

April 26, 2010

Materials used for this lecture

- L. Herda, R. Urtasun and P. Fua. Hierarchical Implicit Surface Joint Limits for Human Body Tracking. In Computer Vision and Image Understanding, (CVIU) 2005.
- L. Herda, R. Urtasun, P. Fua, A. Hanson. Automatic Determination of Shoulder Joint Limits using Quaternion Field Boundaries. International Journal of Robotics Research (IJRR), 22(6): 419 - 436, 2003.

Contents of today's lecture?

We will look into:

- Generative approaches to pose estimation
- We will focus on joint limit priors

Notation

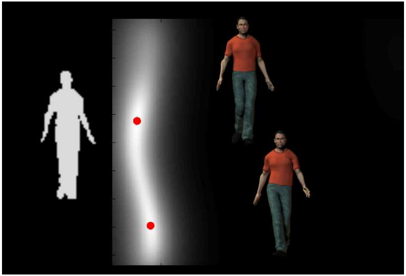
ϕ — the state
$\mathbf{I}$ – the image
$\mathbf{x}$ – the latent representation
N — number of training samples
$\mathbf{I}_{n:0}$ — image observations up to time n
$\mathbf{y}_{n:0}$ — poses up to time n

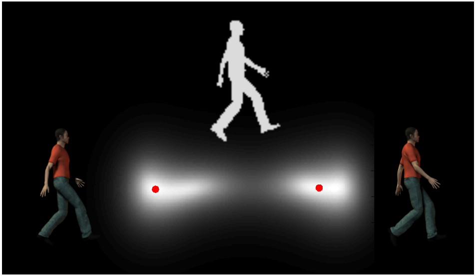
The problem of human pose estimation

- The goal is given an image $\mathbf{I}$ to estimate the 3D location and orientation of the body parts $\mathbf{y}$.

Challenges of pose estimation

- Poor imaging: motion blurred, occlusions, cluttered, etc.
- Non-convex problem: multimodal solutions





Pose estimation

- **Generative approaches:** focus on modeling

$$p(\phi|\mathbf{I}) = \frac{p(\mathbf{I}|\phi)p(\phi)}{p(\mathbf{I})}$$

- **Discriminative approaches:** focus on modeling directly

$$p(\phi|\mathbf{I})$$

Today we will talk about generative approaches.
Later in the class we will cover discriminative approaches.

Generative approaches

Generative approach models

$$p(\phi|\mathbf{I}) = \frac{p(\mathbf{I}|\phi)p(\phi)}{p(\mathbf{I})}$$

Types of generative approaches:

- **Bayesian approaches:** focus on approximating $p(\phi|\mathbf{I})$, usually via sampling (e.g., particle filter).
- **Optimization or energy-based techniques:** focus on computing the MAP or ML estimate of $p(\phi|\mathbf{I})$.

Generative approaches

Generative approach models

$$p(\phi|\mathbf{I}) = \frac{p(\mathbf{I}|\phi)p(\phi)}{p(\mathbf{I})}$$

Types of generative approaches:

- **Bayesian approaches:** focus on approximating $p(\phi|\mathbf{I})$, usually via sampling (e.g., particle filter).
- **Optimization or energy-based techniques:** focus on computing the MAP or ML estimate of $p(\phi|\mathbf{I})$.

Common to all of them is the need to model

- **Image likelihood:** $p(\mathbf{I}|\phi)$
- **Priors:** $p(\phi)$

Generative approaches

Generative approach models

$$p(\phi|\mathbf{I}) = \frac{p(\mathbf{I}|\phi)p(\phi)}{p(\mathbf{I})}$$

Types of generative approaches:

- **Bayesian approaches:** focus on approximating $p(\phi|\mathbf{I})$, usually via sampling (e.g., particle filter).
- **Optimization or energy-based techniques:** focus on computing the MAP or ML estimate of $p(\phi|\mathbf{I})$.

Common to all of them is the need to model

- **Image likelihood:** $p(\mathbf{I}|\phi)$
- **Priors:** $p(\phi)$

In general $p(\mathbf{I})$ is assumed constant and ignored. The different trackers then depend on the different modeling choices and optimization procedures.

Generative approaches

Generative approach models

$$p(\phi|\mathbf{I}) = \frac{p(\mathbf{I}|\phi)p(\phi)}{p(\mathbf{I})}$$

Types of generative approaches:

- **Bayesian approaches:** focus on approximating $p(\phi|\mathbf{I})$, usually via sampling (e.g., particle filter).
- **Optimization or energy-based techniques:** focus on computing the MAP or ML estimate of $p(\phi|\mathbf{I})$.

Common to all of them is the need to model

- **Image likelihood:** $p(\mathbf{I}|\phi)$
- **Priors:** $p(\phi)$

In general $p(\mathbf{I})$ is assumed constant and ignored. The different trackers then depend on the different modeling choices and optimization procedures.

Particle filter revisited

The posterior density is described with three terms

$$p(\phi_n|\mathbf{I}_{n:0}) = \frac{p(\mathbf{I}_n|\phi_n)p(\phi_n|\mathbf{I}_{n-1:0})}{p(\mathbf{I}_n|\mathbf{I}_{n-1:0})}$$

- **Prior:** defines the knowledge of the model

$$p(\phi_n|\mathbf{I}_{n-1:0}) = \int p(\phi_n|\phi_{n-1})p(\phi_{n-1}|\mathbf{I}_{n-1:0})d\phi_{n-1}$$

Particle filter revisited

The posterior density is described with three terms

$$p(\phi_n|\mathbf{I}_{n:0}) = \frac{p(\mathbf{I}_n|\phi_n)p(\phi_n|\mathbf{I}_{n-1:0})}{p(\mathbf{I}_n|\mathbf{I}_{n-1:0})}$$

- **Prior:** defines the knowledge of the model

$$p(\phi_n|\mathbf{I}_{n-1:0}) = \int p(\phi_n|\phi_{n-1})p(\phi_{n-1}|\mathbf{I}_{n-1:0})d\phi_{n-1}$$

- **Likelihood:** $p(\mathbf{I}_n|\phi_n)$ determines the measurement noise model
- **Evidence:** which involves

$$p(\mathbf{I}_n|\mathbf{I}_{n-1:0}) = \int p(\mathbf{I}_n|\phi_n)p(\phi_n|\mathbf{I}_{n-1:0})d\phi_n$$

Particle filter revisited

The posterior density is described with three terms

$$p(\phi_n|\mathbf{I}_{n:0}) = \frac{p(\mathbf{I}_n|\phi_n)p(\phi_n|\mathbf{I}_{n-1:0})}{p(\mathbf{I}_n|\mathbf{I}_{n-1:0})}$$

- **Prior:** defines the knowledge of the model

$$p(\phi_n|\mathbf{I}_{n-1:0}) = \int p(\phi_n|\phi_{n-1})p(\phi_{n-1}|\mathbf{I}_{n-1:0})d\phi_{n-1}$$

- **Likelihood:** $p(\mathbf{I}_n|\phi_n)$ determines the measurement noise model
- **Evidence:** which involves

$$p(\mathbf{I}_n|\mathbf{I}_{n-1:0}) = \int p(\mathbf{I}_n|\phi_n)p(\phi_n|\mathbf{I}_{n-1:0})d\phi_n$$

Optimization techniques

It is defined as minimizing the following programs:

$$\phi^*_{ML} = \underset{\phi}{\operatorname{argmin}} - \log p(\mathbf{I}|\phi)$$

$$\phi^*_{MAP} = \underset{\phi}{\operatorname{argmin}} - \log p(\mathbf{I}|\phi) - \log p(\phi)$$

It suffers from the following problems:

- Local minima: usually $-\log p(\mathbf{I}|\phi)$ is a non-convex function of ϕ.
- Initialization: usually hand initialized or use discriminative approaches.
- Drift: As times goes, the estimate gets worst.
- Difficult to define a good general $-\log p(\mathbf{I}|\phi)$.

Optimization techniques

It is defined as minimizing the following programs:

$$
\begin{aligned}
\phi^*_{ML} &= \operatorname*{argmin}_{\phi} -\log p(\mathbf{I}|\phi) \\
\phi^*_{MAP} &= \operatorname*{argmin}_{\phi} -\log p(\mathbf{I}|\phi) - \log p(\phi)
\end{aligned}
$$

It suffers from the following problems:

- Local minima: usually $-\log p(\mathbf{I}|\phi)$ is a non-convex function of ϕ.
- Initialization: usually hand initialized or use discriminative approaches.
- Drift: As times goes, the estimate gets worst.
- Difficult to define a good general $-\log p(\mathbf{I}|\phi)$.

Usually computationally more efficient than particle filter (if not use GPU).

This can be combined with particle filter to create hybrid monte-carlo.

Optimization techniques

It is defined as minimizing the following programs:

$$\begin{aligned}\phi^*_{ML} &= \operatorname*{argmin}_{\phi} -\log p(\mathbf{I}|\phi) \\ \phi^*_{MAP} &= \operatorname*{argmin}_{\phi} -\log p(\mathbf{I}|\phi) - \log p(\phi)\end{aligned}$$

It suffers from the following problems:

- Local minima: usually $-\log p(\mathbf{I}|\phi)$ is a non-convex function of ϕ.
- Initialization: usually hand initialized or use discriminative approaches.
- Drift: As times goes, the estimate gets worst.
- Difficult to define a good general $-\log p(\mathbf{I}|\phi)$.

Usually computationally more efficient than particle filter (if not use GPU).

This can be combined with particle filter to create hybrid monte-carlo.

In the next lectures we will look at ...

Priors: $p(\phi)$

- Joint limits
- Shape priors
- Pose priors
- Dynamical priors
- Physics

Likelihood models: $p(\mathbf{I}|\phi)$

- Monocular tracking: 2D-3D correspondences, silhouettes, edges, template matching, etc.
- Multi-view tracking: stereo, visual hull, etc.

In the next lectures we will look at ...

Priors: $p(\phi)$

- Joint limits
- Shape priors
- Pose priors
- Dynamical priors
- Physics

Likelihood models: $p(\mathbf{I}|\phi)$

- Monocular tracking: 2D-3D correspondences, silhouettes, edges, template matching, etc.
- Multi-view tracking: stereo, visual hull, etc.

Note that I have defined ϕ as a general quantity, not just the pose.

In the next lectures we will look at ...

Priors: $p(\phi)$

- Joint limits
- Shape priors
- Pose priors
- Dynamical priors
- Physics

Likelihood models: $p(\mathbf{I}|\phi)$

- Monocular tracking: 2D-3D correspondences, silhouettes, edges, template matching, etc.
- Multi-view tracking: stereo, visual hull, etc.

Note that I have defined ϕ as a general quantity, not just the pose.

Priors already cover in the class

Priors already seen in class:

- Pose prior: Dimensionality reduction techniques
- Dynamics priors: LDS, HMMs.

Today we will look into joint limits

Joint limits

We will look into increasing complexity of joint limits

- Min-max euler angles
- Joint sinus cones
- Spherical polygons
- Triangular Bezier patches
- Implicit surface representation of limits
- Hierarchical joint limits: e.g., elbow limits depends on the shoulder rotations.

Min-max joint limits

- We are going to talk about 3 dof joints.
- Ranges on three independent axis rotations.
- Problem: it is not realistic

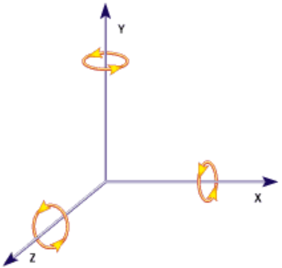
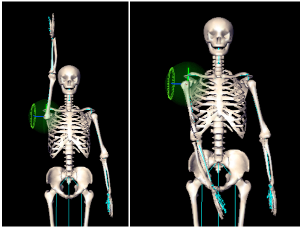
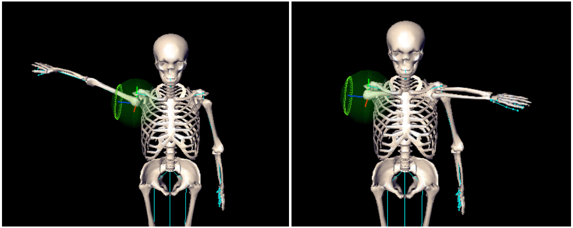


Figure: Min-Max joint limits

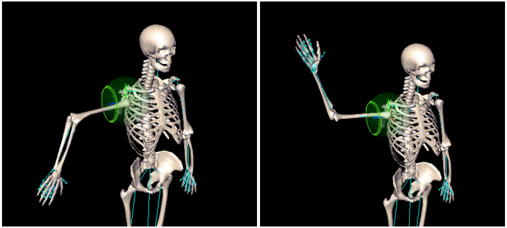
Typical values of min-max limits for shoulder



Around x-axis [-90,100]



Around y-axis [-30,180]



Around z-axis [-90,90]

Joint sinus cones

- Joint sinus cones (Engin and Chen, 1989): only limit swing

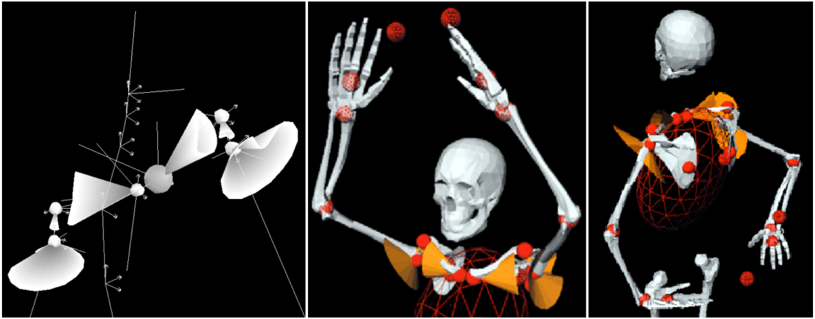
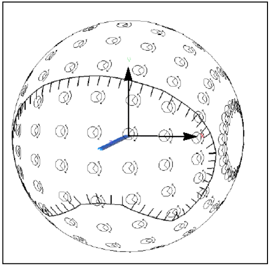


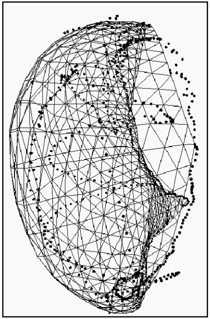
Figure: Illustrations from (Maurel and Thalmann, 1998)

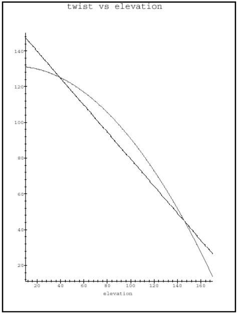
Spherical polygons

- Limits angular rotation and defines local twist ranges (Korein, 1985).



- Limits angular rotation and defines twist versus elevation (Tolani et al., 2000).





Desired properties

Joint limits should ...

- define motion boundaries for swing AND twist.
- easily allow to determine whether a rotation is valid or not.
- be applicable to any joint.
- incorporate coupling between joints

A data-driven approach to joint limits

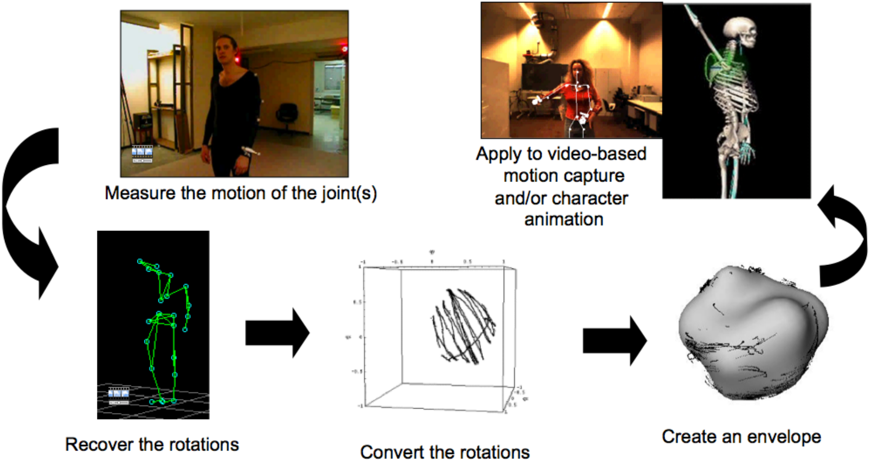


Figure: (Herda, Urtasun and Fua, 2004)

Capture

Measure joint motion using optical motion capture

- Strategically placed markers.
- Cover entire range of motion if possible.

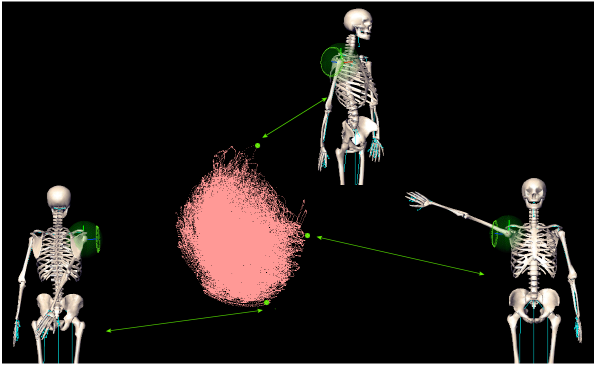
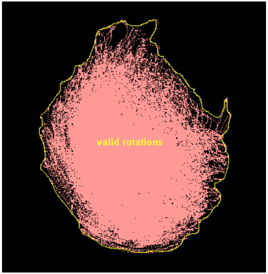


Figure: Vector representation of quaternions for shoulder motion

Boundary representation

Find the boundary of the data that represents the joint limits

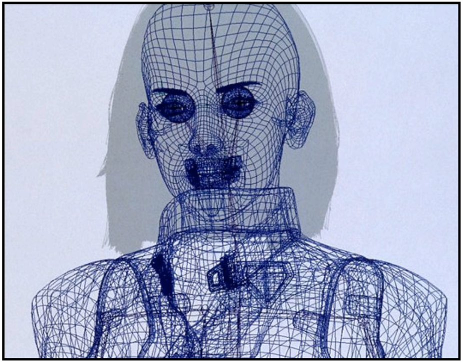


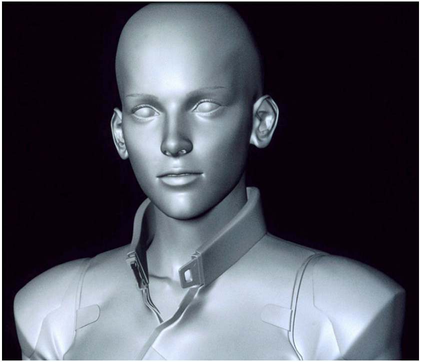
Choose a representation:

- Implicit surfaces
- Mesh

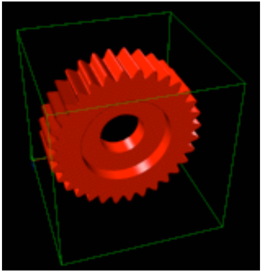
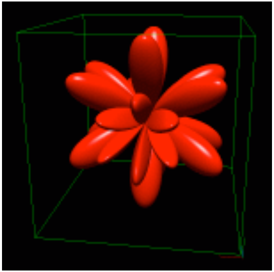
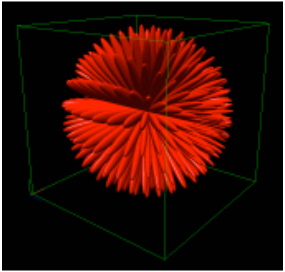
Mesh representation

- Explicit representation (triangles, polygons)
- Computationally costly
- High storage cost





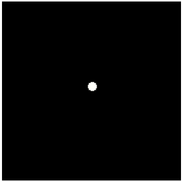
- are smooth and continuous
- have an analytical expression
- use few parameters
- can be locally influenced, to capture shape detail

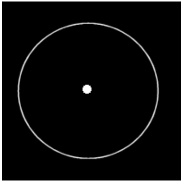




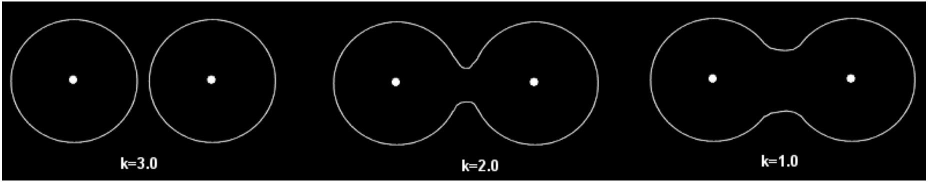
Implicit surfaces II

A 3D implicit surface with spherical primitives is defined by:

- a centre or skeleton (x, y, z)
- a radius or thickness (e)
- a stiffness (k)





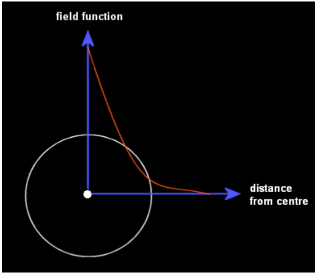


Implicit surfaces III

- Each spherical primitive has an energy field around it, defined by a field function f_i (Tsingos et al., 1995):

$$f_i(P) = \begin{cases} -k_i r + k_i e_i + 1 & \text{if } r \in [0, e_i] \\ \frac{1}{4}[k_i(r - e_i) - 2]^2 & \text{if } r \in [e_i, R_i] \\ 0 & \text{otherwise} \end{cases}$$

where $r = d(P, S_i)$, and $R_i = e_i + \frac{2}{K_i}$ is the radius of influence.



The surface is the level set of all primitives:

$$f(P) = \sum_{i=1}^{N} f_i(P)$$

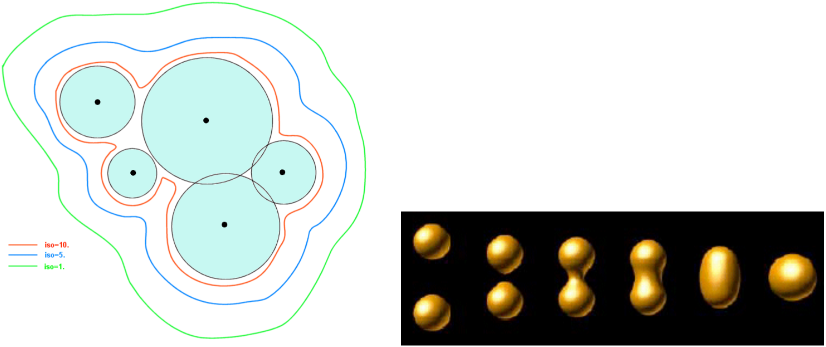


Figure: Isocurves of implicit surfaces

Illustration of representing joint limits

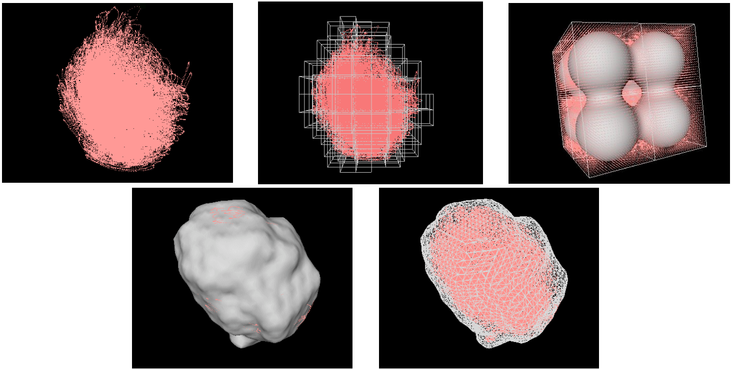
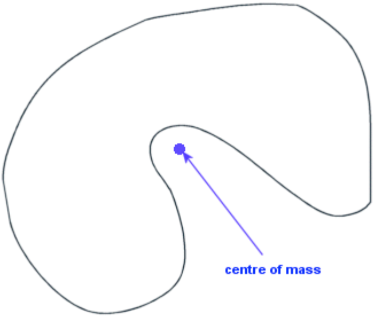


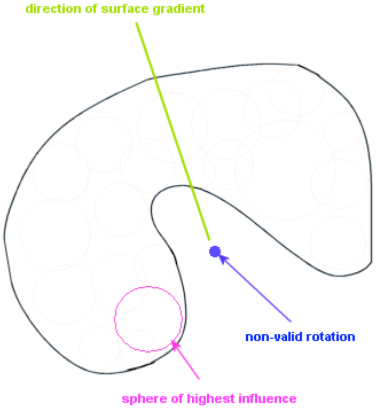
Figure: Fitting implicit surfaces to volumetric data

Applications to character animation

Use projective gradient descent and move in the direction of

- center of mass
- global gradient
- sphere of highest influence





Animation results

Inter-subject variance

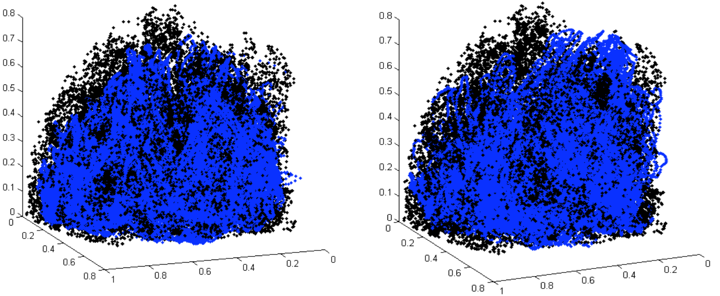
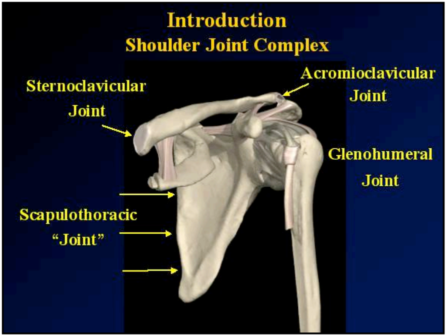


Figure: (left) Male vs (right) Female. The average inter-point distance is 0.0314 ± 0.04 for the male and 0.0403 ± 0.05 for the female.

Modeling coupling joints

We will focus on the upper arm

- Coupled sterno-clavicular (clavicular) and gleno-humeral (shoulder) joints (2 DOF + 3 DOF)
- Coupled shoulder and elbow joints (3 DOF + 2 DOF)



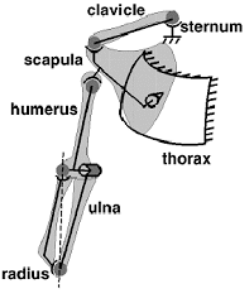


Figure: Coupling between joints. (left) sterno-clavicular, (right) shoulder-elbow.

Modeling coupling joints

We will focus on the upper arm

- Coupled sterno-clavicular (clavicular) and gleno-humeral (shoulder) joints (2 DOF + 3 DOF)
- Coupled shoulder and elbow joints (3 DOF + 2 DOF)

Figure: Clavicular joint contributes to 1/3rd of shoulder elevation, until it reaches its limit

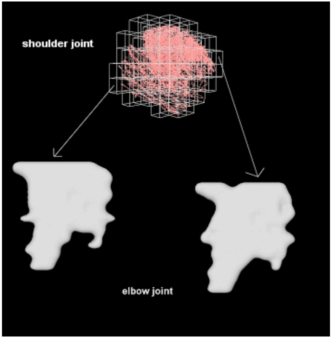
Motion measurements

- Strategically placed markers, around clavicular, shoulder and elbow joints.
- Measure the coupled ranges of motion of
  - the shoulder and elbow joints
  - the clavicular and shoulder joints

Limits captured

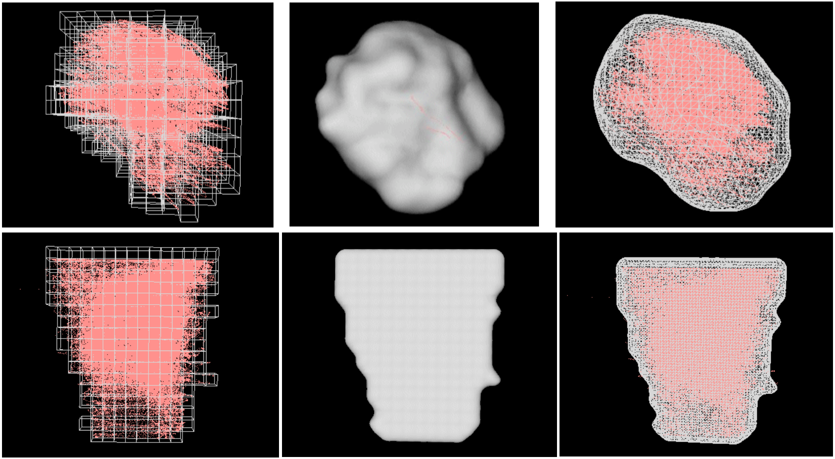
Hierarchical joint limits

- Re-use voxelisation to create parent joint clusters.
- For each such cluster, create a child joint implicit surface.
- Problems for clavicle and elbow joint rotations:
  - Surface-like quaternions are not readily voxelisable.
  - So-defined implicit surface not directly applicable.
  - Convert to Euler angles with one zero component.

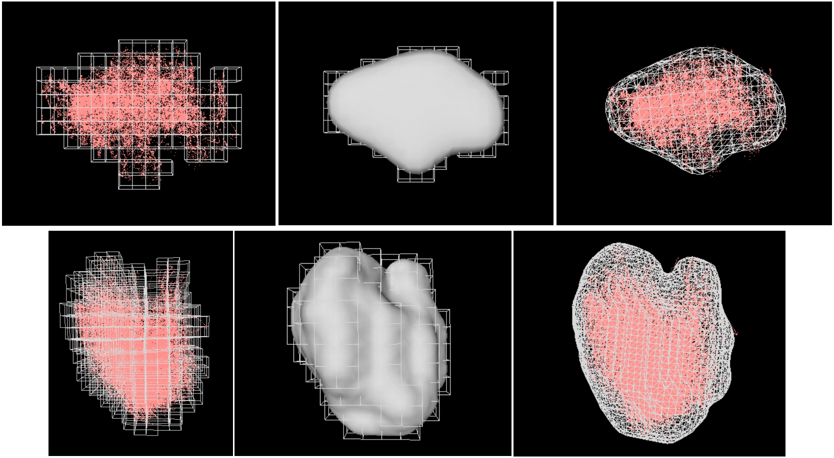


Quaternions and equivalent Euler angle rotations

Without coupling shoulder and elbow



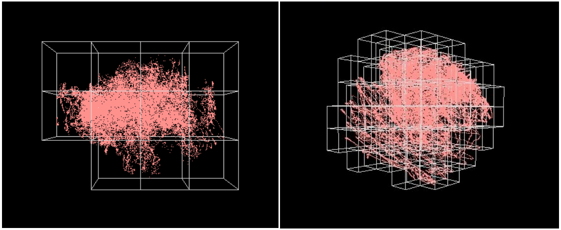
Without coupling clavicle and shoulder



Hierarchical joint limits

- Lower-resolution voxelisation of parent joint data, respectively clavicle and shoulder joints.
- For each sub-set of child joint data points
  - Approximate by an implicit surface.
  - Refine voxelisation by morphing for better joint limits continuity.

Example for child joint limits



Hierarchical clavicular/shoulder joints

Hierarchical clavicular/shoulder joints II

Hierarchical shoulder/elbow joints I

Hierarchical shoulder/elbow joints

Application to character animation I

Figure: (left) Unconstrained. (right) Constrained animation.

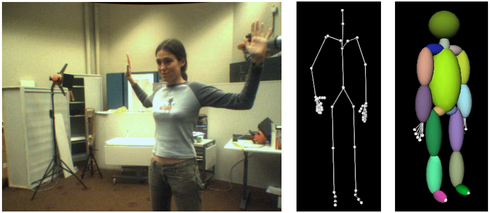
Application to character animation II

Application to character animation III

Application to character animation IV

Application to video-based motion capture

- Input data: Stereo
- Model: Articulated skeleton, were each body part is an ellipsoidal primitive



Application to video-based motion capture

- Input data: Stereo
- Model: Articulated skeleton, were each body part is an ellipsoidal primitive
- Optimization with multiple constraints
  - Weighting strategy: give the constraint a higher weight, but yields composite solution.
  - Task priority strategy: find a posture that satisfies the joint limits

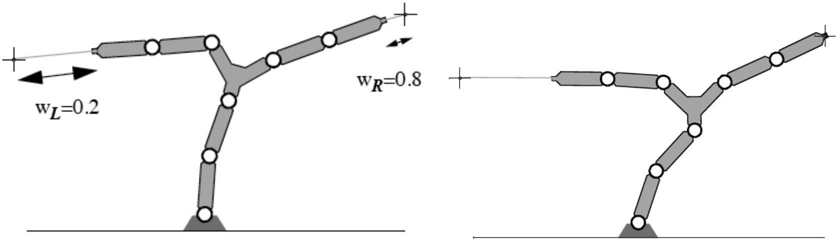


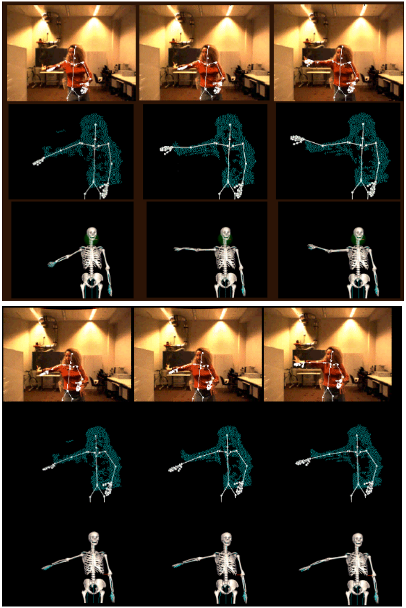
Figure: (left) Weighting strategy. (right) Priority strategy (Baerlocher01)

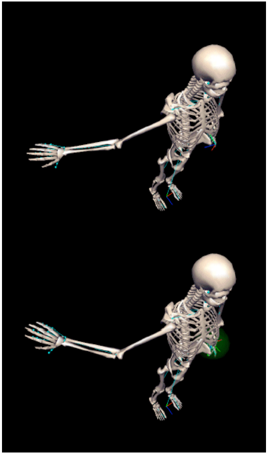
Video-based motion capture I

Figure: (left) Unconstrained tracking.

Figure: Tracking with joint limit priors

Detailed comparison I



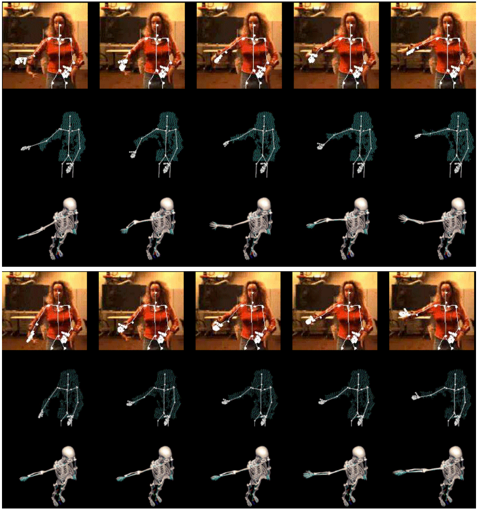


Video-based motion capture II

Figure: (left) Unconstrained tracking.

Figure: Tracking with joint limit priors

Detailed comparison II



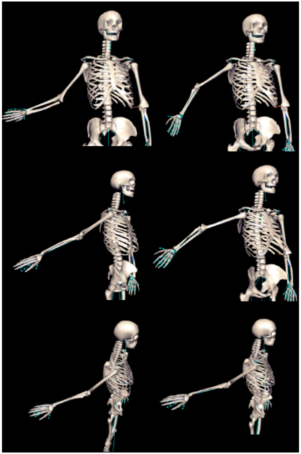
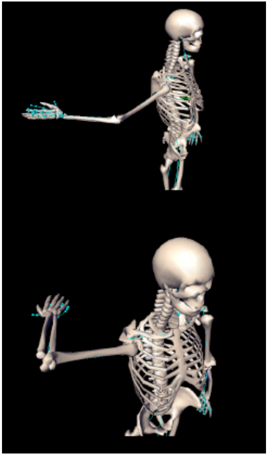


Figure: (left) Unconstrained tracking.

Figure: Tracking with joint limit priors

Detailed comparison III

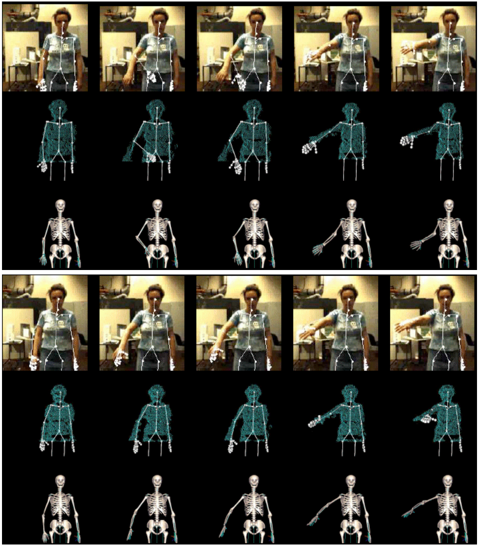


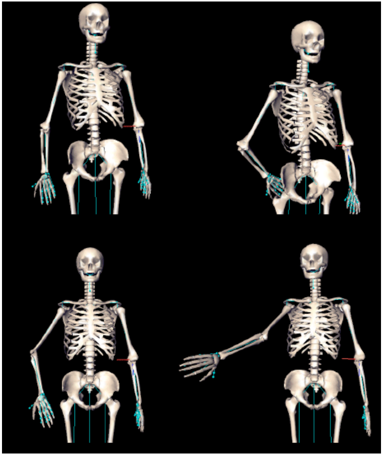
Video-based motion capture IV

Figure: (left) Unconstrained tracking.

Figure: Tracking with joint limit priors

Detailed comparison IV



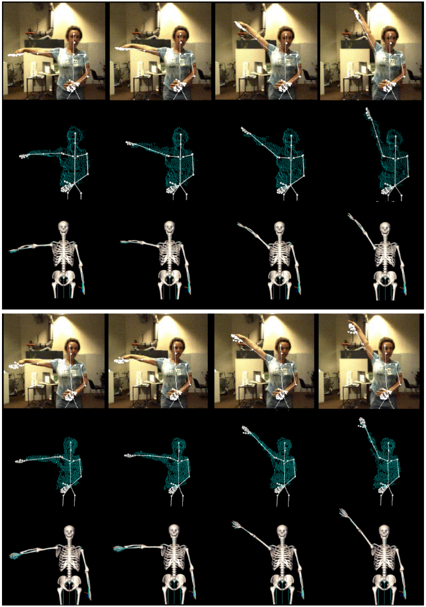


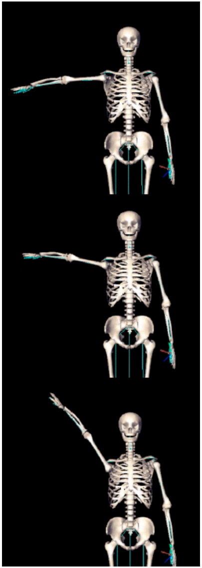
Video-based motion capture V

Figure: (left) Unconstrained tracking.

Figure: Tracking with joint limit priors

Detailed comparison V





Conclusions

- Joint limits constrain any type of motion capture.
- For 3 DOF joints, includes all motion components, i.e. inter-joint coupling.
- Extension to coupled joints.
- Implicit expression allows rapid validation and differentiation.

More?

- If you want to learn more, look at the additional material.
- Otherwise, do the research project on this topic!
- Next week we will look into image likelihoods and shape priors