SCAPE: Shape Completion and Animation of People

Introduction

- ➔ Graphics applications often require a complete surface model for rendering and animation.
- ➔ Obtaining a complete model of a particular person is often difficult or impossible.
- ➔ SCAPE — a data-driven method for building a unified model of human shape.

Introduction

Key features of the method :

➔ This method learns separate models of body deformation — one accounting for changes in pose and one accounting for differences in body shape between humans.

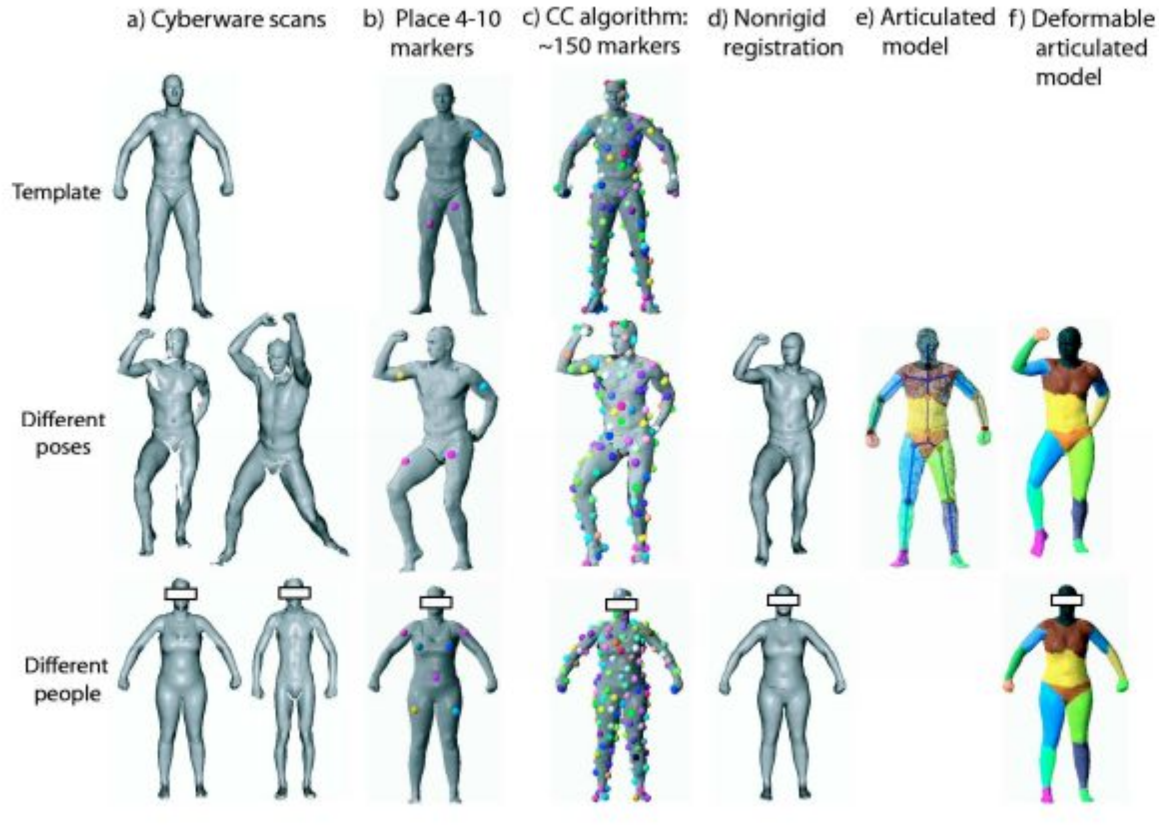
➔ The model provide a level of detail sufficient to produce dense full body meshes, and capture details such as muscle deformations of the body in different poses.

➔ Applied to two main tasks :
  - Partial view completion
  - Full 3D animation of human

Pipeline


Acquiring and Processing Data Meshes
Pose Deformation
Body-Shape Deformation
Shape Completion

Acquiring and Preprocessing Meshes



Acquiring and Preprocessing Meshes

Range Scanning

➔ Surface data is acquired using a Cyberware WBX whole-body scanner.

➔ The scanner captures range scans from four directions simultaneously.

➔ Two data sets are obtained:
  - **Pose data set :** scans of 70 poses of particular person in a wide variety of poses.
  - **Body data set :** scans of 37 different people in a similar (but not identical) pose.

Acquiring and Preprocessing Meshes

Range Scanning

➔ One mesh is selected from the pose data as the **Template Mesh.**

➔ All other meshes are called **Instance Meshes.**

➔ Template mesh acts as a reference mesh for all the instance meshes.

➔ The template mesh is hole-filled using an algorithm by **Davis et al.[2002].**

Acquiring and Preprocessing Meshes

Correspondence

➔ Bring the template mesh into correspondence with each of the other mesh instances.

➔ Markers are obtained using an algorithm called Correlated Correspondence (CC) **[Anguelov et al. 2005]**

➔ initialize the CC algorithm by placing 4–10 markers by hand on each pair of scans.

➔ The result of the algorithm is a set of 140–200 (approximate) correspondence markers between the two surfaces.

Acquiring and Preprocessing Meshes

Non-rigid Registration

➔ Given a set of markers between two meshes.

➔ The task is to bring the meshes into close alignment, while simultaneously aligning the markers.

➔ A standard algorithm **[Hahnel et al. 2003]** is applied to register the template mesh with all of the meshes in the data set.

Acquiring and Preprocessing Meshes

Recovering the Articulated Skeleton

- ➔ A skeleton is constructed for the template mesh automatically, using the algorithm of **[Anguelov et al. 2004]**.
- ➔ The algorithm exploits the fact that vertices on the same skeleton joint are spatially contiguous, and exhibit similar motion across the different scans.
- ➔ The algorithm automatically constructs a skeleton with 18 parts, which is converted to a skeleton with 16 parts by combining symmetric parts.

Acquiring and Preprocessing Meshes

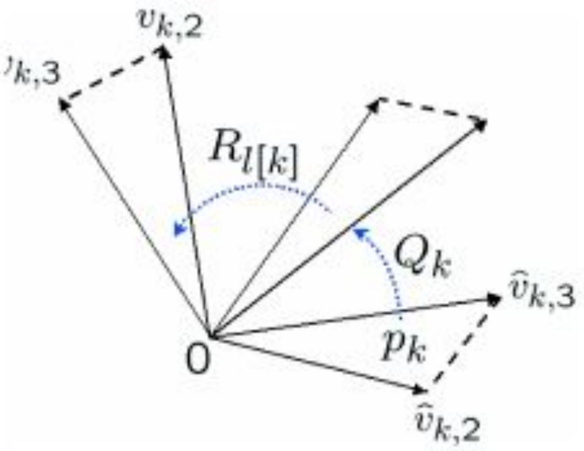
Data Format and Assumptions

- ➔ The resulting data set consists of a template mesh **X** and a set of instance meshes $\mathbf{Y = \{Y_1 ,\ldots,Y_N\}}$.
- ➔ The template mesh $\mathbf{X = \{V_X ,P_X \}}$ has a set of vertices $\mathbf{V_X = \{x_1,\ldots, x_M\}}$ and a set of triangles $\mathbf{P_X = \{p_1,\ldots, p_P\}}$
- ➔ Each instance mesh has the same set of points and triangles as the template mesh, so let $\mathbf{\{y_1^i ,\ldots, y_1^M\}}$ be the set of points in instance mesh $\mathbf{Y^i}$.
- ➔ For each mesh $\mathbf{Y^i}$, a set of absolute rotations $\mathbf{R^i}$ is defined for the rigid parts of the model, where $\mathbf{R_l^i}$ is the rotation of joint **l** in instance **i**.

Pose Deformation

- ➔ We want to model the deformations which align the template with each mesh Y^i.
- ➔ Let triangle p_k contain the points $x_{k,1}$, $x_{k,2}$, $x_{k,3}$.
- ➔ Translate the $x_{k,1}$ to the global origin.
- ➔ Deformations will be applied to the triangle edges $\mathbf{v_{k,j} = x_{k,j} - x_{k,1}\ ;\ j = 2,3}$.
- ➔ First, we apply a 3×3 linear transformation matrix $\mathbf{Q^i_k}$ to the triangle.
- ➔ The deformed polygon is then rotated by $\mathbf{R^i_l}$.

Pose Deformation



$$v^i_{k,j} = R^i_{\ell[k]} Q^i_k \hat{v}_{k,j}, \quad j = 2, 3$$

Pose Deformation

Learning the pose deformation model

➔ Initial objective : $\operatorname*{argmin}_{y_1,\ldots,y_M} \sum_k \sum_{j=2,3} \|R^i_{\ell[k]} Q^i_k \hat{v}_{j,k} - (y_{j,k} - y_{1,k})\|^2$

➔ Joint rotations are represented with their twist coordinates.

➔ Let **M** denote any 3 × 3 rotation matrix, and let $\mathbf{m}_{\mathbf{ij}}$ be its entry in i-th row and j-th column. The twist **t** for the joint angle is a 3D vector, and can be computed from the following formula :

$$
\begin{aligned}
t &= \frac{\|\theta\|}{2\sin\|\theta\|}\begin{bmatrix} m_{32}-m_{23} \\ m_{13}-m_{31} \\ m_{21}-m_{12} \end{bmatrix} \\
\text{with } \theta &= \cos^{-1}\left(\frac{\operatorname{tr}(M)-1}{2}\right).
\end{aligned}
$$

Pose Deformation

Learning the pose deformation model

➔ Learn a regression function for each triangle p_k, which predicts the transformation matrices $\mathbf{Q}^i_k$ as a function of the twists of its two nearest joints, i.e.

$$\triangle r^i_{\ell[k]} = (\triangle r^i_{\ell[k],1}, \triangle r^i_{\ell[k],2})$$

➔ Associate a 7×1 regression vector $\mathbf{a}_{k,lm}$ with each of the 9 values of the matrix Q, and write :

$$q^i_{k,lm} = \mathbf{a}^T_{k,lm} \cdot \begin{bmatrix} \triangle r^i_{\ell[k]} \\ 1 \end{bmatrix} \qquad l,m = 1,2,3$$

Pose Deformation

Learning the pose deformation model

➔ Our goal now is to learn these parameters $\mathbf{a}_{k,lm}$.

$$\underset{\mathbf{a}_{k,lm}}{\operatorname{argmin}} \sum_i \left([\triangle r^i \;\; 1]\mathbf{a}_{k,lm} - q^i_{k,lm} \right)^2$$

➔ Unfortunately, the transformations $\mathbf{Q}^i_k$ for the individual triangles are not known.

➔ We follow **Sumner et al. [2004**] and **Allen et al. [2003]**, and introduce a smoothness constraint which prefers similar deformations in adjacent polygons that belong to the same rigid part.

$$\underset{\{Q^i_1,\ldots,Q^i_P\}}{\operatorname{argmin}} \quad \sum_k \sum_{j=2,3} \|R^i_k Q^i_k \hat{v}_{k,j} - v^i_{k,j}\|^2 + \\ w_s \sum_{k_1,k_2 \text{ adj}} I(\ell_{k_1} = \ell_{k_2}) \cdot \|Q^i_{k_1} - Q^i_{k_2}\|^2,$$

Body-Shape Deformation

➔ Body-shape variation is applied independently of the pose variation, by introducing a new set of linear transformation matrices $\mathbf{S}^i{}_k$, one for each instance i and each triangle k.

$$v^i_{k,j} = R^i_{\ell[k]} S^i_k Q^i_k \hat{v}_{k,j}.$$

➔ Similarly, objective function is given by :

$$\underset{S^i}{\operatorname{argmin}} \sum_k \sum_{j=2,3} \| R^i_k S^i_k Q^i_k \hat{v}_{k,j} - v^i_{k,j} \|^2 + w_s \sum_{k_1, k_2 \text{ adj}} \| S^i_{k_1} - S^i_{k_2} \|^2.$$

Shape Completion

- We are given sparse information about an instance mesh, and wish to construct a full mesh consistent with this information.
- Assume we have a set of markers $Z = z_1,\ldots,z_L$ which specify known positions in 3D for some points $x_1,\ldots, x_L$ on the model mesh.
- We want to find the set of points Y that best fits these known positions, and is also consistent with the SCAPE model.

$$E_H[Y] + w_Z \sum_{l=1}^{L} \|y_l - z_l\|^2$$

Limitations

- ➔ Mesh acquisition process is difficult and complex.
- ➔ Suffers highly due to occlusions.
- ➔ Deformations resulting from other factors are not encoded, like deformation due to pure muscle activity.
- ➔ Tissue perturbations due to motion (like fat wiggling) is not captured.