DRAPE: DRessing Any PErson

Peng Guan[1,*] Loretta Reiss[1,*] David A. Hirshberg[2,†] Alexander Weiss[1,*] Michael J. Black[1,2,†, ‡]

[1]Dept. of Computer Science, Brown University, Providence [2]Max Planck Institute for Intelligent Systems, Tübingen

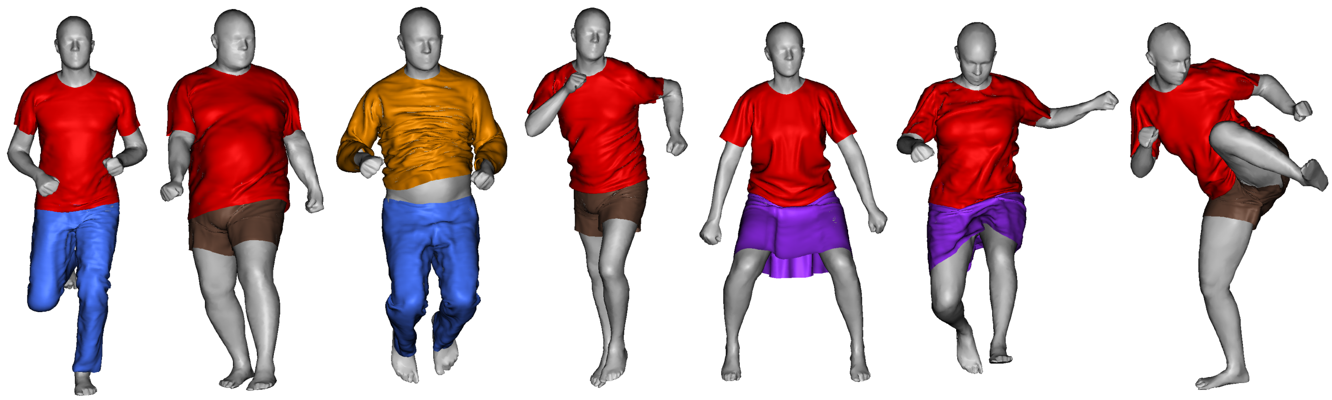


Figure 1: DRAPE *is a learned model of clothing that allows 3D human bodies of any* **shape** *to be dressed in any* **pose***. Realistic clothing shape variation is obtained without physical simulation and dressing any body is completely automatic at run time.*

Abstract

We describe a complete system for animating realistic clothing on synthetic bodies of any shape and pose without manual intervention. The key component of the method is a model of clothing called DRAPE (DRessing Any PErson) that is learned from a physics-based simulation of clothing on bodies of different shapes and poses. The DRAPE model has the desirable property of "factoring" clothing deformations due to body shape from those due to pose variation. This factorization provides an approximation to the physical clothing deformation and greatly simplifies clothing synthesis. Given a parameterized model of the human body with known shape and pose parameters, we describe an algorithm that dresses the body with a garment that is customized to fit and possesses realistic wrinkles. DRAPE can be used to dress static bodies or animated sequences with a learned model of the cloth dynamics. Since the method is fully automated, it is appropriate for dressing large numbers of virtual characters of varying shape. The method is significantly more efficient than physical simulation.

CR Categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation; I.6.8 [Simulation and Modeling]: Types of Simulation—Animation;

Keywords: Clothing animation, dressing, fit, wrinkles, virtual try-on, deformable model, learning

Links: DL PDF

1 Introduction

Clothed virtual characters in varied sizes and shapes are necessary for film, gaming, and on-line fashion applications. Dressing such characters is a significant bottleneck, requiring manual effort to design clothing, position it on the body, and simulate its physical deformation. DRAPE handles the problem of automatically dressing realistic human body shapes in clothing that fits, drapes realistically, and moves naturally. Recent work models clothing shape and dynamics [de Aguiar et al. 2010; Feng et al. 2010; Wang et al. 2010] but has not focused on the problem of fitting clothes to different body shapes.

Physics Based Simulation (PBS) [Baraff and Witkin 1998; Choi and Ko 2002; Bridson et al. 2002] is widely used to model the complex behavior of cloth and can produce highly realistic clothing simulations. An extensive survey of cloth simulation can be found in [Choi and Ko 2005]. PBS, however, is computationally expensive [de Aguiar et al. 2010; Choi and Ko 2005] and the results are specific to a particular body model. Dressing bodies of different shapes requires a separate simulation for every body shape. Additionally, a fundamental problem confronting garment designers is the non-trivial task of choosing clothing sizes and initializing clothing simulation on 3D characters [Choi and Ko 2005].

Our method learns a deformable clothing model that automatically adapts to new bodies. Once the DRAPE model is learned for a particular type of clothing, we can dress any body in that clothing. Unlike the PBS methods, users do not need to choose proper sizes and initial positions of cloth pieces before clothing fitting. The model will reshape the garment to fit the body and "drape" it automatically. Pattern design is completely separated from the process of dressing bodies and can be done by professional pattern makers before training the model. Therefore, users do not need to know about pattern design, enabling much broader applications of clothing animation.

Here we use SCAPE [Anguelov et al. 2005] to represent human bodies of different shapes in different poses. We learn separate SCAPE models for men and women using approximately 2000 aligned laser range scans of different people in a single pose [Robinette et al. 2002] and additional scans of one individual in roughly

* {pguan, ljr, aweiss}@cs.brown.edu
† {david.hirshberg, black}@is.mpg.de
‡Disclosure: MJB is a founder and shareholder of Any Body Inc., which has plans to commercialize 3D body shape technology.

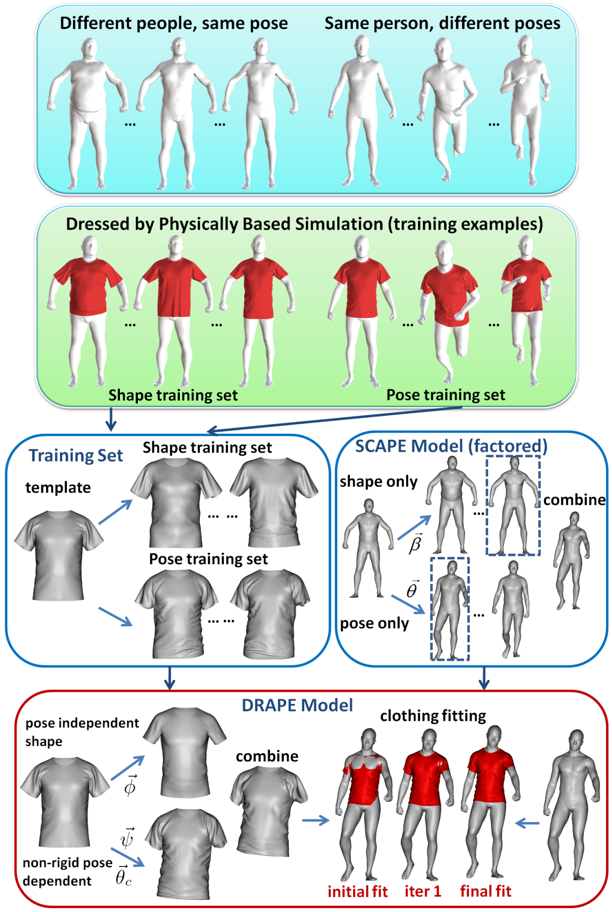


Figure 2: Overview. *We dress the bodies in the shape and pose training sets using PBS to generate clothing examples for learning. DRAPE factors rigid pose, pose-independent shape variation, and pose-dependent wrinkle deformation. The SCAPE model is used to represent the underlying naked body. Given an input body, an appropriate clothing configuration is generated according to the body pose and shape. The clothing fitting process eliminates cloth-body interpenetration to create realistic animations.*

70 poses. This results is an expressive 3D model with parameters controlling a wide range of body shapes ($\vec{\beta}$) and poses ($\vec{\theta}$).

For this study, we designed and graded patterns for five common clothing types: T-shirts, shorts, skirts, long sleeved shirts, and long pants [Beazley and Bond 2003]. The complete system (Figure 2) has three components:

1. Training data: The *shape training set* consists of SCAPE bodies with different shapes in the same pose. The *pose training set* contains a single body shape moving through sequences of poses. For each training body shape, we manually choose a size for each garment and dress the body using PBS (Figure 2, row 2); this becomes our training data.

2. Learned clothing deformation model: For each garment, we learn a *factored clothing model* that represents: *i*) rigid rotation, $\vec{\theta}_c$, of cloth pieces, e.g. the rotation of a sleeve w. r. t. the torso; *ii*) pose-independent clothing shape variations, $\vec{\phi}$, that are linearly predicted from the underlying body shape, $\vec{\beta}$, (learned from the shape training set); and *iii*) pose-dependent non-rigid deformations, $\vec{\psi}$, that are linearly predicted from a short history of body poses and clothing shapes (learned from the pose training set).

3. Virtual fitting: First, we map body shape parameters, $\vec{\beta}$, to clothing shape parameters, $\vec{\phi}$, to obtain a custom shaped garment for a given body. Clothing parts are associated with body parts and the pose of the body is applied to the garment parts by rigid rotation. The learned model of pose-dependent wrinkles is then applied. The custom garment is automatically aligned with the body and interpenetration between the garment and the body is removed by efficiently solving a system of linear equations.

Our model factors the change of clothing shape due to rigid limb rotation, pose-independent body shape, and pose-dependent deformations. As with the original SCAPE model, this allows us to combine deformations induced by different causes. The factored model can be learned from far less data than a model that simultaneously models clothing shape based on pose and body shape. In contrast, training a non-factored model (with pose, shape, and pose-dependent shape intermingled) would require a huge training set with many body shapes performing many motions. The factored model is an approximation that is sufficient for many applications and separates modeling body shape from pose-dependent shape. The method is ideal for applications where the body shape is not known in advance such as on-line virtual clothing try-on where every user has a unique 3D shape or where many different people must be animated (e.g. crowd scenes).

In summary, DRAPE makes the following contributions: 1) Synthetic bodies of any shape are automatically dressed in any pose. 2) A factored model of clothing shape models pose-dependent wrinkles separately from body shape. 3) The method dresses bodies completely automatically at run time. 4) Interpenetration is efficiently handled by solving a linear system of equations.

2 Related Work

Clothing simulation has many applications and no single technique will be appropriate for all of them. Here, we focus on full automation and reasonable realism for applications in gaming, virtual fashion, on-line retail clothing, computer vision, etc. We do not address high-quality, custom, or labor- and compute-intensive simulation.

Cloth Simulation. The extensive literature on cloth simulation focuses on modeling the physical properties of cloth and developing stable methods that can deal with cloth collisions, friction, and wrinkle buckling [Baraff and Witkin 1998; Bridson et al. 2002; Bridson et al. 2003; Choi and Ko 2002; Goldenthal et al. 2007]; see [Choi and Ko 2005; House and Breen 2000; Nealen et al. 2006] for surveys. These methods produce realistic simulations, but at high computational cost. Games and retail clothing applications, however, require efficient solutions because of their interactive nature. Efficient approaches include the Verlet integration scheme [Jakobsen 2001] and GPU acceleration [Bordes et al. 2009].

Cloth Capture. Structured light, stereo, optical flow, special patterns and multi-camera systems can be used to capture cloth undergoing natural motion [de Aguiar et al. 2008; Bradley et al. 2008; Pritchard and Heidrich 2003; Scholz et al. 2005; Stark and Hilton 2007; White et al. 2007]. These techniques do not immediately provide a way to re-purpose the garments to new sizes and poses but could be used to provide training data to a method like ours. There has been recent interest in using such real cloth motions to learn models of cloth deformation and, in particular, wrinkles [Popa et al. 2009; Cordier and Magnenat-Thalmann 2005]. For training, DRAPE requires aligned clothing meshes of different sizes; these are difficult to obtain from scanned garments. Instead, we simulate training clothing using PBS, giving a known alignment between all training instances (cf. [Wang et al. 2010]).

From 2D Patterns to 3D Fitting. Choi and Ko [2005] summarize the major challenges in cloth simulation and the "non-intuitive task of clothing a 3D character with a garment constructed from 2D patterns". There has been relatively little work to address this issue. Cordier et al. [2003] describe a web application that allows users to interactively adjust a 3D mannequin according to a shopper's body measurements and then resize and fit a garment to the body. Decaudin et al. [2006] describe a system in which the users draw 2D sketches of contours and lines on a virtual mannequin, and then the system converts these to 3D surfaces. Umetani et al. [2011] propose an interactive tool for bidirectional editing between a 2D clothing pattern and a 3D draped form. Our approach is different in that all our effort is up front; once a garment is designed, we automate the process of converting it to an infinitely resizable 3D model. Fitting to a new body is then fully automated.

Modeling Wrinkles. Wrinkles are important for producing realistic visual effects, therefore numerous wrinkle generation algorithms have been proposed. One class of methods deforms and blends wrinkles drawn on top of a smooth garment in a set of static poses to synthesize wrinkles as the garment deforms [Hadap et al. 1999; Cutler et al. 2007]. Another approach separates the coarse clothing shape from the fine wrinkle details [Cordier and Magnenat-Thalmann 2002; Cordier and Magnenat-Thalmann 2005; Müller and Chentanez 2010; Feng et al. 2010; Wang et al. 2010; Rohmer et al. 2010; Kavan et al. 2011]. The coarse shape is obtained by running PBS on a low-resolution version of the mesh. Appropriate fine wrinkles are synthesized using example- or learning-based methods. Our approach shares ideas with these methods but goes beyond previous work to address how wrinkles vary with body *shape*.

Clothing in Reduced Space. Instead of dealing with the mesh at the triangle level, related work models complex deformations in a lower-dimensional linear subspace [de Aguiar et al. 2010; James and Twigg 2005; Kavan et al. 2010]. This achieves a huge speed up but with reduced realism. These subspace methods replace cloth simulation with a learned dynamical system, where the input is a 3D body and its motion, and the output is a clothing mesh. Clothing pose and shape are integrated in such models and no separate control is provided. We take a similar, learning-based, approach but extend this idea to include wrinkles that also depend on body shape. The idea of modeling clothing as a low-dimensional deformation is also explored in [Guan et al. 2010]. The authors learn an "eigen clothing" model that describes how clothing deviates from the body as a function of body shape and pose. That work, however, only addresses the case of bodies and clothing in 2D. Kim and Vendrovsky [2008] use the underlying human pose to drive the deformation of clothing. Linear interpolation is used to compute the deformation of clothing for an unseen body pose using clothing examples from a database of pre-computed body-clothing pairs. Our method also uses the underlying body to drive clothing deformation, however, we go beyond their work to additionally drive clothing deformation based on body shape.

3 Simulating Clothing for Training

Learning a DRAPE model requires a shape training set of clothing meshes fit to different body shapes and a pose training set with a single template body in multiple poses. To prepare the training sets, we created 2D graded patterns for T-shirts, shorts, long-sleeved shirts, long pants, and skirts using a commercial design and 3D simulation software (OptiTex International, Israel). Without loss of generality we will use a T-shirt to illustrate the procedures for data generation. Figure 3 illustrates this standard commercial design process while Figure 4 shows examples of the training garments.

A garment is defined in 2D by a number of "grading points"(purple points in Figure 3), which model different sizes [Beazley and Bond 2003]. *Different sizes of the same garment are not simply scaled versions of each other.* Simulation of the garment first requires manually selecting the appropriate size pattern and positioning the clothing pieces in 3D. PBS is done here with OptiTex, but any simulation method could be used. Note, we select one 2D size as the template pattern and align all other 2D sizes to this with the help of the grading points. After 2D alignment, all 3D meshes for each type of clothing are in full correspondence.

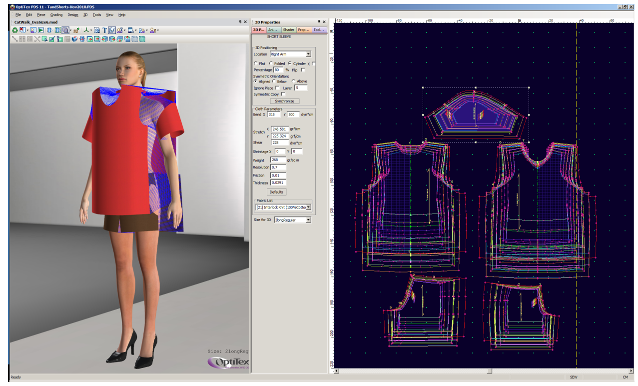


Figure 3: Pattern design. *This screen-shot from a commercial pattern design software system (OptiTex) shows graded patterns for the T-shirt and shorts, with the grade points highlighted as purple dots. Some pieces, such as the sleeves, may be reused for both left and right sides. The center panel controls the parameters of the cloth simulation. On the left, the initial cloth placement is shown with the blue lines indicating the points to be stitched during simulation.*

The training set for *pose-dependent clothing deformation* uses a single male and a single female avatar represented as SCAPE body models [Anguelov et al. 2005]; we use the average male and average female body shapes in the North American CAESAR data set [Robinette et al. 2002]. Using 23 different motion capture sequences we animate the SCAPE avatars and use OptiTex to simulate the clothing in each frame (Figure 4 (middle)). These motion sequences capture a wide range of body poses and include walking, running, jumping, kicking, turning, bending the legs, and so on. For each sequence we simulate different clothing types: T-shirt, shorts, and skirt for the female and T-shirt, shorts, long sleeves, and long pants for the male. The clothing pose training sets consist of more than 3500 different poses with 4 male garments and 3 female garments, for a total of $3500 \times 7 = 24,500$ clothing instances. The model for each clothing type is learned separately. The learned DRAPE model is able to combine upper and lower-body clothing models to produce combinations not seen in training.

Finally, for the *shape training sets*, we used the SCAPE body model to generate 60 males and 60 females that span a wide variety of body shapes. Each body is in the same canonical "template" pose shown in Figure 4 (bottom). Similar to the pose training set, we simulated 4 male garments and 3 female garments resulting in $60 \times 7 = 420$ clothing instances in the shape training set.

4 DRAPE Model

DRAPE is trained using a set of aligned 3D clothing meshes, with T triangles and V vertices. The set contains a template mesh $\mathcal{X}$, a set of *pose* examples $\mathcal{Y} \in \mathbf{P}$, and a set of *shape* example meshes $\mathcal{Y} \in \mathbf{S}$. $\mathcal{X}$ is obtained by dressing an average body in the template pose. $\mathcal{Y} \in \mathbf{P}$ are obtained by running clothing simulation on one animated body. $\mathcal{Y} \in \mathbf{S}$ are obtained by running clothing simulation

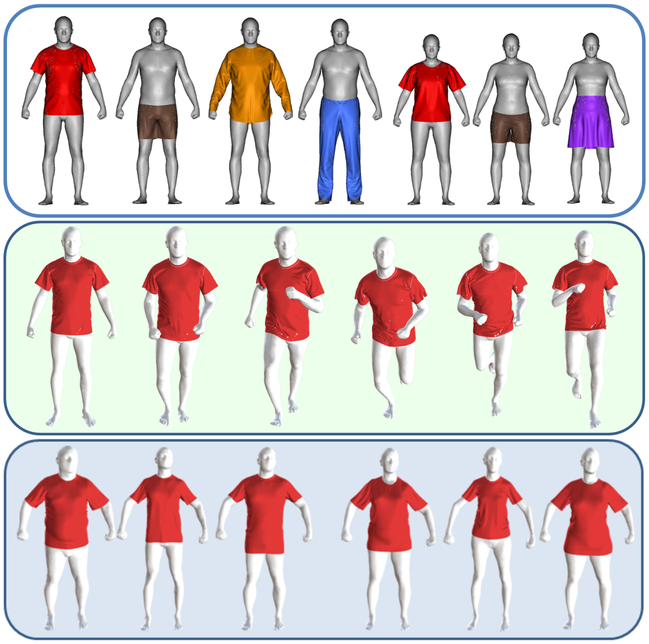


Figure 4: Examples of training data. *(top) Clothing types used here (T-shirt, shorts, long-sleeved shirt, long pants, and skirt). (middle) Example T-shirts in the* **pose** *training set generated from a representative motion sequence. (bottom) Training examples of T-shirts on representative male and female body* **shapes***.*

on different bodies with the same pose as the template. We consider males and females separately.

Since factorization is a crucial property of the model, we use *shape deformation gradients* [Sumner and Popović 2004; Anguelov et al. 2005] to represent deformations between meshes. This allows DRAPE to separate deformations induced by pose and shape and then combine the deformations together. We follow the formulation of SCAPE and present the notation here as it will be needed later. We refer the reader to the above referenced papers for details.

Deformation gradients are linear transformations that align corresponding triangles between a source mesh $\mathcal{X}$ and target mesh $\mathcal{Y}$ with the same topology. Suppose the vertices of a given triangle t in $\mathcal{X}$ are $(\vec{x}_{t,1}, \vec{x}_{t,2}, \vec{x}_{t,3})$ and the corresponding triangle in $\mathcal{Y}$ has the vertices $(\vec{y}_{t,1}, \vec{y}_{t,2}, \vec{y}_{t,3})$. We solve for a 3 by 3 linear transformation A_t such that

$$A_t[\Delta\vec{x}_{t,2}, \Delta\vec{x}_{t,3}, \Delta\vec{x}_{t,4}] = [\Delta\vec{y}_{t,2}, \Delta\vec{y}_{t,3}, \Delta\vec{y}_{t,4}], \qquad (1)$$

where $\Delta\vec{x}_{t,k} = \vec{x}_{t,k} - \vec{x}_{t,1}$ for $k = 2, 3$ and $\Delta\vec{x}_{t,4} = \Delta\vec{x}_{t,2} \times \Delta\vec{x}_{t,3}$. Since A_t is applied to *edge vectors*, it is translation invariant; it encodes the scale, orientation, and skew of triangle t. Following [Sumner and Popović 2004] the virtual edge, $\Delta\vec{x}_{t,4}$, makes the problem well constrained so that we can solve for A_t.

The key idea of a factored model is that it expresses the deformations, A_t, as a series of linear transformations, each corresponding to different aspects of the model. We factor A_t into pose-dependent deformation, rigid part rotation, and body shape deformation:

$$A_t = Q_t R_{p(t)} D_t. \qquad (2)$$

D_t represents variations in clothing shape on different people and is triangle specific. $R_{p(t)}$ is the rigid rotation applied to clothing part p containing triangle t. Q_t is the triangle-specific non-rigid pose-dependent deformation of the garment. This pose-dependent term captures wrinkles resulting from bending and twisting. The order of the factoring matters. D_t is learned from a shape training set where all the bodies are in a template pose, thus D_t is applied first, when clothing is still in the template pose. We then rotate each of the parts and finally apply wrinkle deformations on top of the previous deformations.

DRAPE models different clothing meshes by applying different transformations D_t, $R_{p(t)}$, and Q_t to the template mesh. The deformations, however, are applied to triangles independently and do not guarantee a consistent mesh. Reconstructing the final mesh involves solving for the vertex coordinates, $\vec{y}_i \in \mathcal{Y}$, that best match the deformed triangles in a least squares sense

$$\underset{\vec{y}_1,\ldots,\vec{y}_V}{\operatorname{argmin}} \sum_{t=1}^{T} \sum_{k=2,3} ||Q_t R_{p(t)} D_t \Delta\vec{x}_{t,k} - \Delta\vec{y}_{t,k}||^2. \qquad (3)$$

Figure 5 illustrates each of the DRAPE deformations applied in order. Below we describe them in detail and, in particular, how we learn Q_t and D_t.

4.1 Deformations Due to Body Shape

The shape deformations D_t are learned from $\mathcal{X}$ and $\mathbf{S}$. Recall that the examples in $\mathbf{S}$ have the same pose as $\mathcal{X}$. We solve for the A_t's for each pair of $\mathcal{X}$ and $\mathcal{Y}^j \in \mathbf{S}$ using Equation (1). These deformations are induced by changes in clothing shape that result only from the clothing being draped over different body shapes, so $Q_t R_{p(t)}$ in Equation (2) is the identity and, for a given mesh $\mathcal{Y}^j \in \mathbf{S}$, we can write $A_t^j = D_t^j$. The clothing shape deformations D_t^j for all triangles $t = 1 \ldots T$ are concatenated into a single column vector $\vec{d}^j \in \mathbb{R}^{3 \cdot 3 \cdot T \times 1}$. These are collected into a matrix of deformations $S = [..., \vec{d}^j, ...]$. Principal component analysis (PCA) is used to find a low dimensional subspace, such that $\vec{d}^j$ can be approximated by $U_d\vec{\phi}^j + \vec{\mu}_d$, where U_d is a matrix of the first few principal components of the shape deformation space, and $\vec{\mu}_d$ represents the mean deformation from the template $\mathcal{X}$. Figure 6 illustrates the mean and first three principal components for a female T-shirt.

A new clothing shape is represented by a new set of shape coefficients $\vec{\phi}^*$. These define the shape deformation from the template, $\vec{d}^* = U_d\vec{\phi}^* + \vec{\mu}_d$. This is converted into the appropriate 3×3 deformation matrices, D_t^*, which are applied to the template as illustrated in Figure 5.

The key idea behind automatically dressing a new body is that we can predict the clothing shape parameters, $\vec{\phi}^*$, from a SCAPE body with shape parameters, $\vec{\beta}$ (refer to Figure 2). Given 60 body and clothing training pairs in $\mathbf{S}$, we learn a linear mapping, W, between these vectors using L2-regularized least squares with the weight of the regularized term being 0.2. We then predict clothing parameters for an input body shape $\vec{\beta}$ using the linear equation

$$\vec{\phi}^* = W \cdot \begin{bmatrix} \vec{\beta} \\ 1 \end{bmatrix}. \qquad (4)$$

Since clothing shape deformations are a function of body shape, we write $\hat{D}_t(\vec{\beta})$ to represent the deformation matrix for a triangle t predicted from the body shape given by $\vec{\beta}$. In our work, $\vec{\beta}$ is 20 dimensional and $\vec{\phi}^*$ has only 5 dimensions because we expect the shape model to only contain low frequency deformations.

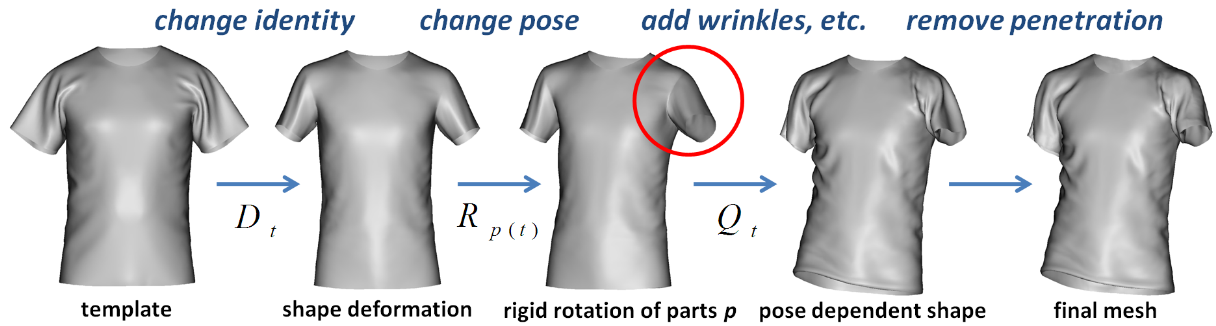


Figure 5: DRAPE clothing deformation process. *(1) The template mesh is deformed to fit a new body shape. (2) The pose of the underlying body is used to apply a rotation to clothing parts. (3) Pose-dependent non-rigid deformation produces wrinkles learned from examples. (4) Vertices are moved locally to remove interpenetration with the underlying body mesh.*

(a) Mean (b) PC1 (c) PC2 (d) PC3

Figure 6: Shape model. *Deviations from the template shape: (a) template deformed by the mean deformation to create a "mean template"; (b-d) mean template deformed along the first three principal component directions ($\pm$ 3 standard deviations).*

Figure 7: Color-coded body and clothing. *The colors show the part correspondences between bodies and clothing. During training, the rigid rotation for each clothing part is the same as the rotation for the corresponding body part. This allows us to transfer a new body pose to the clothing during clothing fitting.*

Figure 8: Learned pose-dependent deformation model. *For each pair, the left piece of clothing shows the physically-simulated example from the pose training set, and the right piece shows the synthesized deformation patterns predicted by our model.*

4.2 Deformations Due to Rigid Part Rotation

The SCAPE body model is composed of parts, which are color coded in Figure 7. Clothing is also naturally composed of parts during its design or can be easily segmented into parts. Each clothing part is associated with a single body part as shown by the color coding in Figure 7. The part correspondences for each garment are defined manually as part of the pattern creation process.

The SCAPE pose is given by the parameters $\vec{\theta}$ (refer to Figure 2); these are relative part rotations along a kinematic tree rooted at the pelvis. These parameters represent rigid 3×3 rotation matrices, R_p for each part p; these are applied to all the triangles in the respective body part. The DRAPE model simply applies these rotations to the corresponding clothing part as defined in Figure 7. For a given garment, all the part rotation parameters relevant to that garment are collected into a clothing pose vector $\vec{\theta}_c$. The part-based rotation for a clothing mesh triangle is denoted as $R_{p(t)}(\vec{\theta}_c)$.

4.3 Deformations Due to Body Pose

We use the pose training set $\mathbf{P}$ to learn a non-rigid pose-dependent clothing deformation model; this captures effects such as wrinkles. Since every $\mathcal{Y}^i \in \mathbf{P}$ corresponds to the same SCAPE body shape, all clothing deformations result from pose changes. This means D_t is the identity in Equation (2) and we write the deformations for each mesh $\mathcal{Y}^i$ and each triangle as $A_t^i = Q_t^i R_{p(t)}(\vec{\theta}_c^i)$, where Q_t^i is the residual triangle deformation after accounting for the part-based rigid rotation $R_{p(t)}(\vec{\theta}_c^i)$ given by the training body pose. Since all the clothing meshes in $\mathbf{P}$ are in correspondence, it is trivial to solve for A_t^i and consequently the non-rigid deformations, Q_t^i.

As with the shape deformation model, the clothing pose deformations, Q_t^i, for all the triangles are concatenated into a single column vector, $\vec{q}^{\,i} \in \mathbb{R}^{3 \cdot 3 \cdot T \times 1}$. We collect every example $\mathcal{Y}^i$ in $\mathbf{P}$ to form a matrix $P = [..., \vec{q}^{\,i}, ...]$. We use PCA to represent a dimensionality-reduced subspace of pose deformation and $\vec{q}^{\,i}$ is approximated by

$U_q\vec{\psi}^i + \vec{\mu}_q$. Depending on the complexity of the clothing type, $\vec{\psi}^i$ is chosen to have $30-50$ dimensions capturing 90% of the variance.

While PCA captures the space of possible deformations, to animate clothing we must relate these deformations to body pose. Cloth exhibits complex dynamical phenomenon w.r.t. the movement of underlying human body. To realistically capture how cloth moves and wrinkles, we learn a second order dynamics model for pose-dependent wrinkle deformation using the method described in [de Aguiar et al. 2010]; refer to that paper for a detailed explanation. The second-order model is important to capture smooth wrinkle transitions with pose variation and fine wrinkle details.

As an example, consider the T-shirt, where $\vec{\theta}_c$ contains three relative part rotations: torso w. r. t. the pelvis, left upper arm w. r. t. the torso, and right upper arm w. r. t. the torso. Each of the part rotations are represented by a 3×1 Rodrigues vector. Therefore, $\vec{\theta}_c$ is 9 dimensional in this case.

The key idea is to write the pose deformation coefficients , $\vec{\psi}^f$, of the current frame, f, as a function of the pose, history of pose-dependent deformations, and body state changes; i.e. $\vec{\psi}^f =$

$$M_1\vec{\theta}_c^f + M_2\vec{\psi}^{f-1} + M_3\vec{\psi}^{f-2} + M_4\vec{z}^{f,f-2} + M_5\vec{z}^{f-1,f-2}, \quad (5)$$

where $M_1..M_5$ are the matrices of the dynamics coefficients to be learned, $\vec{\theta}_c^f$ is a vector of the relevant clothing part rotations at frame f, $\vec{\psi}^{f-1}$ and $\vec{\psi}^{f-2}$ are the previous two frames of pose deformation coefficients. $\vec{z}^{j,k} = \begin{bmatrix} \Gamma^{k-1} \cdot (\vec{\tau}^j - \vec{\tau}^k) \\ \vec{\theta}_c^j - \vec{\theta}_c^k \end{bmatrix}$ encodes the relative body translation (normalized by the global rotation at frame k) and rotation change of frame j with respect to frame k, where Γ^k is the global (i. e., pelvis) rotation of the body at frame k, $\vec{\tau}^j$ and $\vec{\tau}^k$ are the global translations of the body. We normalize the position change between two frames so that the model generalizes better to unseen body movement directions. Note that $\vec{\theta}_c^j, \vec{\theta}_c^k$ are relative part rotations, so that they do not need to be normalized by Γ^k.

We learn a gender-specific dynamics model for each type of clothing. Given the training poses, $\mathbf{P}$, the dynamics coefficients $M_1..M_5$ are learned by solving the following least squares problem constructed from the pose training set:

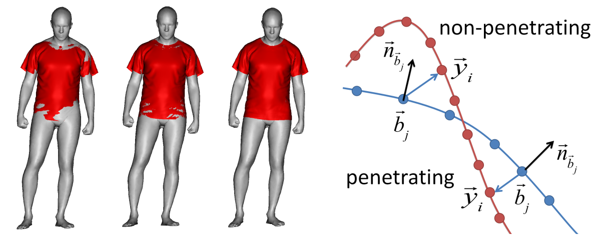
$$\underset{M_1,M_2,M_3,M_4,M_5}{\operatorname{argmin}} \sum_{f=1}^{|\mathbf{P}|} ||\vec{\psi}^f - \begin{bmatrix} M_1^T \\ M_2^T \\ M_3^T \\ M_4^T \\ M_5^T \end{bmatrix}^T \begin{pmatrix} \vec{\theta}_c^f \\ \vec{\psi}^{f-1} \\ \vec{\psi}^{f-2} \\ \vec{z}^{f,f-2} \\ \vec{z}^{f-1,f_2} \end{pmatrix} ||^2. \quad (6)$$

Once $\vec{\psi}^f$ for frame f is predicted from the learned dynamics model using Equation (5), the concatenated pose-dependent deformations will be $\vec{q} = U_q\vec{\psi}^f + \vec{\mu}_q$. Again, this is converted into the appropriate 3×3 deformation matrices. Let $\hat{Q}_t(\vec{\psi}^f)$ represent the deformation matrix for a triangle t. We show in Figure 8 and the supplementary video that our model produces visually plausible clothing wrinkles.

4.4 Predicting New Clothing

Putting everything together, we create a new instance of a garment by solving for the vertex coordinates of $\mathcal{Y}$ such that

$$\underset{\vec{y}_1,\ldots,\vec{y}_V}{\operatorname{argmin}} \sum_{t=1}^{T} \sum_{k=2,3} ||\hat{Q}_t(\vec{\psi}^f) R_{p(t)}(\vec{\theta}_c^f) \hat{D}_t(\vec{\beta}) \Delta\vec{x}_{t,k} - \Delta\vec{y}_{t,k}||^2.$$



(a) initial, after 1st iter, after 3rd iter (b)

Figure 9: Removing interpenetration. *(a) The left, middle, and right figures show the initial clothing prediction, the result after the first iteration of optimization, and the final result respectively. (b) Details of the interpenetration term. The blue dots and red dots represent body and clothing vertices respectively (see text).*

Computationally, the entire process described in this section involves several matrix multiplications and the solution of a sparse linear least squares problem.

5 Refining the Fit

Given a body shape and pose, DRAPE predicts a plausible clothing mesh. However, when the predicted clothing mesh is overlaid on the body (Figure 9(a)), there can be interpenetration between the clothing and the body. Consequently, the prediction step is followed by an efficient refinement step that warps the garment so that it lies entirely outside the body. This is achieved by minimizing a measure of cloth-body interpenetration with respect to the vertices of the garment, regularizing to make the cloth deform plausibly. Our iterative strategy alternates between computing the cloth vertices that penetrate the body, $\mathcal{P}$, and updating the clothing shape. The objective function comprises the following terms:

Cloth-body interpenetration. Given a penetrating vertex on the clothing in $\mathcal{P}$, we compute the nearest vertex on the body and its associated surface normal (Figure 9). We seek a clothing mesh such that all such vertices are pushed outside the body mesh. To that end, we define a penalty

$$p_{\mathcal{C}}(\mathcal{Y}) = \sum_{(i,j)\in\mathcal{C}\wedge i\in\mathcal{P}} ||\epsilon + \vec{n}_{\vec{b}_j}^T(\vec{y}_i - \vec{b}_j)||^2$$

where $\mathcal{C}$ is the set of correspondences between each clothing vertex, $\vec{y}_i$, and its closest body vertex, $\vec{b}_j$. Additionally $\vec{n}_{\vec{b}_j}$ is the normal for body vertex $\vec{b}_j$. Penetration happens when $\vec{n}_{\vec{b}_j}^T(\vec{y}_i - \vec{b}_j) < 0$, otherwise not. The term $\epsilon = -0.3cm$ ensures that clothing vertices lie sufficiently outside the body. This equation has many solutions. To make the cloth deform plausibly, we regularize the solution with two additional terms and one optional term:

Smooth warping. We prefer solutions where the warping of the cloth vertices varies smoothly over the surface of the garment; i.e. we seek to minimize

$$s(\mathcal{Y}) = \sum_{i\in\mathbf{V}} ||(\vec{y}_i - \tilde{\vec{y}}_i) - \frac{1}{|\mathbf{N}_i|}\sum_{j\in\mathbf{N}_i}(\vec{y}_j - \tilde{\vec{y}}_j)||^2$$

where $\mathbf{V}$ is the set of vertices in the garment, $\vec{y}$ are vertices of the warped garment, $\tilde{\vec{y}}$ are vertices of the garment before this iteration, and $\mathbf{N}_i$ is the set of vertices adjacent to vertex i. This term prefers a deformation of a vertex that is similar to the average deformation of its neighbors.

Damping. We prefer solutions where the warped vertices keep their original locations as much as possible; i.e. we seek to minimize

$$d(\mathcal{Y}) = \sum_{i \in \mathbf{V}} ||\vec{y}_i - \tilde{\vec{y}}_i||^2.$$

Tightness (optional). There are several clothing types such as shorts, skirts, and long pants that have a waistband that needs to be in contact with the body. The "tightness" term models this and here we use it only for lower-body clothing:

$$t_{\mathcal{C}}(\mathcal{Y}) = \sum_{(i,j) \in \mathcal{C} \wedge i \in \mathcal{T}} ||\vec{y}_i - \vec{b}_j||^2$$

where $\mathcal{T}$ is a set of vertices corresponding to the clothing waist band as defined by the pattern designer. This term specifies that every waist band vertex should be close to its nearest neighbor, $\vec{b}_j$, on the body. Note that this term could be used to model tight cuffs or any clothing region that fits snugly to the body.

Our goal is to efficiently compute the mesh that minimizes

$$E(\mathcal{Y}) = p_{\mathcal{C}}(\mathcal{Y}) + \lambda_s s(\mathcal{Y}) + \lambda_d d(\mathcal{Y}) + \lambda_t t_{\mathcal{C}}(\mathcal{Y}).$$

$E(\mathcal{Y})$ is a sum of squares of linear functions of the vertices, so we can find its solution efficiently using a linear least squares solver. However, because we only consider the "currently penetrating" vertices, $\mathcal{P}$, we need to solve the least squares problem iteratively so that we do not introduce new penetrating vertices that did not penetrate previously. At each iteration, we update $\mathcal{P}$, construct the sparse least squares problem, solve it, and update the clothing mesh. In our experiments we find that 3 iterations are sufficient to remove most collisions. The entire collision handling step is:

Given a body and a clothing mesh, compute corresponding vertices, $\mathcal{C}$, and only do this once.
$iter = 0$
repeat
  $iter = iter + 1$
  Determine the penetration vertex set $\mathcal{P}$.
  Construct a linear system and solve:
  $\underset{\vec{y}_1,\ldots,\vec{y}_V}{\text{argmin}}\{p_{\mathcal{C}}(\mathcal{Y}) + \lambda_s s(\mathcal{Y}) + \lambda_d d(\mathcal{Y}) + \lambda_t t_{\mathcal{C}}(\mathcal{Y})\}$
until $iter = 3$

The weights decrease with iterations: $\lambda_s = 4, 2, 1$ and $\lambda_d = 0.8, 0.6, 0.4$. For lower clothing with tight waist bands $\lambda_t = 0.2$.

Details. The clothing deformation model is translation invariant, so the three dimensional global translation of the garment must be determined. Note that the global rotation is already defined by the global rotation of the pelvis. During garment creation, we define several anchor points on the garment and the roughly corresponding points on the 3D body mesh. During fitting we compute the translation by minimizing the difference between the clothing and body anchor points.

To layer multiple pieces of clothing, we independently predict and position all pieces then refine from the inside out. For example, we predict the positions of pants and a T-shirt, refine the pants to be outside the body, and then refine the T-shirt to be outside the combined vertex set of the pants and body (here, the nearest neighbors $\mathcal{C}$ are computed between the upper-clothing and the combined vertex set). Combining the body and lower clothing is done efficiently by segmenting the body at the waist vertices and taking the union of the remaining upper body vertices and the lower clothing vertices.

6 Experimental Results

We evaluate the performance of the DRAPE model on different clothing types, body shapes, and motion sequences.

Qualitative Evaluation. To illustrate the behavior of the model, we synthesize clothing on 2 test sequences present in the pose training set and 10 novel test sequences not present in the training set. For each test motion sequence, we synthesize multiple bodies with different random shapes using the SCAPE body model. We then dress these bodies with different combinations of upper and lower clothing types. Here, we use 20 body shape coefficients ($\vec{\beta} \in \mathbb{R}^{20\times1}$), 5 clothing shape coefficients ($\vec{\phi} \in \mathbb{R}^{5\times1}$), and 50 pose-dependent clothing deformation coefficients ($\vec{\psi} \in \mathbb{R}^{50\times1}$). The choices of dimensions for $\vec{\beta}$ and $\vec{\phi}$ are discussed later. Figures 1 and 10 and the accompanying video illustrate that the method synthesizes clothing with detailed wrinkles and generalizes well to body shapes and poses not present in the training set.

We also visually compare the results of our method (with and without dynamics) to PBS. Figure 11 illustrates the results with two poses: 1) a male model rotating his torso, and 2) a female model in the middle of a jump. Figure 11 shows that the OptiTex simulations (a) contain more high frequency wrinkles than our method with dynamics (b). This is to be expected as our approach is an approximation to the physically-simulated clothing used for training. However, the strength of our method is being able to produce infinitely variable clothing sizes for different body shapes (Figure 12). Figure 11 (c) shows the results of our method without modeling dynamics; i. e., a zero order model that only uses $\vec{\theta}^f_c$ in Equation (5) to predict pose-dependent deformation. Comparing Figure 11 (b) and (c), we see that modeling dynamics is important for maintaining fine wrinkles, especially for fast motions.

Quantitative Evaluation. We take a male *T-shirt* as the representative clothing type for all quantitative experiments. The results are similar for other clothing types.

First, we verify the assumption that the pose-dependent non-rigid wrinkle deformations can be learned by linear regression. We expect the synthesized meshes produced by DRAPE to be smoother than the ground truth PBS meshes because the linear model is an approximation of the "real" wrinkle patterns. The effect of this smoothing is shown in Figure 11 (d) for a representative 176 frame test sequence (including running, jumping, and stopping) simulated using OptiTex (blue) and animated by DRAPE with dynamics (red) or DRAPE without dynamics (green). We compute the mean curvature at each vertex and then take the mean of this over all vertices in the garment; this provides an objective measure of the overall amount of wrinkles in the cloth. The plot shows that 1) we lose approximately $5 - 15\%$ of the high frequency wrinkles due to the linear regression approximation and 2) modeling dynamics greatly helps to maintain fine wrinkles.

Second, we explore the performance of clothing shape prediction, $\vec{\phi}$, as a function of the dimensionality of the SCAPE body coefficients $\vec{\beta}$ (refer to Equation (4)). We use the average Euclidean vertex distance to measure shape prediction error. We use leave-one-out cross validation to predict the i^{th} clothing instance using the PCA model and the linear shape predictor learned from all the remaining 59 instances excluding i. Figure 13 shows average shape prediction error over the 60 examples as a function of the dimensionality of the SCAPE body shape coefficients $\vec{\beta}$. If too many principal components are used, the model tends to over-fit the wrinkles and produce higher errors. The best generalization performance is achieved with approximately 20 PCA dimensions; this might increase with more shape training data. Thus, use use 20 body shape parameters, $\vec{\beta}$, in our experiments.

Speed and Memory. The run time performance for different garments and mesh resolutions is shown in Table 1. Our method is implemented using Matlab (single threaded) without special opti-

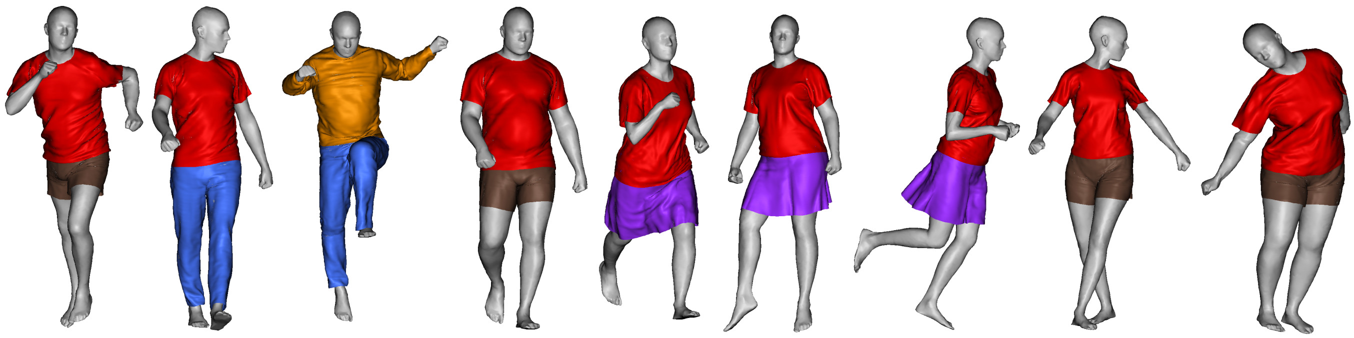
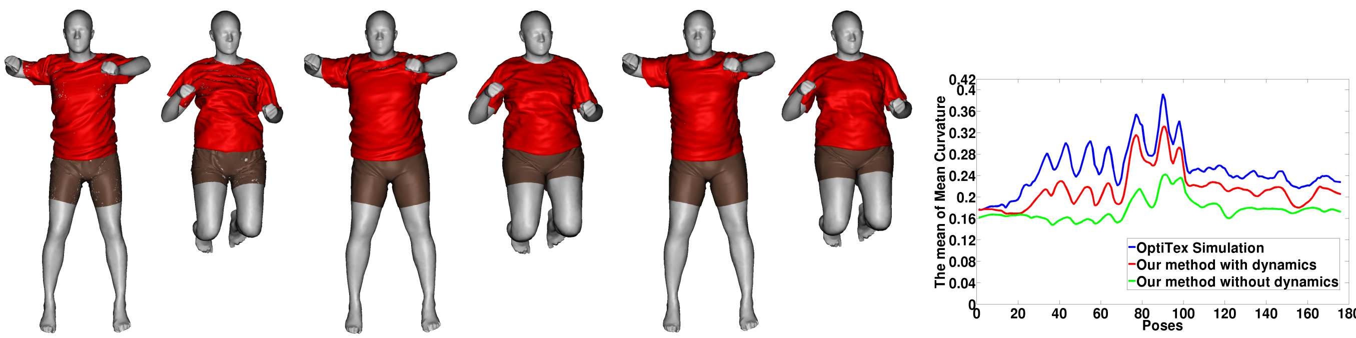


Figure 10: More DRAPE results (test sequences not present in training set). *We randomly combine upper clothing type, lower clothing type, pose, and body shape to generate synthetically clothed people. See accompanying* **video** *for more results.*



(a) OptiTex (b) DRAPE with dynamics (c) DRAPE without dynamics (d) Wrinkle comparison

Figure 11: Wrinkles. *Comparison between OptiTex simulation on mean bodies (a) and the DRAPE model with (b) and without (c) dynamics on novel bodies. (d) Measures how "wrinkled" the garment is in terms of the mean of the mean curvature. One test sequence with a motion not appearing in the training set is shown (176 frames). The DRAPE model (with dynamics) captures the wrinkles well while the model without dynamics over smooths the clothing.*

	Mesh Res		Run time (sec/frame) DRAPE			OptiTex	
Garment	#Vert	#Tri	Syn	Fit	Total	Single	Animation
T	18903	37446	0.1	0.8	0.9	46	12.1
Sh	10028	19686	0.06	0.4	0.46	20	5.3
Sk	8933	17582	0.06	0.4	0.46	35	7.2
LS	17136	34026	0.1	0.7	0.8	75	17.9
LP	15980	31746	0.09	0.6	0.69	62	15.7
T+Sh	28931	57132	0.16	1.2	1.4	122	28.0
LS+LP	33116	65772	0.19	1.3	1.6	308	37.6

Table 1: Run time performance. *Comparison of the run time performance of our method and the OptiTex package for various garments and resolutions. "T", "Sh", "Sk", "LS", "LP" stand for T-shirt, Shorts, Skirt, Long Sleeves, Long Pants respectively. "Syn" stands for clothing mesh synthesis while "Fit" represents the time for solving body-cloth interpenetration and preparation time. OptiTex-Single shows the run time for a single frame simulation and OptiTex-Animation shows the amortized run time per frame in an animation.*

mization such as GPU acceleration. The OptiTex run time does not include manually choosing the appropriate clothing size and placing the cloth pieces in appropriate initial positions.

For a single frame simulation, our method is much faster ($40-160$X) than the commercial physical simulation. If we run cloth simulation on a motion sequence, the amortized run time per-frame for OptiTex improves a lot, but is still around 15X slower than our method. This is because OptiTex makes use of temporal coherence. Our method fits clothing to each pose individually, therefore the per-frame run time for an animation is the same as for a single pose.

All timings were obtained with a 32 bit desktop machine with a 3.2 GHz AMD Phenom^{TM}II processor, 4.0 GB of memory, and an NVIDIA GeForce 8600 GT video card. Our method is not memory intensive. Consider a clothing mesh with 25000 triangles and a body model with 25000 triangles. Using floats for the vertices and normals, we need 450KB in total for the body and clothing to fit into memory. The shape PCA bases take 18MB (20 dimensions). The pose PCA bases take 27MB (30 dimensions). Representing the linear systems for computing the clothing deformation and clothing refinement takes approximately 400KB and 750KB respectively. This easily fits in the memory of a smart phone.

In addition to the run-time cost, there is an up-front cost of creating the training set for learning. The garment design process is completely standard and graded patterns like those used here exist for any mass produced garment already. Preparing the shape training set involves dressing each of the 60 training bodies once using the PBS system. The pose training set requires dressing the template body and simulating the motion sequences. Once the training data is created, learning the shape and pose-dependent models is very fast (minutes). Our advantage can be summarized as "simulate once, use often."

7 Limitations and Discussion

While DRAPE generates realistic clothing for different body shapes and poses, it has several limitations. First, the learned shape and

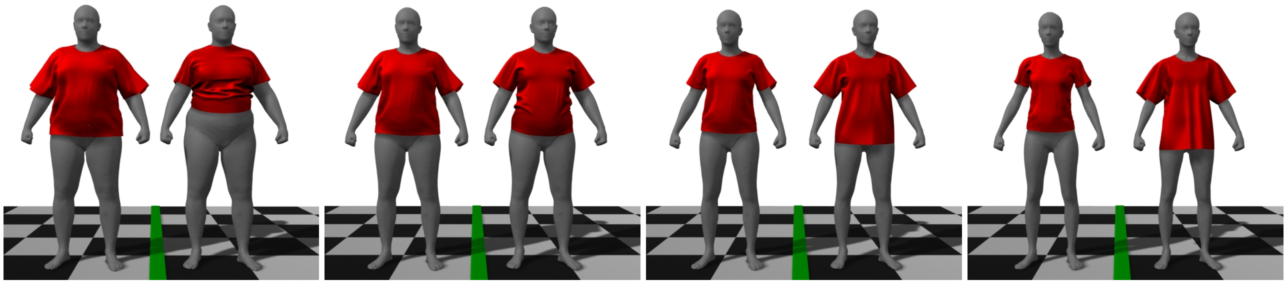
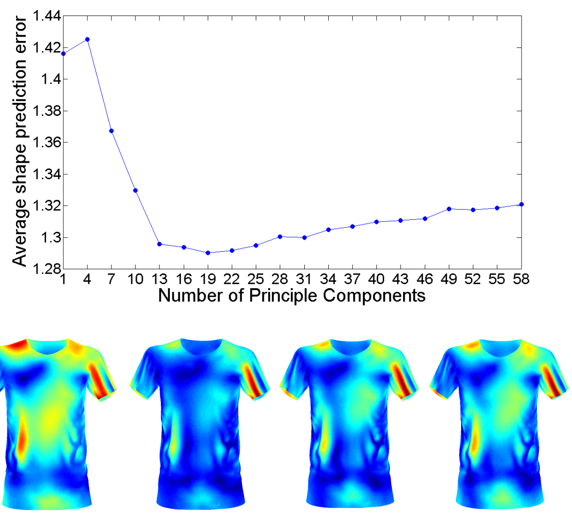


Figure 12: Importance of fit. *We compare bodies of different shapes clothed using DRAPE (left) and OptiTex simulation (right). The OptiTex simulation uses a fixed size T-shirt, emphasizing how the quality of the simulation depends heavily on choosing the right sized garment. In contrast, DRAPE automatically predict the appropriate, infinitely-sized, clothing for every body.*



(a) 10-Dim (b) **20-Dim** (c) 30-Dim (d) 40-Dim

Figure 13: Shape prediction accuracy versus subspace dimension. *The shape prediction error (in* cm*) does not decrease monotonically with the number of principal components. Over fitting occurs with more than 20 dimensions. These errors are illustrated on one of the ground truth clothing meshes, with hot/cold colors representing large/small errors.*

pose deformation models are independent and, when they are composed during synthesis, unnatural wrinkle patterns may be generated. Here we do not claim a physically realistic model of wrinkles, but rather demonstrate that often the simple factored model produces visually appealing results in practice. To minimize the occurrence of unnatural combinations, while retaining realism, we use a fairly smooth shape model and a higher frequency pose model (cf. [Wang et al. 2010]). The lower frequency shape model is naturally obtained by using fewer principal components for the clothing shape coefficients $\vec{\phi}$. The assumption is that low frequency wrinkles are related to body shape while high frequency wrinkles are largely determined by the body motion. While DRAPE handles interpenetration between the body and the clothing and between upper clothing and lower clothing, it does not model cloth self-penetration in the same clothing item.

It should be noted that the learned model is only as good as the input it is trained from. As shown here, the model is an approximation and DRAPE garments are smoother than the simulations. Here we used a particular commercial package for simulation but higher quality clothing simulations, or real cloth capture, would produce a more realistic DRAPE model. While there is some loss of fidelity compared with the training data, the advantages of the method are that the fitting is automatic, the model generalizes to different body shapes and it is computationally efficient. For many applications, particularly involving dressing many unknown body shapes, the trade off of automation for fidelity may be appropriate.

8 Conclusions and Future Work

DRAPE is a complete solution for dressing people in a variety of shapes, poses, and clothing types. DRAPE is learned from standard 2D clothing designs simulated on 3D avatars with varying shape and pose. Once learned, DRAPE adapts to different body shapes and poses without the redesign of clothing patterns; this effectively creates infinitely-sized clothing. A key contribution is that the method is automatic. In particular, animators do not need to place the cloth pieces in appropriate positions to dress an avatar.

Future work will explore a wider range of garment and fabric types. We will also learn models of "tucked in" clothing and more complex garments with pleats, cuffs, collars, and closures (buttons and zippers). Finally, clothing fit is not just about body shape but also involves individual preference. By training the model with different fit preferences (e.g. loose and tight) we should be able to add a "comfort" axis to the PCA shape basis that can be independently controlled.

Acknowledgements

This work was supported in part by the NIH-NINDS EUREKA program (R01–NS066311) and the Office of Naval Research (Contract W911QY-10-C-0172). We thank OptiTex for their assistance, Disney Research Pittsburgh for providing the training and testing motion sequences, and Caleb Weinreb for his Maya expertise.

References

ANGUELOV, D., SRINIVASAN, P., KOLLER, D., THRUN, S., RODGERS, J., AND DAVIS, J. 2005. SCAPE: Shape Completion and Animation of PEople. *ACM Trans. Graphics (Proc. SIGGRAPH) 24*, 3, 408–416.

BARAFF, D., AND WITKIN, A. 1998. Large steps in cloth simulation. *ACM Trans. Graphics (Proc. SIGGRAPH)*, 43–54.

BEAZLEY, A., AND BOND, T. 2003. *Computer-Aided Pattern Design and Product Development*. Wiley-Blackwell.

BORDES, J., MAHER, M., AND SECHREST, M. 2009. Nvidia apex: High definition physics with clothing and vegetation. In *Game Developers Conference*.

BRADLEY, D., POPA, T., SHEFFER, A., HEIDRICH, W., AND BOUBEKEUR, T. 2008. Markerless garment capture. *ACM Trans. Graphics (Proc. SIGGRAPH)*, 99:1–99:9.

BRIDSON, R., FEDKIW, R., AND ANDERSON, J. 2002. Robust treatment of collisions, contact and friction for cloth animation. *ACM Trans. Graphics (Proc. SIGGRAPH)*, 594–603.

BRIDSON, R., MARINO, S., AND FEDKIW, R. 2003. Simulation of clothing with folds and wrinkles. *Eurographics Symposium on Computer Animation*, 28–36.

CHOI, K.-J., AND KO, H.-S. 2002. Stable but responsive cloth. *ACM Trans. Graphics (Proc. SIGGRAPH) 21*, 604–611.

CHOI, K.-J., AND KO, H.-S. 2005. Research problems in clothing simulation. *Computer-Aided Design 37*, 6, 585–592.

CORDIER, F., AND MAGNENAT-THALMANN, N. 2002. Real-time animation of dressed virtual humans. *Computer Graphics Forum 21*, 3, 327–335.

CORDIER, F., AND MAGNENAT-THALMANN, N. 2005. A data-driven approach for real-time clothes simulation. *Computer Graphics Forum 24*, 173–183.

CORDIER, F., SEO, H., AND MAGNENAT-THALMANN, N. 2003. Made-to measure technologies for an online clothing store. *Computer Graphics and Applications*, 38–48.

CUTLER, L. D., GERSHBEIN, R., WANG, X. C., CURTIS, C., MAIGRETK, E., AND PRASSO, L. 2007. An art-directed wrinkle system for CG character clothing. *Graphical Models 69*, 5-6, 219–230.

DE AGUIAR, E., STOLL, C., THEOBALT, C., AHMED, N., SEIDEL, H.-P., AND THRUN, S. 2008. Performance capture from sparse multi-view video. *ACM Trans. Graphics (Proc. SIGGRAPH) 27*, 3, 1–10.

DE AGUIAR, E., SIGAL, L., TREUILLE, A., AND HODGINS, J. K. 2010. Stable spaces for real-time clothing. *ACM Trans. Graphics (Proc. SIGGRAPH) 29*, 4.

DECAUDIN, P., JULIUS, D., WITHER, J., BOISSIEUX, L., SHEFFER, A., AND CANI, M.-P. 2006. Virtual garments: A fully geometric approach for clothing design. *Computer Graphics Forum 25*, 3, 625–634.

FENG, W.-W., YU, Y., AND KIM, B. 2010. A deformation transformer for real-time cloth animation. *ACM Trans. Graphics (Proc. SIGGRAPH) 29*, 4.

GOLDENTHAL, R., HARMON, D., FATTAL, R., BERCOVIER, M., AND GRINSPUN, E. 2007. Efficient simulation of inextensible cloth. *ACM Trans. Graphics (Proc. SIGGRAPH) 26*, 3.

GUAN, P., FREIFELD, O., AND BLACK, M. J. 2010. A 2D human body model dressed in eigen clothing. In *European Conf. on Computer Vision*, 285–298.

HADAP, S., BANGERTER, E., VOLINO, P., AND MAGNENAT-THALMANN, N. 1999. Animating wrinkles on clothes. In *Visualization*, 175–182.

HOUSE, D., AND BREEN, D. 2000. *Cloth Modeling and Animation*. AK Peters, Ltd.

JAKOBSEN, T. 2001. Advanced character physics. In *Game Developers Conference*, 383–401.

JAMES, D. L., AND TWIGG, C. D. 2005. Skinning mesh animations. *ACM Trans. Graphics (Proc. SIGGRAPH) 24*, 3, 399–407.

KAVAN, L., SLOAN, P.-P., AND O'SULLIVAN, C. 2010. Fast and efficient skinning of animated meshes. *Computer Graphics Forum 29*, 2, 327–336.

KAVAN, L., GERSZEWSKI, D., BARGTEIL, A. W., AND SLOAN, P.-P. 2011. Physics-inspired upsampling for cloth simulation in games. *ACM Trans. Graphics (Proc. SIGGRAPH) 30*, 4.

KIM, T.-Y., AND VENDROVSKY, E. 2008. Drivenshape: A data-driven approach for shape deformation. In *Symposium on Computer Animation*, 49–55.

MÜLLER, M., AND CHENTANEZ, N. 2010. Wrinkle meshes. *Eurographics Symposium on Computer Animation*, 85–92.

NEALEN, A., MÜLLER, M., KEISER, R., BOXERMAN, E., AND CARLSON, M. 2006. Physically based deformable models in computer graphics. *Computer Graphics Forum 25*, 4, 809–836.

POPA, T., ZHOU, Q., BRADLEY, D., KRAEVOY, V., FU, H., SHEFFER, A., AND HEIDRICH, W. 2009. Wrinkling captured garments using space-time data-driven deformation. *Computer Graphics Forum 28*, 2, 427–435.

PRITCHARD, D., AND HEIDRICH, W. 2003. Cloth motion capture. *Computer Graphics Forum 22*, 3, 263–271.

ROBINETTE, K., BLACKWELL, S., DAANEN, H., BOEHMER, M., FLEMING, S., BRILL, T., HOEFERLIN, D., AND BURNSIDES, D. 2002. Civilian American and European Surface Anthropometry Resource (CAESAR) final report. Tech. Rep. AFRL-HE-WP-TR-2002-0169, US Air Force Research Laboratory.

ROHMER, D., POPA, T., CANI, M.-P., HAHMANN, S., AND SHEFFER, A. 2010. Animation wrinkling: Augmenting coarse cloth simulations with realistic-looking wrinkles. *ACM Trans. Graphics (Proc. SIGGRAPH Asia) 29*, 6.

SCHOLZ, V., STICH, T., KECKEISEN, M., WACKER, M., AND MAGNOR, M. 2005. Garment motion capture using color-coded patterns. *Computer Graphics Forum 24*, 3, 439–447.

STARK, J., AND HILTON, A. 2007. Surface capture for performance-based animation. *Computer Graphics and Applications 27*, 3, 21–31.

SUMNER, R. W., AND POPOVIĆ, J. 2004. Deformation transfer for triangle meshes. *ACM Trans. Graphics (Proc. SIGGRAPH) 23*, 2 (July), 399–405.

UMETANI, N., KAUFMAN, D. M., IGARASHI, T., AND GRINSPUN, E. 2011. Sensitive couture for interactive garment editing and modeling. *ACM Trans. Graphics (Proc. SIGGRAPH) 30*, 4.

WANG, H., HECHT, F., RAMAMOORTHI, R., AND O'BRIEN, J. F. 2010. Example-based wrinkle synthesis for clothing animation. *ACM Trans. Graphics (Proc. SIGGRAPH) 29*, 4 (July), 1–8.

WHITE, R., CRANE, K., AND FORSYTH, D. 2007. Capturing and animating occluded cloth. *ACM Trans. Graphics (Proc. SIGGRAPH) 26*, 3.