Image-Based 3D Body Tracking

888 Seminar presentation

Autumn'07

Youding Zhu

Outline

1. ***Image-based 3D Body Tracking***
   - *Literature review*
   - *Challenges*
   - *Sub-fields*
2. ***Scaled Motion Dynamics for Markerless Motion Capture***
   - *Key point of the paper*
   - *Overview of the method*
   - *Review: Level set segmentation*
   - *Review: twist representation*
   - *Level set segmentation for pose estimation*
   - *Incorporate training data pattern*
   - *Results*
3. ***Markerless Deformable Mesh Tracking for Human Shape and Motion Capture***
   - *Key point of the paper*
   - *Overview of the method*
   - *StepA*
   - *StepB*
   - *Results*

Image-based 3D Body Tracking: literature review

Goal of the research:

is to estimate the body configuration, i.e. joint angles, from captured images.

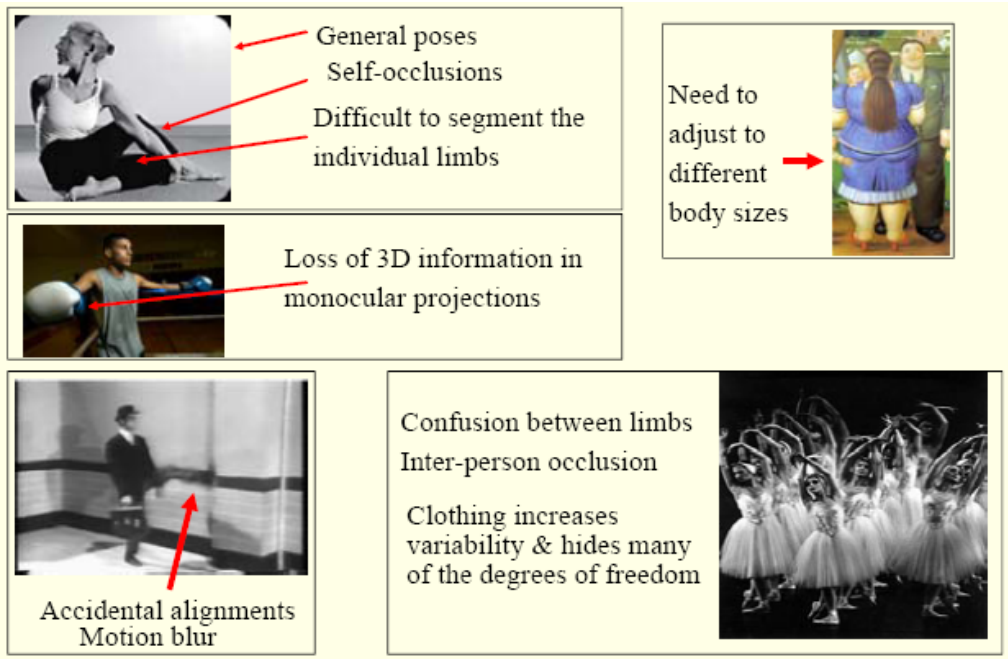
Active topic as reflected by the variety of publications:

Approaches	Papers
Model-based method	
Multiple Camera	Gavrila96 [11], Kakadiaris00 [13], Deutscher00 [15], Cheung00 [9], Delamarre01 [7], Carranza03 [14], Kehl06 [24]
Single Camera	Yamamoto91 [20], Bregler98 [3], Sidenbladh00 [12], Sminchisescu03 [4], Lee04 [18], Sigal04 [19]
Depth Camera	Grest05 [6], Knoop06 [26], Ziegler06 [16]
Learning-based method	
Multiple Camera	Ren05 [17]
Single Camera	Howe99 [23], Rosales01 [25], Mori02 [22], Shakhnarovich03 [10], Agarwal06 [1], Sminchisescu05 [5]

Image-based 3D Body Tracking:challenges

Yet the problem remains unsolved, and is challenging for vision because of

1. *the high number of degrees of freedoms arisen from the dynamic range of poses during human activities;*
2. *the diversity of visual appearance caused by clothing;*
3. *the visual ambiguities arisen from self-occlusion of non-rigid 3D object;*
4. *the background clutters.*



Cited from HumanMotionAnalysis by Bill Triggs

Image-based 3D Body Tracking: research topics

Moeslund et al present a thorough survey about the field up to year 2006.

They divide the field into four areas:

1. *Initialization. Ensuring that a system commences its operation with a correct interpretation of the current scene.*
2. *Tracking. Segmenting and tracking humans in one or more frames.*
3. *Pose estimation. Estimating the pose of a human in one or more frames.*
4. *Recognition. Recognizing the identity of individuals as well as the actions, activities and behaviors performed by one or more humans in one or more frames.*

Scaled Motion Dynamics for Markerless Motion Capture

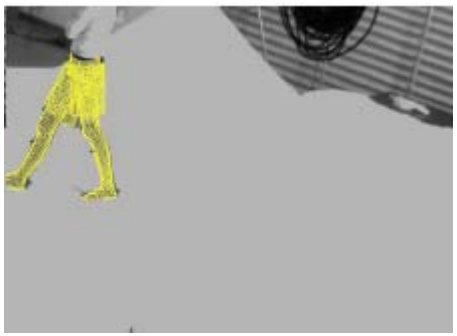
- *Key point of the paper*
- *Review: Level set segmentation*
- *Review: twist representation*
- *Level set segmentation for pose estimation*
- *Incorporate training data pattern*
- *Results*

Scaled Motion Dynamics: key point

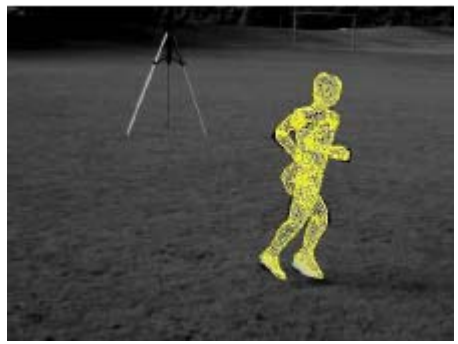
Additional a-priori information (e.g. motion db) about familiar pose configurations

(1) constrains the search space

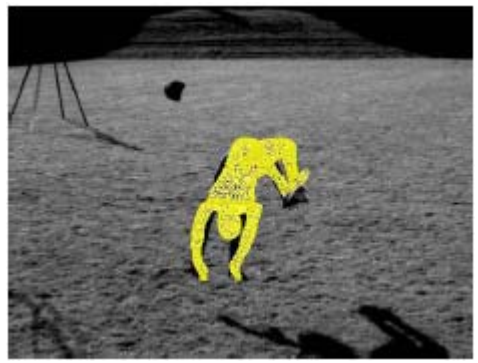
(2) and helps considerably to handle more difficult scenarios with partial occlusions, background clutter, or corrupted image data.



Walking



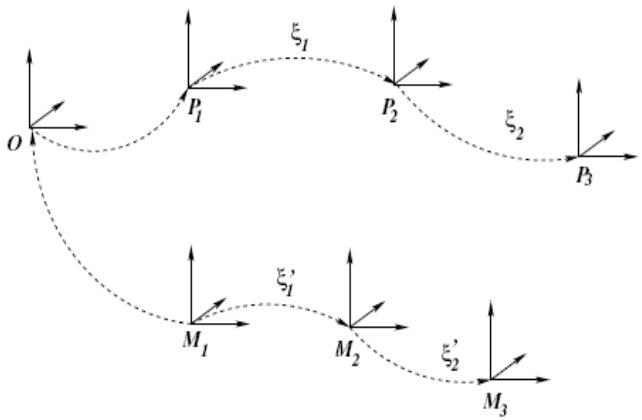
Joggling



Cartwheel

Scaled Motion Dynamics: method overview

The paper studied three ideas:



(1) Twist motion representation to achieve motion scaling (velocity re-sampling)

$$\text{argmin}_{s,j} \sum_{v=0}^{m-1} \left(\sqrt{\sum_{k=1}^{n} (\theta_{k,t-v} - \widetilde{\theta}^{s}_{k,j-v})^2} \right)$$

$$\hat{\xi}' = g \log \left(\exp(\hat{\tilde{\xi}}^{s}_{j+1}) \exp(\hat{\tilde{\xi}}^{s}_{j})^{-1} \right) g^{-1}.$$

$$\underline{\hat{\xi}} := \hat{\xi}' \frac{\nu}{\tilde{\nu}},$$

(2) Matching the similar motion pattern and predict

$$E(\Phi, p_1, p_2, \chi) =$$

$$\underbrace{-\int_{\Omega} \left(H(\Phi) \log p_1 + (1 - H(\Phi)) \log p_2 + \nu |\nabla H(\Phi)| \right) dx}_{\text{segmentation}}$$

$$\underbrace{+\lambda \int_{\Omega} (\Phi - \Phi_0(\chi))^2 dx}_{\text{shape error}}$$

$$+ (\log(\exp(\hat{\underline{\xi}}) \exp(\hat{\xi})^{-1}), \underline{\Theta} - \Theta)$$

(3) Incorporate prediction into level set segmentation for pose estimation

Scaled Motion Dynamics: twist representation review

Twist representation of rigid body motion:

$$\xi \in se(3) = \left\{ (v, \hat{\omega}) \middle| v \in R^3, \hat{\omega} \in so(3) \right\}$$

$$so(3) = \left\{ A \in R^{3\times 3} \middle| A = -A^T \right\}$$

$$\theta\omega = \theta \begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{pmatrix}, \text{ with } \|\omega\|_2 = 1$$

$$\theta\hat{\omega} = \theta \begin{pmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{pmatrix}$$

Scaling with twist representation (varying velocity):

$$\theta\xi = \theta(\omega_1, \omega_2, \omega_3, v_1, v_2, v_3)^T, \|\omega\|_2 = 1, \quad (4)$$

$$\theta\hat{\xi} = \theta \begin{pmatrix} 0 & -\omega_3 & \omega_2 & v_1 \\ \omega_3 & 0 & -\omega_1 & v_2 \\ -\omega_2 & \omega_1 & 0 & v_3 \\ 0 & 0 & 0 & 0 \end{pmatrix}. \quad (5)$$

Scaled Motion Dynamics: twist representation review

se(3) to SE(3):

$$\exp(\theta\hat{\xi}) = \begin{pmatrix} \exp(\theta\hat{\omega}) & (I - \exp(\theta\hat{\omega}))(\omega \times v) + \omega\omega^T \boldsymbol{v}\theta \\ 0 & 1 \end{pmatrix}$$

$$\exp(\theta\hat{\omega}) = I + \hat{\omega}\sin(\theta) + \omega^2(1 - \cos(\theta))$$ (Rodriguez formula)

SE(3) to se(3):

$$M = \begin{pmatrix} R & t \\ 0 & 1 \end{pmatrix}$$

if $R = I$ → $\theta\xi = \theta(0, 0, 0, \frac{t}{\|t\|}), \qquad \theta = \|t\|.$

otherwise →

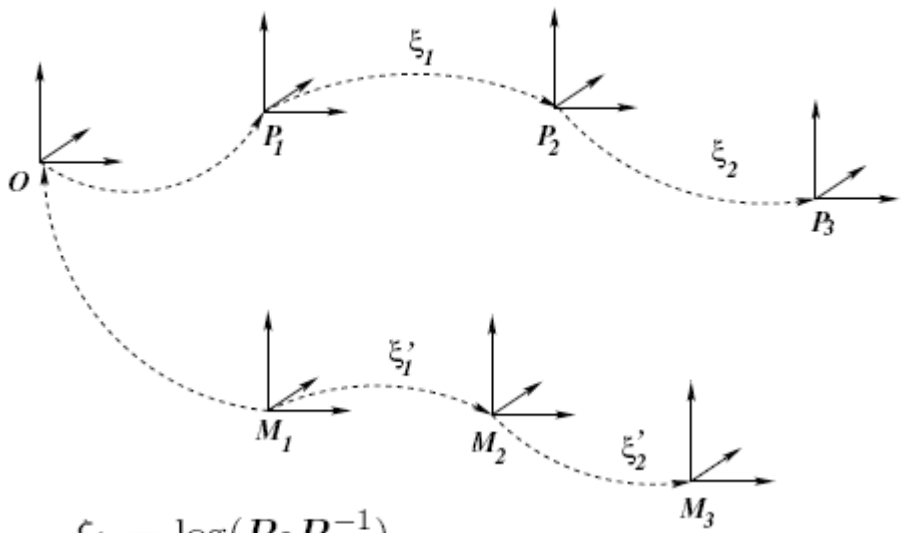
$$\theta = \cos^{-1}\left(\frac{trace(R) - 1}{2}\right) \quad \omega = \frac{1}{2\sin(\theta)} \begin{pmatrix} r_{32} - r_{32} \\ r_{13} - r_{31} \\ r_{21} - r_{12} \end{pmatrix}$$

$$A = (I - \exp(\theta\hat{\omega}))\hat{\omega} + \omega\omega^T\theta \qquad v = A^{-1}t.$$

Scaled Motion Dynamics: twist representation review

Advantages with twist representation:

(1) Coordinate transformation with twist representation:



$$\xi_1 = \log(P_2 P_1^{-1})$$

$$\xi_1' = g\xi_1 g^{-1} = M_1 P_1^{-1} \xi_1 P_1 M_1^{-1}$$

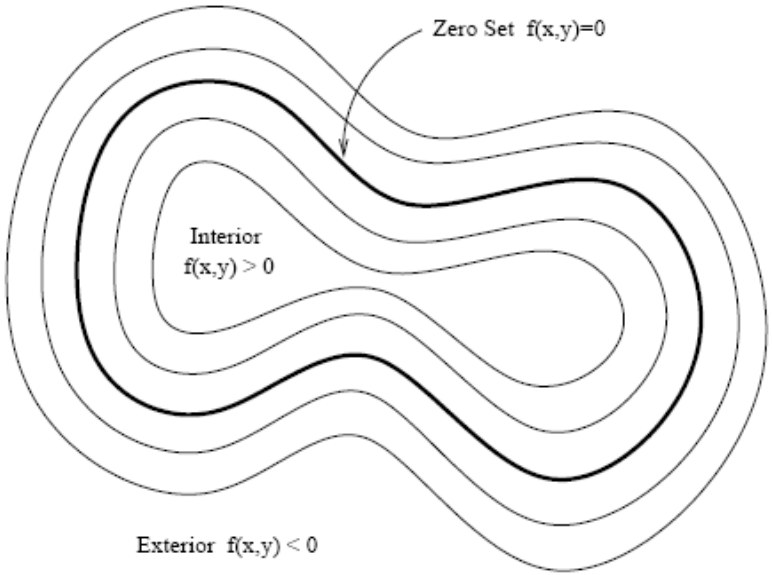
(2) Motion scaling with twist representation:

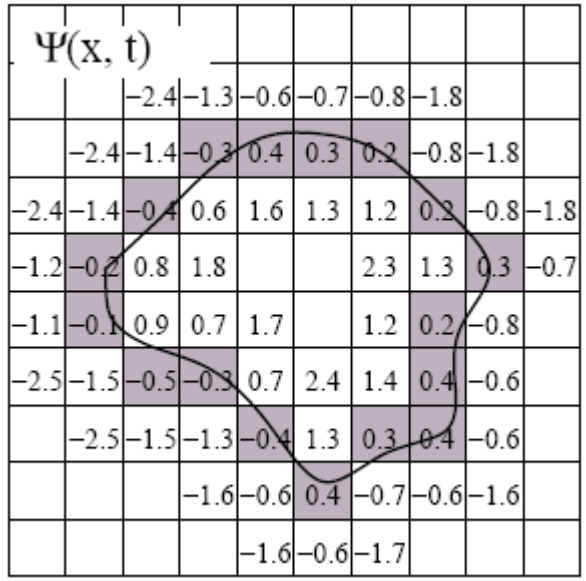
$$\xi'' = \nu\xi'$$

(3) Standard motion predict with twist representation:

$$A_{pred} = \exp(\xi'') A_{current}$$

Scaled Motion Dynamics: level set segmentation review





Cited from ITK software manual

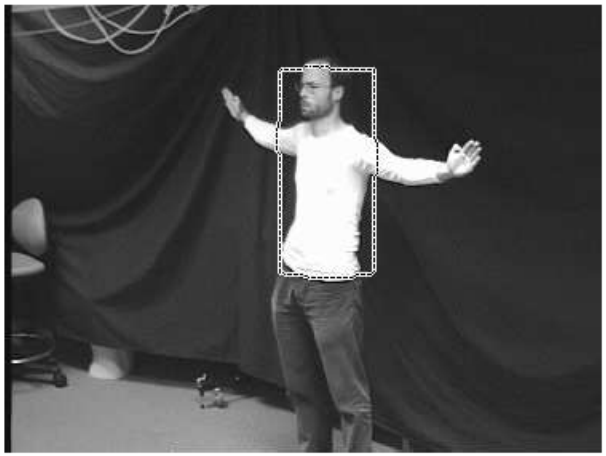
(1) Define level set function: $\psi(\mathbf{X}, \mathbf{t})$

(2) Evolving the level set function according image observation and other factors

$$\frac{d}{dt}\psi = -\alpha \mathbf{A}(\mathbf{x}) \cdot \nabla\psi - \beta P(\mathbf{x}) \mid \nabla\psi \mid + \gamma Z(\mathbf{x})\kappa \mid \nabla\psi \mid$$

(3) Zero level set function defines the boundary/segmentation $\mathrm{f(x,y)=0}$

Scaled Motion Dynamics: Level set segmentation



$$\Phi(x) > 0 \quad \text{if } x \in \Omega_1$$
$$\Phi(x) < 0 \quad \text{if } x \in \Omega_2$$

(1) The data within each region should follow its distributions

(2) The contour should be minimal

minimize $$E(\Phi, p_1, p_2) = -\int_\Omega \big(H(\Phi(x)) \log p_1(I(x)) + (1 - H(\Phi(x))) \log p_2(I(x)) + \nu |\nabla H(\Phi(x))|\big)\, dx \qquad (15)$$

(1) H is a step function

(2) p_1 and p_2 are density functions of corresponding regions

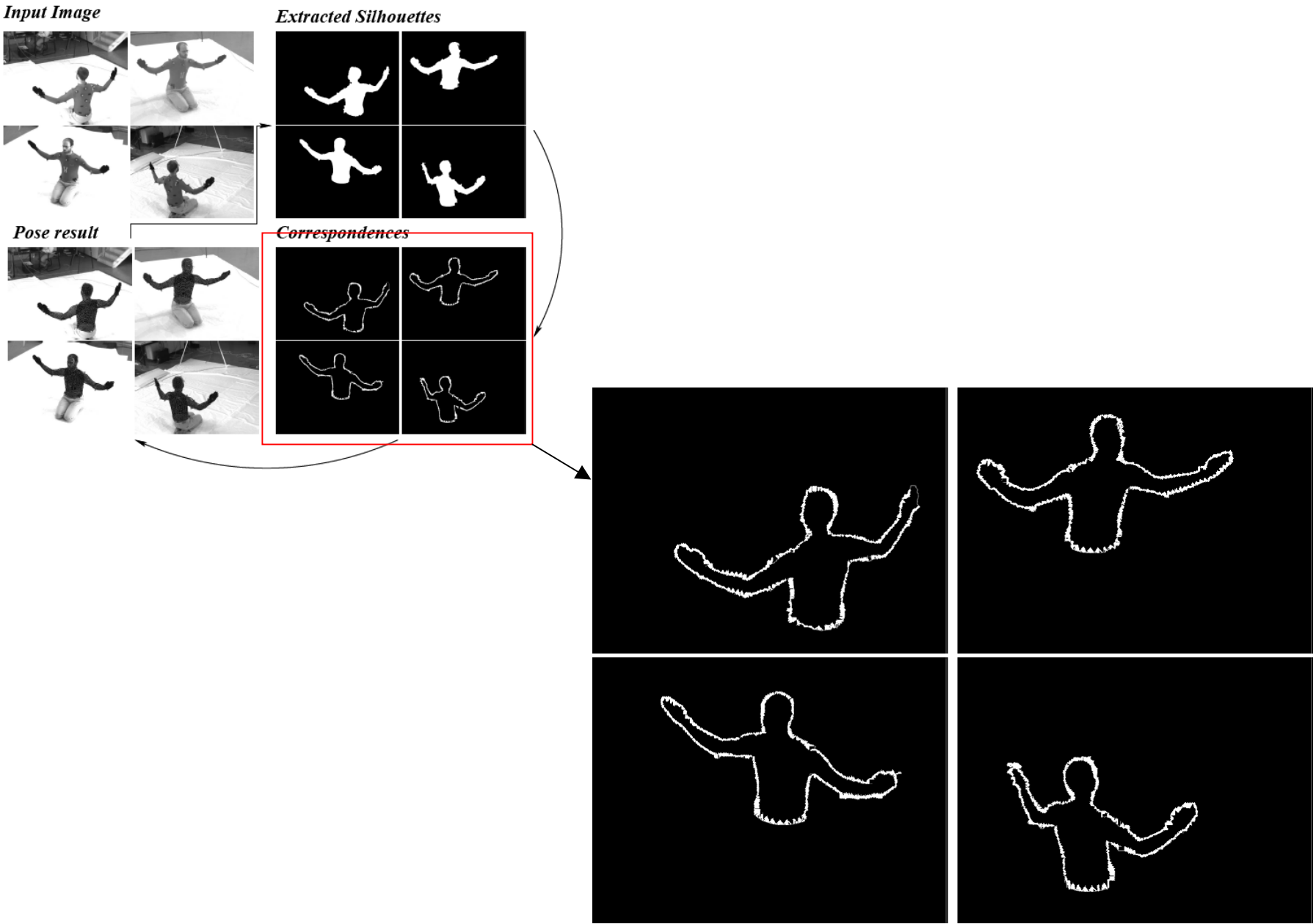
Two iterative steps:

(1) Minimize over Φ using level set segmentation to estimate the partition

$$\frac{\partial \phi}{\partial t} = -\frac{\partial E(\phi)}{\partial \phi} \quad \text{???} \quad \partial_t \Phi = H'(\Phi)\left(\log \frac{p_1}{p_2} + \nu \operatorname{div}\left(\frac{\nabla \Phi}{|\nabla \Phi|}\right)\right)$$

(2) Minimize over p_1 and p_2 using the estimated partition

Level set segmentation for pose estimation



Level set segmentation for pose estimation

minimize $E(\Phi, p_1, p_2, \chi) =$

$$\underbrace{-\int_\Omega \left(H(\Phi)\log p_1 + (1-H(\Phi))\log p_2 + \nu|\nabla H(\Phi)|\right) dx}_{\text{segmentation}}$$

$$\underbrace{+\lambda \int_\Omega (\Phi - \Phi_0(\chi))^2 dx}_{\text{shape error}}$$

(1) The data within each region should follow its distributions

(2) The contour should be minimal

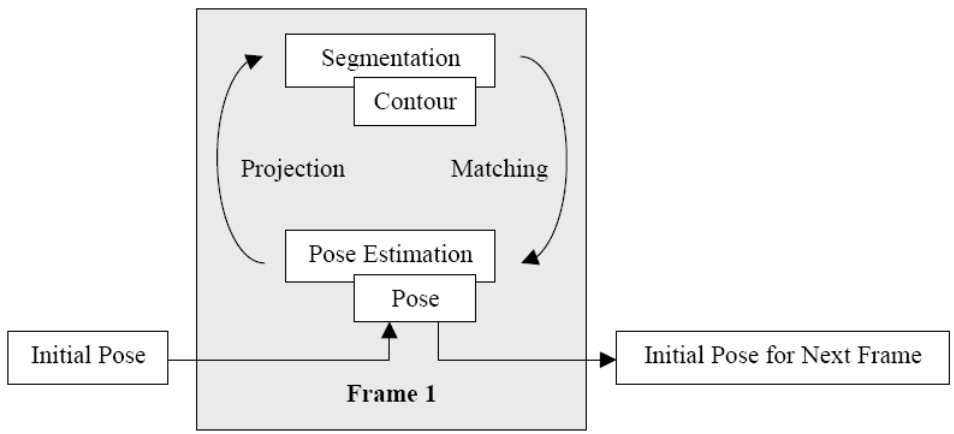
(3) The contour should be close to the projected model contour

Two iterative steps:

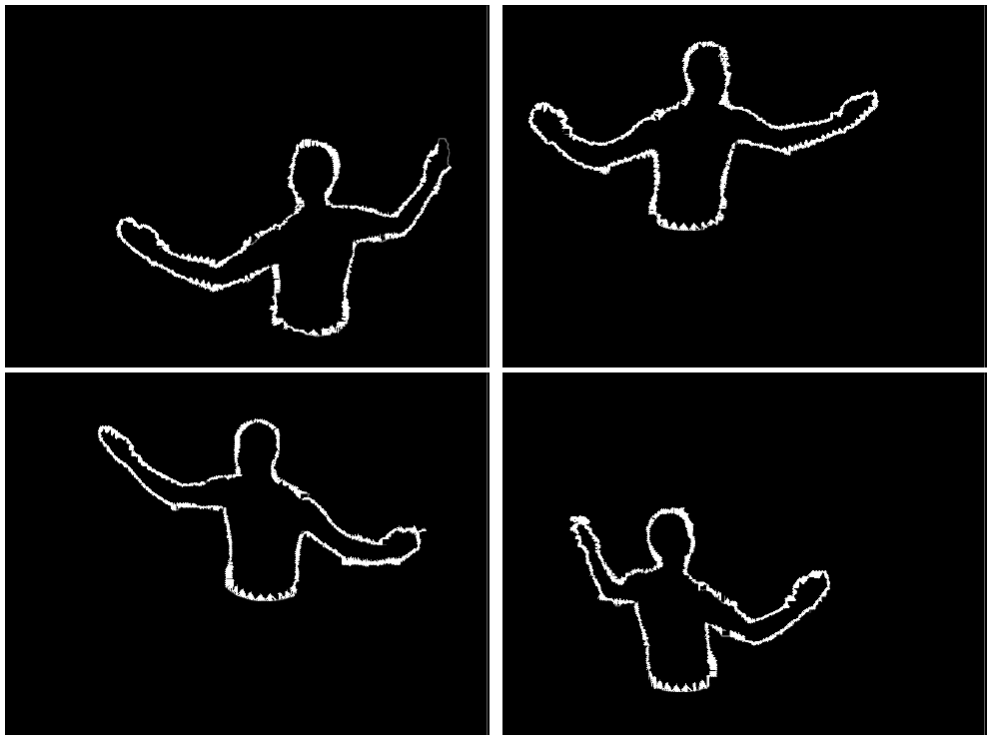
(1) Segmentation or minimize over Φ using level set segmentation to estimate the partition with fixed pose parameters

$$\partial_t \Phi = H'(\Phi)\left(\log\frac{p_1}{p_2} + \nu\,\mathrm{div}\left(\frac{\nabla\Phi}{|\nabla\Phi|}\right)\right) + 2\lambda\,(\Phi_0 - \Phi).$$

(2) Minimize over pose parameters with fixed



Pose estimation with ICP



(1) Project the model to image plane

(2) Compute the closest point correspondence

(3) Set up a set of equation using correspondences

$$(\exp(\theta\hat{\xi})\exp(\theta_1\hat{\xi}_1)\ldots\exp(\theta_j\hat{\xi}_j)X_i)_{3\times 1}\times n_i - m_i \quad = \quad 0.$$

(4) Solve pose parameters

Why training pattern is needed

The authors said:

(1) This quality of pose estimation based on how well the image data determines the solution. In misleading situations, the minimum of the energy above might not be the true pose.

(2) Moreover, it uses the local minimization scheme, and the system can in general not recover after it has lost track.

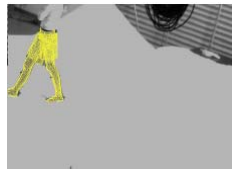
Suggestion:

(1) To compute a pose prediction from training data and keep the solution close to this prediction in case the image data is misleading or insufficient, such as frame drops

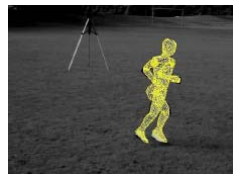
Pose estimation with training data

Given a set of training examples:

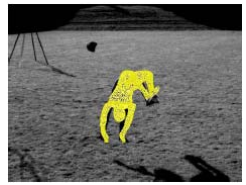
$$\{\widetilde{\chi}_i := (\widetilde{\xi}_i, \widetilde{\theta}_{1,i}, \ldots \widetilde{\theta}_{n,i}) := (\widetilde{\xi}_i, \widetilde{\Theta}_i) | i = 0 \ldots N\}$$



Walking



Joggling



Cartwheel

Assuming we have already tracked m frames:

$$\langle \chi_{t-m+1} \cdots \chi_t \rangle$$

Question1. How to predict the pose at next frame:

$$\underline{\chi} = (\underline{\xi}, \underline{\Theta})$$

(1) Matching

(2) Prediction

Question2. How to incorporate prediction to pose estimation

Pose estimation with training data: matching

Matching with twist representation:

(1) Scaling and interpolating the motion:

$\mathcal{P} = \langle \widetilde{\chi}_0 \dots \widetilde{\chi}_N \rangle$ where $\widetilde{\chi}_i := (\xi_i, \theta_{1,i}, \dots \theta_{n,i})$ → Original motion set

$\mathcal{P}^s = \{\widetilde{\chi}_i^s := (\widetilde{\xi}_i^s, \widetilde{\theta}_{1,i}^s, \dots \widetilde{\theta}_{n,i}^s) | i = 0 \dots \lceil sN \rceil$

$s \in [0.5 \dots 2]$ → Sampled motion set

(2) Find the best matching over velocity and time

$$\text{argmin}_{s,j} \sum_{v=0}^{m-1} \left(\sqrt{\sum_{k=1}^{n} (\theta_{k,t-v} - \widetilde{\theta}_{k,j-v}^s)^2} \right)$$

Pose estimation with training data: matching

Prediction with twist representation:

(1) Predict the joint angles:

$$\underline{\Theta} \quad = \quad \Theta_t + \partial\widetilde{\Theta}^s_{j+1} = \Theta_t + (\widetilde{\Theta}^s_{j+1} - \widetilde{\Theta}^s_j)$$

From matched motion

(2) Predict the root body motion:

$$\hat{\xi}' \quad = \quad g \log \left(\exp(\hat{\tilde{\xi}}^s_{j+1}) \exp(\hat{\tilde{\xi}}^s_j)^{-1} \right) g^{-1}$$

From matched motion

g represents the transformation from prior to the current coordinate system

(3) Scaling the predicted root body motion based on velocity difference between current estimation and prior:

$$\underline{\hat{\xi}} \quad := \quad \hat{\xi}' \frac{\nu}{\widetilde{\nu}}$$

ν → Velocity from current tracking

$\widetilde{\nu}$ → Velocity from matched motion

Pose estimation with training data: Pose estimation

Pose estimation:

$$E(\Phi, p_1, p_2, \chi) =$$

$$\underbrace{-\int_\Omega \left(H(\Phi)\log p_1 + (1-H(\Phi))\log p_2 + \nu|\nabla H(\Phi)|\right) dx}_{\text{segmentation}}$$

$$\underbrace{+\lambda \int_\Omega (\Phi - \Phi_0(\chi))^2 dx}_{\text{shape error}}$$

$$+ (\log(\exp(\hat{\xi})\exp(\hat{\xi})^{-1}), \underline{\Theta} - \Theta)$$

Integrating the predicted pose into energy function

Scaled Motion Dynamics: results

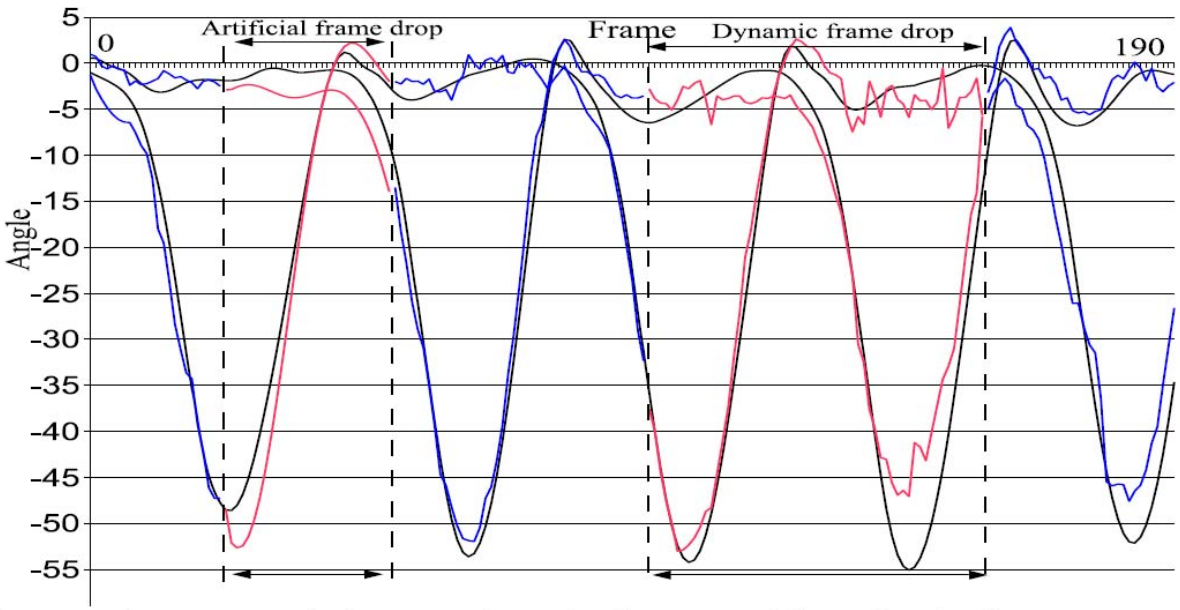


Figure 5. Knee joint angles during tracking including a static frame drop and the dynamic noise from Figure 4.

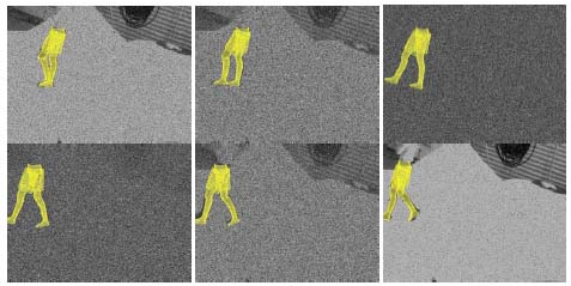


Figure 4. Dynamic noise during tracking.

Averaging errors of the knee angles are 2.58 and 2.83 degrees for the artificial and dynamic frame drop, respectively

Scaled Motion Dynamics: results

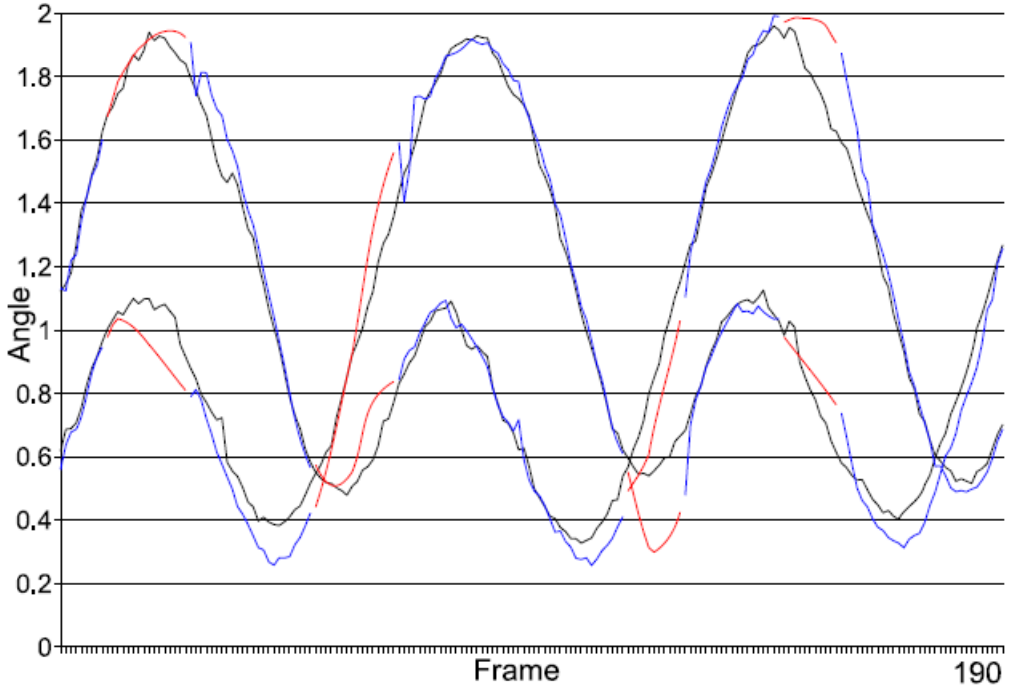


Figure 10. Knee angles of the jogging sequence. Black: Silhouette based MoCap system. Blue/red: The same sequence without frame drops (blue) and with frame drops (red).

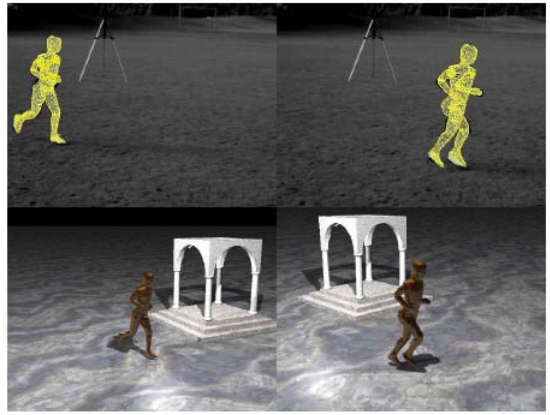
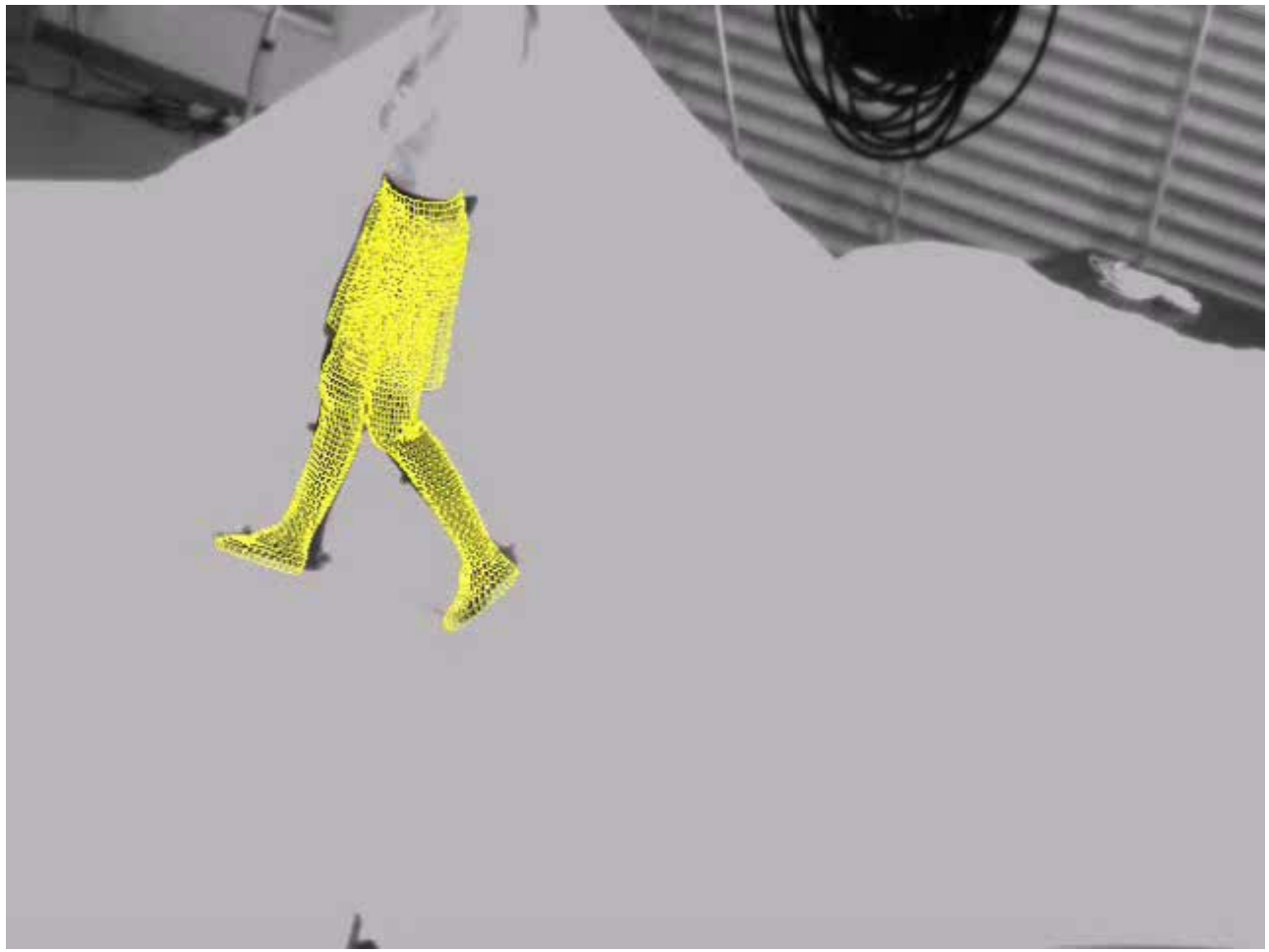


Figure 6. Example frames of an outdoor jogging sequence: The top images visualize the overlay of our estimated model in one of four cameras, the bottom images show the pose result in a virtual environment.

During the frame drops, the averaging absolute difference between the result with and without image data is 7.3 degrees.

Scaled Motion Dynamics: results



Markerless Deformable Mesh Tracking for Human Shape and Motion Capture

- *Key point of the paper*
- *Overview of the method*
- *StepA*
- *StepB*
- *Results*

Deformable mesh tracking: key point

To jointly capture the motion and deformation, for example,

(1) able to track people wearing apparel

Figure 1. Our method realistically captures the motion and the dynamic shape of a woman wearing a Japanese kimono from only eight video streams.

The authors claimed:

(1) To the best of out knowledge, this is the first system of its kind that can capture the motion and non rigid surface deformations of arbitrary subjects from only a handful of cameras

(2) The previous methods needs either heavy manual work or cannot achieve same accuracy

Deformable mesh tracking: method overview

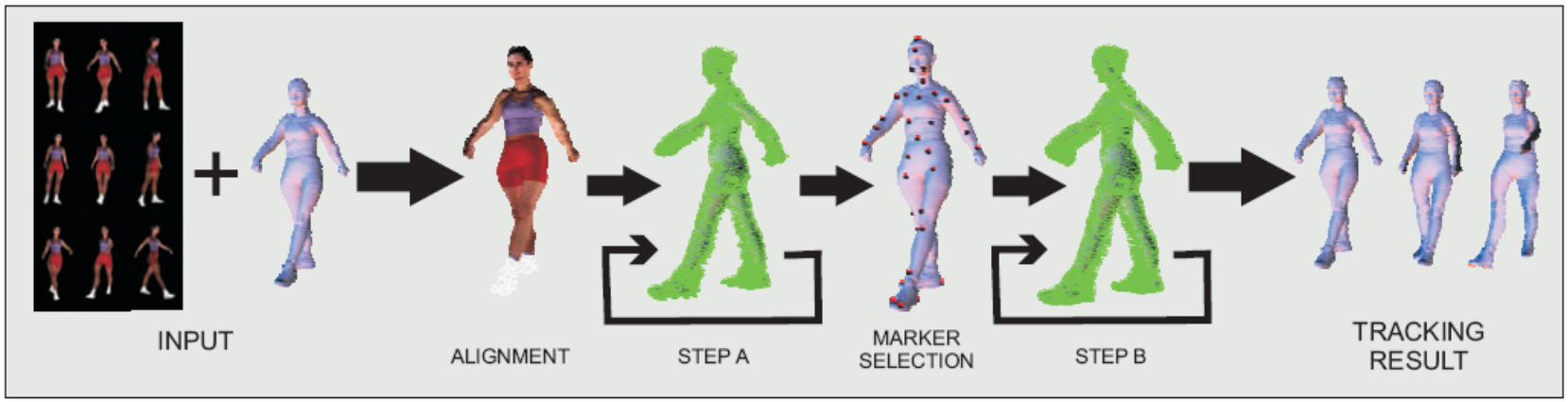


Figure 2. Overview of our marker-less deformable mesh tracking framework: Given a laser-scan of a person and a multi-view video sequence showing her motion, the method deforms the scan in the same way as its real-world counterpart in the video streams.

Given: multi-view video sequences

mesh model from laser scanning

(1) Alignment: initial alignment between mesh model and shape-from-silhouette reconstruction using ICP (not in detail)

(2) Step A:3D flow-driven mesh tracking

(3) Marker selection from 3D flow estimation

(4) Step B: 3D flow-driven Laplacian mesh deformation/tracking

Deformable mesh tracking: Step A

Step A:3D flow-driven mesh tracking

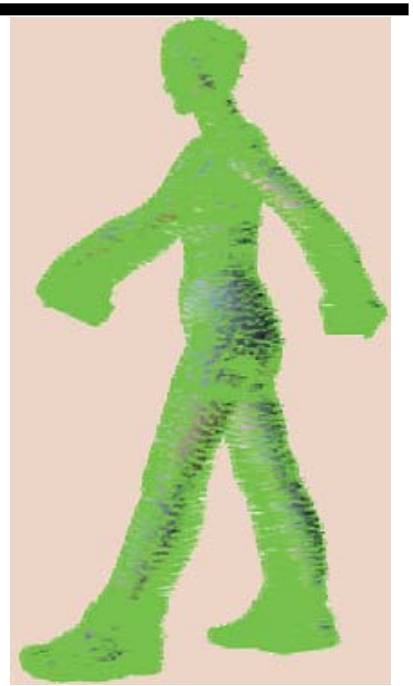
Generate the texture for the model using recorded images: $I_t^0 \cdots I_t^{K-1}$

Project the texture model to the camera views: $T_t^0 \cdots T_t^{K-1}$

Estimate the model vertex 3D flow: $\vec{f}(v_i) = (x_i, y_i, z_i)$

Filter 3D flow field with a low-pass Gaussian kernel on the valid vertices

Move the model vertices using the estimated flow field, accumulate flow: $\vec{d}_{\text{ACCUM}}(v_i) = \vec{d}_{\text{ACCUM}}(v_i) + \vec{f}(v_i)$



$E_{ov}(t+1) < TR_{ov}$

No

Yes

Keep the motion field

$\vec{d}(t, v_i) = \vec{d}(t-1, v_i) + \vec{d}_{\text{ACCUM}}(v_i)$

Deformable mesh tracking: marker selection

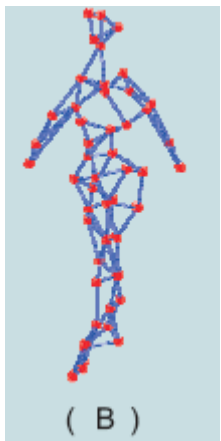
(1) Curvature based segmentation

(2) Candidate vertices are selected that are closed to the centroids of segments

(3) Compute the errors for the candidate vertices

$$tsc(v_i) = \frac{1}{N_F * K} \sum_{t=0}^{N_F} \sum_{k=0}^{K} (1 - PROJ_{sil}^{k}(p_i + \vec{d}(t, v_i), t)) \quad (1)$$

$$mov(v_i) = \frac{1}{N_F} \sum_{t=0}^{N_F} (\|\vec{d}(t, v_i) - \frac{1}{N_V} \sum_{j=0}^{N_V} \vec{d}(t, v_j)\|) \quad (2)$$



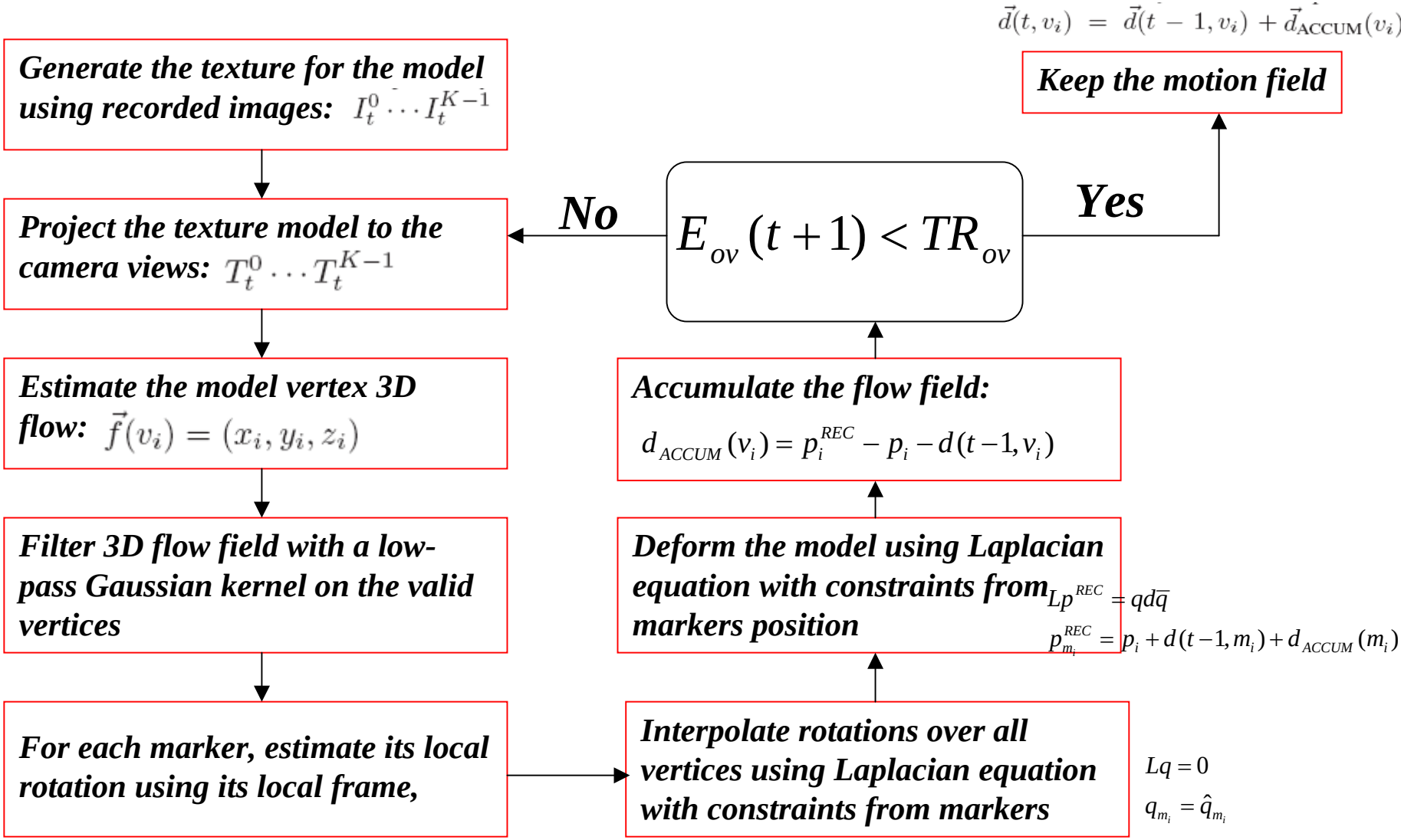
(4) Select marker vertices and create marker graph

$$tsc(v_i) < \mathrm{TR_{TSC}}$$

$$mov(v_i) < \mathrm{TR_{MOV}}.$$

Deformable mesh tracking: Step B

Step B:3D flow-driven Laplacian mesh tracking



Deformable mesh tracking: Results

METHOD	TIME	VOLCHG	MQLT	ERROR
RAWFL	109s	17.65%	0.46	98.66mm
ST-A	111s	4.97%	0.30	49.39mm
BR / ST-AB	111s	2.79%	0.035	26.45mm
BA	426s	2.77%	0.029	35.28mm
LK	89s	10.73%	1.72	76.24mm

Table 1. Different algorithmic alternatives are compared in terms of run time, volume change (*VOLCHG*), mesh quality (*MQLT*), and position error (*ERROR*). Our proposed pipeline with the dense optical flow method by Brox et al. (*BR/ST-AB*) leads to the best results.