Markerless Human Motion Capture
Graphics, Vision and Video - Interdisciplinary Topics in Visual Computing Seminar

Aurela Shehu
Saarland University
s9ausheh@stud.uni-saarland.de

June 14, 2012
Contents

1 Introduction

2 Markerless Human Motion Capture
   • ”Using a Sums of Gaussians Body Model”
   • ”Using Unsynchronized Moving Cameras”

3 Conclusions & Discussion
Human Motion Capture

process of analysing human movements from video data
Human Motion Capture Applications

(a) Movies

(b) Animation for Games

(c) Sport Science
Body Motion Capture Problem

- synchronized + calibrated cameras
- multi-view video sequences
- 3D Human Body Model

human body shape + pose for every frame
Method Overview (C. Stoll, N. Hasler, J. Gall, H. Seidel, C. Theobalt "Fast Articulated Motion Tracking using a Sums of Gaussians Body Model" (ICCV) 2011)
SoG-based Image domain and Body model

- $B_i$: Gaussian kernel
  - 3D case: 3D sphere
  - 2D case: 2D superpixel

\[
K(x) = \sum_{i=1}^{n} B_i(x)
\]

Color model $C = \{ c_i \}_i$
2D-2D SoG Similarity

- how to compare two SoG images?
- two SoG images $K_a, K_b$, associated color models $C_a, C_b$
- similarity
  - overlapping of Gaussians + image similarity

$$E(K_a, K_b, C_a, C_b) = \int_{\Omega} \sum_{i \in K_a} \sum_{j \in K_b} d(c_i, c_j) B_i(x) B_j(x) dx$$

$d(c_i, c_j)$ similarity measure between color models
Objective function

- goal: estimate pose-parameters $\Theta$ (position, angle joints) of kinematic skeleton from input images $I$
- given
  - $n_{\text{cam}}$ cameras $C_l$ with SoG images $(K_l, C_l)$
  - 3D body model $(K_m, C_m)$ parametrized by $\Theta$
- similarity function

$$E(\Theta) = \frac{1}{n_{\text{cam}}} \sum_{l=1}^{n_{\text{cam}}} \frac{1}{E(K_l, K_l)} E(K_l, \Psi_l(K_m(\Theta)), C_l, C_m)$$

- objective function

$$\mathcal{E}(\Theta) = E(\Theta) - w_l E_{\text{lim}}(\Theta) - w_a E_{\text{acc}}(\Theta)$$
Actor specified body model estimation

- manually initialize pose parameters $\Theta +$ estimate (refinement) of $\Theta$
- optimize shape parameters $\Theta_{shape}$ that define bone lengths, position, variance of each blob
- calculate Gaussian blob mean color $c_i$
Articulated Motion Tracking

- estimate current pose parameters
  - given estimated pose of the model in the previous frames
    \[
    \Theta_0^t = \Theta^{t-1} + \alpha(\Theta^{t-1} - \Theta^{t-2})
    \]

- optimizes parameters:
  maximize objective function
  \[
  \mathcal{E}(\Theta) = E(\Theta) - w_l E_{lim}(\Theta) - w_a E_{acc}(\Theta)
  \]

- conditioned gradient ascent
  \[
  \Theta_{i+1}^t = \Theta_i^t + \nabla E(\Theta_i^t) \circ \sigma_i
  \]

\[
\sigma_{i+1}^{(l)} = \begin{cases} 
\sigma_i^{(l)} \mu^+, & \text{if } \nabla E(\Theta_i^t) > 0 \\
\sigma_i^{(l)} \mu^-, & \text{if } \nabla E(\Theta_i^t) \leq 0 
\end{cases}
\]
Results

"Using a Sums of Gaussians Body Model"
Method Overview (Hasler et al. "Markerless Motion Capture with Unsynchronized Moving Cameras" (CVPR) 2009)
‘Using Unsynchronized Moving Cameras’

Diagram:

1. Camera Calibration using SfM
2. Camera Synchronization
3. Motion Capture
Single Camera Structure-from-Motion

- find corresponding feature points in consecutive frames (KLT-Tracker, SIFT-matching)
- filter out moving feature points, $p_{j,k}$ (RANSAC with multi-view constraints)
- estimate $3 \times 4$ camera matrix $A_k$ parameters
- determine 3D object point $P_j$
- Bundle adjustment:

$$\arg \min_{A_k, P_j} \sum_{j=1}^{J} \sum_{k=1}^{K} d(p_{j,k}, A_k P_j)^2$$
Multi-Camera Structure-from-Motion

- SfM for each camera
  - $N$ camera matrices reconstructions $A_{k,n}$, 3D object points $P_{j,n}$
- register $N$ reconstructions into a global coordinate system
  - estimate transformation $H$ between independent reconstructions
  - find and merge tracked in at least two cameras common 3D object points

$$\arg \min_{A,P} = \sum_{n=1}^{N} \sum_{j=1}^{J} \sum_{k=1}^{K} d(p_{j,k,n}, A_{k,n}P_{j,n})^2$$
3D Background Reconstruction

- estimate geometry of the static background of the scene
- reconstruction of a surface from a sparse set of point cloud $P_{j,n}$
- remove outliers that do not form surfaces (tensor voting filter)
- smooth out the remaining noise (bilateral moving least squares filtering)
- triangle mesh reconstruction
Camera Calibration using SfM

camera synchronization

Motion Capture
Synchronizing Audio Signals

- at least one sound source in the scene
- $\alpha_i$ : audio signal captured by the i-th camera
- cross correlation between the audio signal of cameras $i,j$ :
  \[ \alpha_i \ast \alpha_j \equiv \bar{\alpha}_i(-t) \ast \alpha_j(t) \]
- efficient computation of cross correlation using Fast Fourier Transform (FFT)
Synchronizing Audio Signals

- requirement: the observed scene is small
- audio delay between signals: peak of the cross correlation signal
Correction for Large Camera Displacements

- camera position $c_i$ known from the calibration step
- known position of the sound source $s$
- delay between audio signals of camera $i,j$

$$d_{ij} = \Delta_{ij} + \frac{1}{c} (d(c_j - s) - d((c_j - s))$$

- $N$ cameras, $N-1$ unknown $\Delta_i$ (temporal shift of every camera)
- $N(N-1)/2$ equations ($N-1$ linearly independent)
- extend to unknown source
"Using Unsynchronized Moving Cameras"
Kinematic Chains

- respective movement of a point $X_i$

$$X'_i = \exp(\theta \xi)(\exp(\theta_1 \xi_1) \ldots \exp(\theta_n \xi_n))X_i$$

- pose configuration $(6+n)$-D vector

$$\chi = (\hat{\xi}, \theta_1, \ldots, \theta_n) = (\hat{\xi}, \Theta)$$

- task: compute vector $\chi$ from calibrated and synchronized data
Silhouette Extraction

- image segmentation (level set function $\Phi \in \Omega \rightarrow \mathbb{R}$)
- minimize energy

\[
E(\Phi, p_1, p_2, \chi) = \lambda \int_{\Omega} (\Phi - \Phi_0(\chi))^2 dx - \\
\int_{\Omega} H(\Phi)\log p_1 + (1 - H(\Phi))\log p_2 + v|\nabla H(\Phi)| dx
\]

- output: segmentation of the images
Pose Estimation

- given image points on the contour line to reconstruct 3D projection rays
- projection ray: 3D plucker line $L_i = (n_i, m_i)$ ($3D$ unit direction $n_i$, $3D$ moment $m_i$)
- error function for each point-line pair
  $$X'_i(\hat{\xi}, \Theta) \times n_i - m_i = 0$$
- linearisation of equation
- iteration to optimize all correspondences simultaneously
Results

"Using Unsynchronized Moving Cameras"
Conclusions

- two methods on markerless human motion capture presented
- novelties
  - represent 3D body model, image domain as a Sum of Gaussians
  - exploit audio signal for camera synchronization
- initialization phase → actor-specified 3D body model
- online tracking, 2D SoG images computation, estimate parameters of kinematic model
- camera calibration, background reconstruction,
- camera synchronization, pose parameter estimation
Discussion

- object wear tight clothes
- two actors with the same clothes
- online tracking should start from one of the four "initialization" poses
- cannot faithfully model highly textured regions
- difficult to accurately track twisting motions
- cameras number greater or equal to 5
- outdoor scenes
3D-2D SoG Similarity

\[
E(K_I, \Psi(K_m), C_I, C_m) = \sum_{i \in K_I} \min\left( \left( \sum_{j \in \Psi(K_m)} E_{ij} \right), E_{ii} \right)
\]

- \(K_I\) : image model
- \(\Psi(K_m)\) : projected SoG model
Discussion

- large audio delays $\rightarrow$ inaccurate tracking
- fast movement $\rightarrow$ necessary prediction of subject’s motion
- sound source distinctness
- more than one actors
Future Work

- real-time human motion capture
- usage of cheap, low-quality user cameras
- minimize required cameras number
- flexible number of interacting actors in scene
- face motion analysis