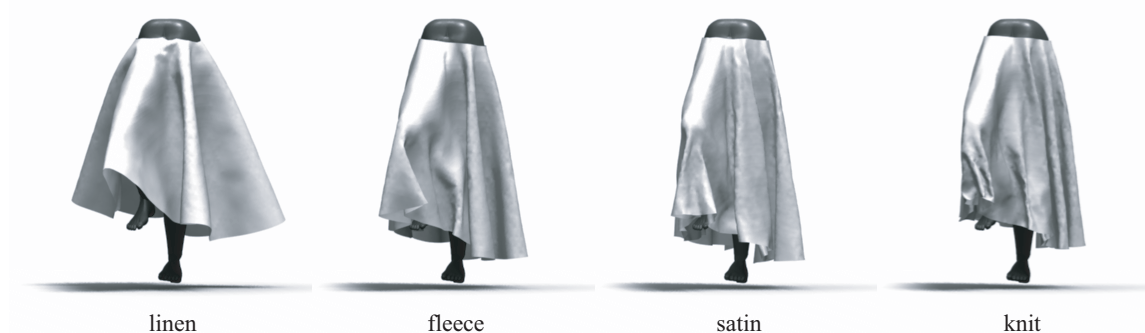


Estimating Cloth Simulation Parameters from Video

Kiran S. Bhat,¹ Christopher D. Twigg,^{1,2} Jessica K. Hodgins,¹ Pradeep K. Khosla,¹ Zoran Popović² and Steven M. Seitz,²

¹ School of Computer Science, Carnegie Mellon University

² Department of Computer Science and Engineering, University of Washington
<http://graphics.cs.cmu.edu/projects/skirts>



Abstract

Cloth simulations are notoriously difficult to tune due to the many parameters that must be adjusted to achieve the look of a particular fabric. In this paper, we present an algorithm for estimating the parameters of a cloth simulation from video data of real fabric. A perceptually motivated metric based on matching between folds is used to compare video of real cloth with simulation. This metric compares two video sequences of cloth and returns a number that measures the differences in their folds. Simulated annealing is used to minimize the frame by frame error between the metric for a given simulation and the real-world footage. To estimate all the cloth parameters, we identify simple static and dynamic calibration experiments that use small swatches of the fabric. To demonstrate the power of this approach, we use our algorithm to find the parameters for four different fabrics. We show the match between the video footage and simulated motion on the calibration experiments, on new video sequences for the swatches, and on a simulation of a full skirt.

1. Introduction

Several recent major movie releases have demonstrated that the motion of clothing adds greatly to the appearance of a virtual character. This effect is particularly compelling for scenes that include both real and synthetic actors such as those with Yoda and Anakin Skywalker in *Episode II: Attack of the Clones*. In such scenes, the virtual clothing must move and be rendered so that it blends in seamlessly with the motion and appearance of the real clothing in the scene. Realistic virtual clothing is possible now because of recent advances in cloth simulation techniques^{4, 9, 5, 37, 6}.

The motion of fabric is determined by resistance to bending, stretching, shearing, external forces, aerodynamic effects, friction, and collisions. Although with the right set of parameters good simulators produce very realistic looking motion, choosing parameters that will provide a particular appearance remains a time consuming task that requires the computation and viewing of many forward simulations. Some parameters can be chosen based on the animator's intuition about the fabric—a knit fabric is more stretchy than a woven fabric such as linen, for example. But not all the parameters of a cloth simulator are intuitive or map directly to

measurements that can be made by a system such as the Kawabata system²². In our paper, we address this problem by using optimization to automatically determine these parameters from a sequence of video frames of the fabrics under consideration.

The parameters are optimized on a set of static shots and motion clips of a small swatch of a particular fabric and then tested on a simulation of a full skirt made from that fabric. We designed the swatch tests to span the space of behaviors that we expect to see in the final sequences of motion with the skirt so that all parameters can be tuned appropriately. We use simulated annealing for the optimization step with an optimization function that assesses the extent to which the folds in the simulated and physical fabric match. This match is evaluated by means of a shape metric that uses projected light to detect surface orientation in real and simulated fabrics. The metric is tuned to be most sensitive along folds and to discount planar regions.

We use the system to find the parameters for four different fabrics. We show the match between the video footage and the simulated motion on the calibration experiments, on new video sequences for the swatches, and on a simulation of a full skirt as shown in the image on the previous page.

2. Related Work

Cloth modeling has a long history, dating back to work in the textile community from the mid-1930s by Peirce²⁷. Work on cloth modeling in computer graphics has focused on developing dynamic simulation techniques that are both realistic and fast. Baraff and Witkin describe a cloth model that uses stiff springs with implicit time integration⁴. This model was subsequently adapted to reduce the over-damping due to implicit integration⁹. Explicit time integration approaches¹⁸ use weaker springs for stretching and shearing, often explicitly limiting the amount of stretching^{29, 6}. Choi and Ko introduced a bending energy model that more accurately captures the fine creases and bends of cloth⁹. Lahey provides a comprehensive overview of cloth hysteresis models from the perspective of computational fabric mechanics²³. Extensive work has also been done on modeling collisions and friction. Cloth self-collision is handled either by untangling the cloth^{37, 39, 3} or by preemptively avoiding collisions^{30, 20, 6}. Various potential field methods have been used for general collision detection and response^{33, 32}.

Despite this large body of work on cloth simulation models, little work has appeared in the computer graphics literature on estimating the parameters of these models so that they match the behavior of real fabrics. Cloth parameter estimation has been studied in the textile community (for an overview, see Breen and colleagues¹⁷), but such methods have not yet enjoyed wide-spread use in the computer graphics community. An important exception is the work by Breen⁵ who used the Kawabata system²² to measure bending, shearing, and tensile parameters by subjecting a swatch

of fabric to a series of mechanical tests and measuring the force needed to deform it into a standard set of shapes. Although the Kawabata system can provide accurate measurements, these measurements are problematic for computer graphics cloth simulation problems for two reasons. First, there might not be a direct and simple mapping between the parameters for a particular cloth model and the Kawabata parameters. Second, the Kawabata system does not measure *dynamic* cloth parameters, e.g. air drag or damping, which are of key importance for moving cloth.

One promising approach for modeling cloth parameters is to automatically search for parameters that match real, observed cloth. Jovic and Huang fit parameters of a particle-based cloth model to fit a range scan of real cloth in a static rest configuration, draped over a sphere²¹. More challenging still, they attacked the problem of measuring the 3D geometry of an object from the resting shape of a piece of cloth draped over it, a problem that we do not consider in this paper. However, Jovic and Huang did not treat the problem of measuring dynamic parameters or demonstrate accurate results across a range of fabric types.

More distantly related are techniques for computing the geometry of cloth from images. Coarse estimates of the time-varying geometry of cloth can be computed using traditional stereo matching techniques by using two or more cameras and treating each time instant independently (see Scharstein and Szeliski³¹ for an overview). More accurate results may be obtained by projecting structured light patterns on the cloth (see Zhang et al.⁴⁰ for an overview). Rather than computing shape at every time instant independent from the next, it can be advantageous to integrate images over time to improve accuracy. Two examples of promising work along these lines are Carceroni and Kutulakos⁸ and Torresani et al.³⁴; both studies demonstrated reconstructions of moving cloth.

3. Cloth Model

Because our framework for estimating cloth simulation parameters is independent of the cloth model, we can, in principle, select a specific model that meets a set of criteria such as accuracy or simulation speed. Our choice of a cloth model was guided by two principles, realism and practicality. We wanted to use a model that was sophisticated enough to capture the detailed dynamic behavior found in real fabrics but still straightforward to implement. Because our intention was to apply the learned cloth model parameters to arbitrary garments with varying triangle resolution, it was also important that the cloth parameters correctly scale to varying resolutions of cloth.

We used the model described by Baraff and Witkin as the basis for our cloth simulator⁴. This model has sufficient richness to produce a wide variety of cloth behaviors. The underlying meshing is triangular, making clothing modelling

easier. More importantly, its input parameters are independent of meshing, so that parameters recovered on one mesh (the test swatch) can safely be transferred to another (the skirt). While nonlinear models such as the buckling behavior of Choi and Ko⁹ could potentially capture more realistic details of cloth, there is no straightforward way to scale the parameters of these models to meshes of varying resolutions. We expect that future application of our parameter-estimation framework to other scale-invariant cloth models will provide even more realistic results.

The model developed by Baraff and Witkin formulates the energy of a particular triangle in terms of so-called *condition functions* $\mathbf{C}(\mathbf{x})$ such that the total potential energy associated with the system is given by

$$\mathbf{E}_u = \frac{k_s}{2} \mathbf{C}(\mathbf{x}) \mathbf{C}^T(\mathbf{x}) \quad (1)$$

where k_s is a stiffness coefficient associated with the particular condition function. Forces are then simply calculated by

$$\mathbf{F} = \nabla_{\mathbf{x}} \mathbf{E}_u \quad (2)$$

Damping forces are similarly formulated in terms of the $\mathbf{C}(\mathbf{x})$,

$$\mathbf{d} = -k_d \frac{d\mathbf{C}}{d\mathbf{x}} \dot{\mathbf{C}}(\mathbf{x}) \quad (3)$$

We thus associate a stiffness coefficient k_s and a damping coefficient k_d with each of the $\mathbf{C}(\mathbf{x})$. In their paper, Baraff and Witkin describe a set of $\mathbf{C}(\mathbf{x})$ consisting of an in-plane stretch term, an in-plane shear term, and an out-of-plane bending term, giving a total of six parameters we can use to tune the internal cloth model. We refer the reader to their paper⁴ for the full details. We note, however, that (as they allude to in footnote 5) *energy* should scale linearly with triangle area to ensure scale independence. Therefore, we need to be careful when substituting $\mathbf{C}(\mathbf{x})$ for stretch and shear into eq. 1 that the resulting formula is linear in a rather than quadratic.

In the course of running our experiments, we discovered that a linear drag model such as that used in previous cloth work^{4,9} was not able to capture dynamic aspects of cloth. In order to add additional air-drag degrees of freedom to our cloth model without resorting to fully modeling aerodynamics²⁵, we developed a simple nonlinear alternative. To calculate the drag force on a triangle, we decompose the average velocity on the face into two components, one normal to the surface ($\mathbf{v}_N$) and one tangential ($\mathbf{v}_T$). Total drag force is then a linear function of tangential velocity and a quadratic function of normal velocity, with an additional term k_f that controls the degree of nonlinearity,

$$\mathbf{f}_{drag} = -a \left(\frac{k_N |\mathbf{v}_N|^2}{1 + k_f |\mathbf{v}_N|^2} \frac{\mathbf{v}_N}{|\mathbf{v}_N|} + k_T \mathbf{v}_T \right)$$

where a is the area of the given triangle. The linear term

is merely Stokes's law¹; the quadratic term matches better the experimental behavior of macroscopic bodies in low Reynold's number flow¹⁴. The addition of the $|\mathbf{v}_N|^2$ term in the denominator which makes the force asymptotic as $\mathbf{v}_N \rightarrow \infty$ was partially motivated by the observed phenomenon of *drag crisis*¹⁴, where under certain circumstances the drag can actually drop at the onset of turbulence¹. The optimizer is free to eliminate this behavior or other terms of this equation by setting the corresponding parameters to zero.

Initially, we used a first-order implicit Euler time integration scheme similar to the one described by Baraff and Witkin⁴. Unfortunately, we found that implicit integration introduced damping which could not be eliminated by optimizing cloth parameters. We had more success in matching realistic cloth motions by using higher-order explicit methods. The results in this paper all use an adaptive 4th-order accurate Runge-Kutta methods with embedded error estimation². While this method offers the advantages of familiarity and automatic bounding of error, it is rather slow, and recent work suggests that using 2nd-order backward differences⁹ or Newmark schemes⁷ may be a better choice.

For collision handling, we use a model similar to Bridson and colleagues⁶ which combines repulsion forces with impulses to robustly prevent all collisions before they occur. However, separating repulsion forces from the cloth internal dynamics and applying them outside the Runge-Kutta solver affected stability and resulted in visible artifacts. Instead, we apply repulsion forces inside the solver loop, so that the solver's own internal error estimation can remove these artifacts. The drawback of this technique is speed, because the system must check for collisions every time it evaluates the state derivatives (as opposed to once every *collision timestep* as in Bridson et al.⁶). To achieve acceptable performance, we used a number of collision culling algorithms, including hybrid top-down/bottom-up update²⁴, fast triangle reject tests²⁶, and a curvature-based criterion for rejecting self-collisions that was first introduced by Volino and Thalmann³⁸ and later refined by Provot³⁰.

4. A Metric for Matching Simulation to Video

We use a perceptually motivated metric to compare the motion of cloth in simulation with a video sequence of real fabric motion. Our algorithm compares the two sequences frame by frame and computes an average error across the entire sequence. Real fabrics exhibit a wide variety of motion ranging from soft and flowing (satin) to stiff (linen). Our metric captures the complex dynamics of cloth motion and also helps to distinguish between different fabrics.

Researchers in computational neurobiology hypothesize that the human perceptual system is sensitive to *moving edges* in video^{11, 12, 36}. Studies have shown that the receptive fields of simple cells in the macaque cortex act as edge or line detectors, responding to oriented edges or lines in natural scenes^{19, 35, 10}. In cloth, these edges correspond to folds,

which are regions of high variation in shape. Hence, our perceptually motivated metric for cloth compares two video sequences, one from simulation and one from the real world, and returns a number that measures the differences in their folds. The metric also penalizes the silhouette mismatch between the two sequences.

Fold Detection and Representation: Folds appear as soft edges in video whose appearance is dependent on material properties and lighting. Haddon and Forsyth^{15, 16} describe a learning approach for detecting and grouping folds (and grooves) in images of fabrics. Their technique can handle lighting effects caused by diffuse inter-reflections in cloth. However, most fabrics have very complicated reflectance properties. In our experiments, we normalize the effects of lighting and material reflectance by projecting a structured light pattern of horizontal stripes onto the fabric.

From the light-striped video sequence, we compute the *dominant orientation* for each edge pixel by convolving it with a steerable filter bank¹³. In our implementation, we use the G2/H2 quadrature pair with kernel size 12 as the basis filters. Details of computing the dominant orientation from the coefficients of filter bank response are given in Appendix I of Freeman and Adelson¹³. We convolve the image with the filter bank, compute the filter coefficient responses, blur the coefficients using a gaussian kernel, and compute the dominant orientation from these coefficients. We define the resulting orientation image as an *angle map*, shown in Fig. 1. The angle map, which measures the local orientation of the projected pattern, has a constant value when the surface is planar and varies at folds. We threshold the gradient of the angle map to get a *gradient mask* M_k for each frame of video (Fig. 1).

$$M_k(i, j) = \begin{cases} 1, & \|\delta(i, j)\| \geq \tau \\ 0, & \|\delta(i, j)\| < \tau \end{cases} \quad (4)$$

where τ is a user defined threshold and $\|\delta(i, j)\|$ is the magnitude of the gradient of the angle map at (i, j) . The gradient mask is non-zero at regions of high gradients, corresponding to folds, and zero at planar regions.

Fold Comparison: Our metric computes the frame by frame sum of squared differences (SSD) between masked angle maps in simulation with video. We preprocess the input video sequence to compute the angle map at each frame. Similarly, in simulation, we render the cloth shape using the current parameter values and project the same striped pattern, to get a striped simulation sequence. We compute the angle map at every frame in simulation from this sequence. We then compute the SSD of the angle values for all overlapping points in the two angle maps. We pre-multiply this difference with the gradient mask, which helps to emphasize the differences in fold regions over planar regions (Fig. 2). We sum the error across all frames to compute the overall error across the entire sequence. The error at any particular

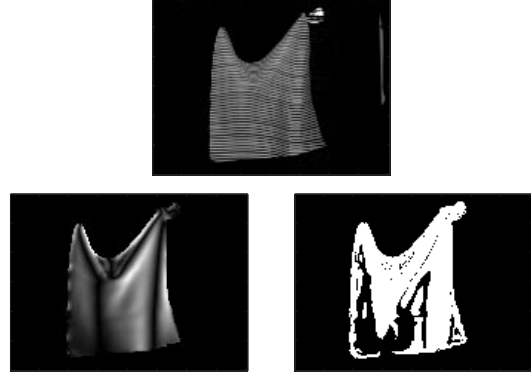


Figure 1: Top Row: Input light striped image. Bottom Row (left to right): angle map and gradient mask.

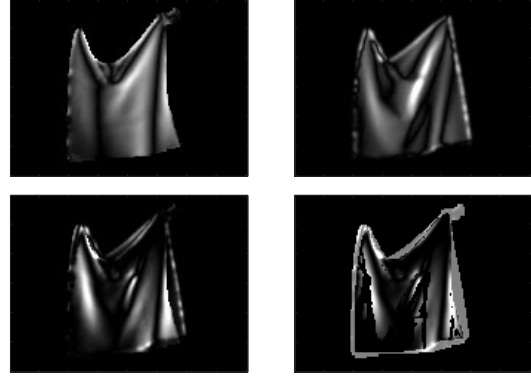


Figure 2: The stages in the metric pipeline. Top row (left to right): Angle map from video, angle map from simulation. Bottom row (left to right): angle map difference, final metric value for this frame (angle map difference multiplied by gradient mask from video).

frame k along the sequence is

$$E_k^{fold} = \sum_{i=0}^{S_x} \sum_{j=0}^{S_y} M_k(i, j) \cdot (\theta_k^{real}(i, j) - \theta_k^{sim}(i, j))^2 \quad (5)$$

where (S_x, S_y) is the size of the angle maps and θ^{real} , θ^{sim} are the angle values from real and simulation angle maps respectively.

Silhouette Comparison: In addition to the angle map error, we penalize the silhouette mismatch between the simulation and the video of real cloth. This penalty is proportional to the difference between the two silhouettes, i.e., the number of mismatched pixels.

$$E_k^{silh} = \sum_{i=0}^{S_x} \sum_{j=0}^{S_y} |A_k^{real}(i, j) - A_k^{sim}(i, j)| \quad (6)$$

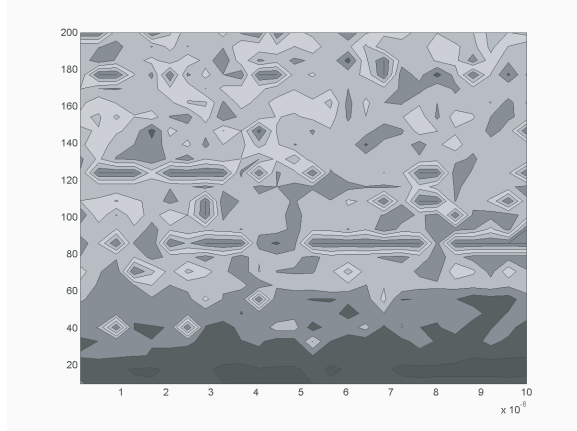


Figure 3: This plot shows angle map error as a function of bend and stretch stiffness parameters. Dark areas indicate regions of small error and bright areas correspond to large errors. Note that the space is fairly noisy. The minimum found by the optimizer is contained in the large dark region in the lower portion of the plot.

where

$$A_k(i, j) = \begin{cases} 1, & \text{inside silhouette} \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

The total error in frame k is

$$E_k = E_k^{fold} + \alpha E_k^{silh} \quad (8)$$

where α is a user-defined weight that controls the relative contribution of the two terms. We used a value of 0.1 for α in our experiments. The error across the entire sequence of length N frames is given by

$$E = \sum_{k=1}^N E_k \quad (9)$$

5. Parameter Identification

We use optimization to estimate the parameters of the cloth simulator from video. Before we describe the details of the optimizer, we look at the error space of the angle map metric, which gives us useful insight about the parameters of the system. Fig. 3 shows the variation of error for different values of bend stiffness and stretch stiffness coefficients for satin. To generate this error map, we compared the angle map from one frame in video with several angle maps in simulation. From the figure, it is evident that the error space is fairly noisy, with many local minima, motivating the need for a global optimization technique.

In addition to the parameter values, we estimate the relative importance of each parameter for a given experiment by performing a perturbation analysis at the solution point. The importance or *sensitivity* of a parameter p depends on

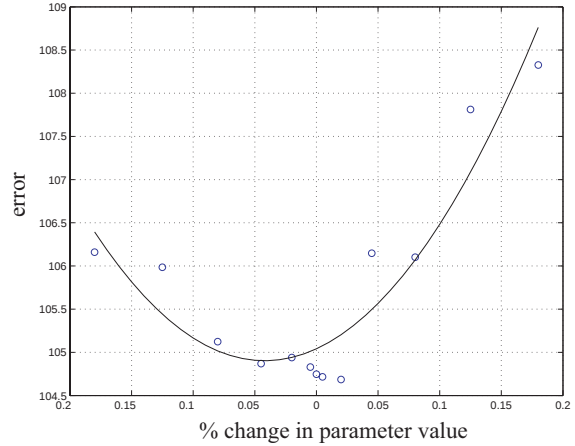


Figure 4: Perturbation analysis at the solution for bend stiffness parameter.

its local gradient $\frac{\partial E}{\partial p}$; it relates a small change in parameter value to a change in the error value. Instead of computing the gradient, we robustly compute the *variability* of the parameters, defined as $\frac{\partial p}{\partial E}$. To compute the variability, we perturb each parameter of the simulator individually up to $\pm 0.20\%$ of its value, compute the error and fit a quadratic to the data (Fig. 4). From the quadratic, the variability is computed as the change in parameter values that results in a 1% change in the error. Parameters with low variability have high sensitivity and are estimated reliably for a given experiment.

6. Optimization Framework

We use simulated annealing to find the parameters that minimize the error function given in eq. 9. Simulated annealing initially explores the space in a semi-random fashion and eventually takes downhill steps. The likelihood that it will take a step in a direction that is not locally optimal is a function of the *temperature* (Fig. 5). We chose to use the continuous simulated annealing method presented in Press et al.²⁸, which combines the Metropolis algorithm with the downhill simplex method for continuous n -variable optimization. We found it useful to reset the simplex with the current best solution when the temperature reduces by a factor of 3. Prior to optimization, we perform an exhaustive search for each fabric, where we choose four values for each cloth parameter across its entire range. This corresponds to a very coarse sampling of the parameter space. We simulate the fabric for all points in this coarse set and compute the error for each point by comparing against the real fabric. We initialize the optimizer with the point corresponding to the minimum error. We have found that this strategy allows the optimizer to locate a good minimum of the space.

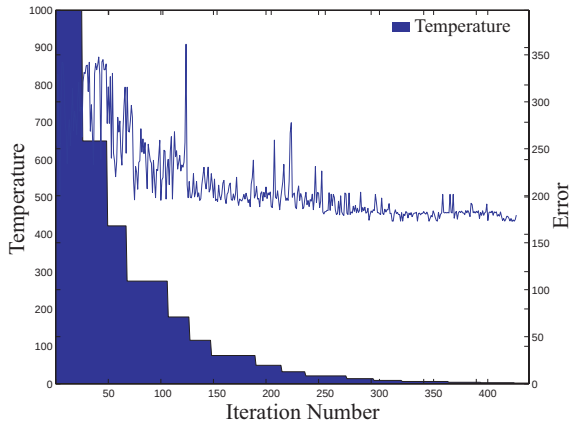


Figure 5: Progress of the simulated annealing optimizer as measured by error. The temperature decrease is governed by a geometric cooling schedule.

7. Experiments

We designed a few simple experiments to capture the dynamics of the different types of fabrics and the air/cloth interaction. The experiments are easy to perform, capture, and repeat; yet they demonstrate the complex dynamics of cloth motion. The parameters obtained from the simple experiments were used to simulate skirts and other complex fabric motions. In essence, our experiments were designed to be a *calibration* setup for estimating the static and dynamic parameters of a cloth simulator.

We perform two estimation experiments for each fabric, a *static test* and *waving test*. We used four types of fabrics: linen, fleece, satin and knit. These fabrics exhibit a wide range of static and dynamic behavior and span a large range of real fabrics.

We perform the static and waving tests on a small swatch of each fabric. In the static test, the two top corners of the fabric are held stationary, and the fabric is allowed to sag under gravity. For a fixed separation between the top corners, different fabrics attain different static shapes as shown in Fig. 6. The static test give a good estimate for the static stiffness and bend parameters. In the waving test, one of the top corners of the fabric is fixed and the other corner is moved back and forth (Fig. 7). The waving motion of fabrics in simulation is affected by their dynamic parameters. We see from the accompanying videos that real fabrics exhibit a wide range of interesting motions. Different fabrics also exhibit different types of motion for the same input excitation. We designed the waving motion to roughly match the types of motion occurring in real garments such as skirts. This gives reasonable estimates for cloth parameters while avoiding the need to optimize directly on complex fabric geometries (e.g. skirts) involving many collisions.

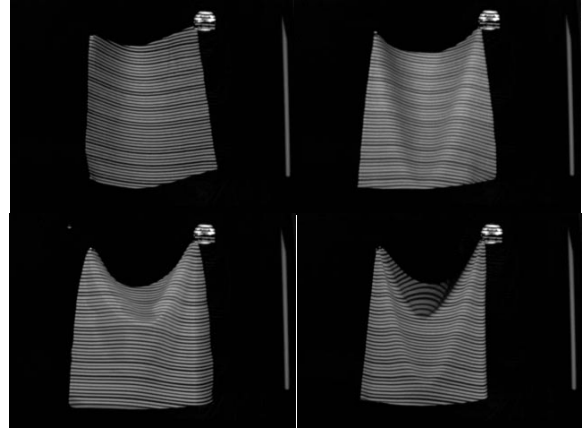


Figure 6: The static test with four real fabrics. Top row (left to right): linen and fleece. Bottom row: satin and knit. Top corner separation is identical across all four fabrics.

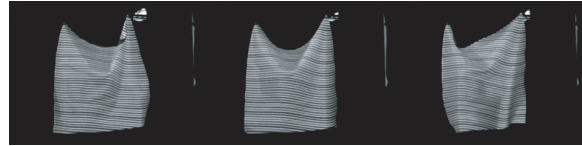


Figure 7: Three frames from the waving test for satin.

8. Results

In this section, we report the results of simulation parameters obtained using our technique applied to four fabrics: linen, fleece, satin and knit. We measured the mass and dimensions of the fabrics. We also accurately measure the position of the two top corners using a Vicon motion capture system. We compute the projection matrices for the camera and projector using a calibration grid comprising of several motion capture markers. We performed two trials per experiment, each with slightly different initial conditions and optimized on the first 50 frames of video in each trial. Each trial took approximately 50 hours to converge on a 2.8GHz Intel Xeon processor (approximately 600 iterations of simulated annealing). For this reason, we started the optimizations on the two trials (per fabric) with the same initial guess and chose parameters (optimized) that minimized the total error on the two trials.

Static test. We perform optimization on two trials for each fabric; the results are shown in Fig. 8 and Fig. 9. The two trials have different separation distances between the top corners. For each fabric, we optimize for six parameters: stiffness and damping parameters for stretch, shear, and bend. The air drag parameters were fixed for this experiment to the mid point of their range of values. The initial values for the two trials are obtained from a coarse exhaustive search (four

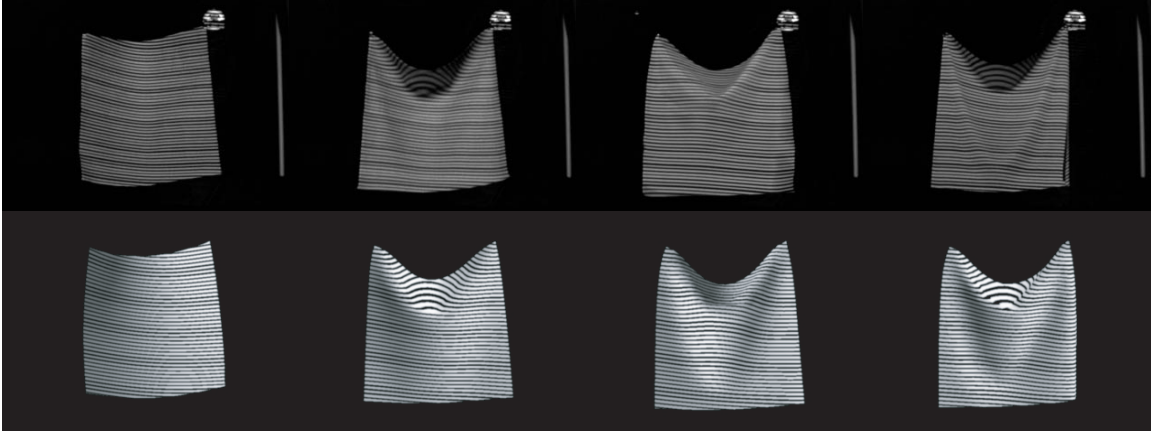


Figure 8: Results of optimization for the static test, trial 1. Top row: real fabrics (left to right) linen, fleece, satin and knit. Bottom row: Corresponding fabrics in simulation.

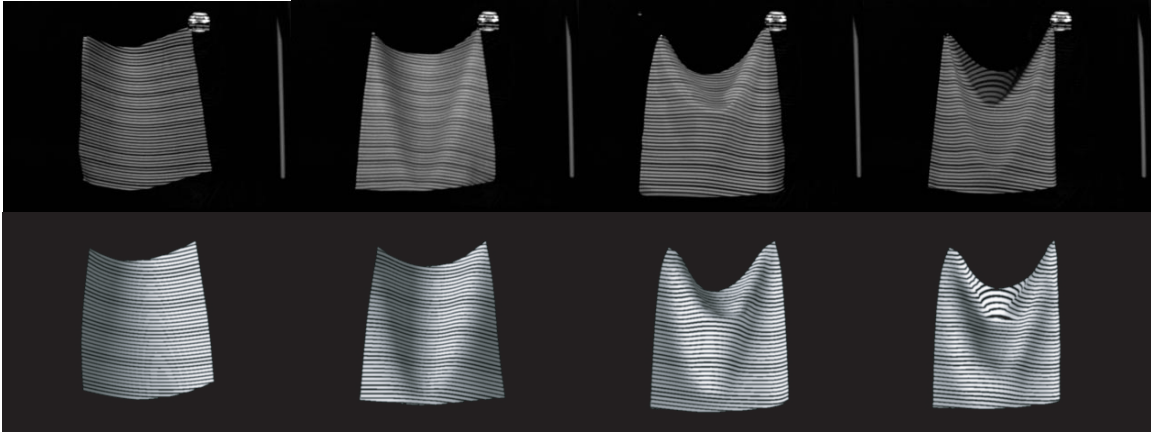


Figure 9: Results of optimization for the static test, trial 2. Top row: real fabrics. Bottom row: Corresponding fabrics in simulation.

	<i>Linen</i>			<i>Fleece</i>			<i>Satin</i>			<i>Knit</i>		
Pars	Start	Exp 1	Exp2	Start	Exp 1	Exp 2	Start	Exp 1	Exp 2	Start	Exp 1	Exp 2
1	1e-3	0.009	0.0045	1e-4	0.0001	0.0001	1e-5	1.106e-5	6.94e-6	1e-6	1.52e-6	1.51e-6
2	4000	404.9	3682.1	50	129.2	200.04	50	19.58	19.38	50	27.97	28.36
3	215.442	175.374	208.15	215.442	103.96	31.391	50	76.81	69.65	50	1226.44	2693.07
4	1e-7	9.92e-7	3.22e-7	2.15e-6	2.13e-7	4.11e-7	1e-7	2.49e-7	3.98e-7	1e-7	1.01e-7	2.27e-7
5	10	12.16	10.17	10	4.78	0.064	10	14.42	3.68	10	10.12	11.83
6	10	2.19	13.17	10	13.86	3.75	10	4.11	4.56	10	0.13	4.04

Table 1: Tabulation of the static parameters from two experiments. Legend: 1=bend, 2=stretch, 3=shear, 4=bend damping, 5=stretch damping, 6=shear damping.



Figure 10: Waving results for satin. The top picture in each block shows the real fabric and the bottom shows the corresponding frame from simulation.

	Bend	Stretch	Shear	Bend Damp	Stretch Damp	Shear Damp	Error
Initial Values	1.0e-05	50	50	2e-07	10	10	179.026
Optimization 1	6.93766e-06	19.3832	69.653	3.98337e-07	3.67932	4.56238	104.747
Optimization 2	7.77204e-06	20.2884	32.6492	2.08755e-07	1.95807	10.6535	104.502
Optimization 3	8.75613e-06	19.8365	50.8304	2.56854e-07	7.08276	9.25576	103.501
Optimization 4	9.55647e-06	19.2745	74.7429	3.14821e-07	5.47909	1.06559	103.243
Optimization 5	8.47762e-06	20.1119	36.762	2.3997e-07	8.38981	11.9167	103.849
Variability (in %)	9.18	8.10	23.01	21.11	>100	>100	

Table 2: Performance of simulated annealing on several optimizations. All the optimizations start with values which are within $\pm 5\%$ of the initial values given in the first row. Parameters with high variability (e.g., stretch damping) are estimated poorly and vary significantly across the different optimizations. However, parameters with low variability (e.g., bend) are consistent across multiple optimizations.

	Linen			Fleece			Satin			Knit		
Pars	Start	Exp 1	Exp2	Start	Exp 1	Exp 2	Start	Exp 1	Exp 2	Start	Exp 1	Exp 2
1	1e-3	0.001	0.0008	1e-4	1.13e-5	0.0001	1e-5	6.41e-6	5.64e-6	1e-6	1.12e-6	1.16e-6
2	4000	2016.8	2935.26	50	82.61	89.32	50	26.42	32.37	50	69.75	12.68
3	215.442	167.833	465.73	215.443	255.198	296.861	50	97.77	74.24	50	37.48	59.99
4	1e-7	3.17e-7	4.76e-7	2.15e-6	1.36e-6	1.31e-6	1e-7	1.48e-6	1.24e-7	1e-7	1.04e-7	5.4e-7
5	10	2.71	5.17	10	2.39	5.92	10	0.57	4.48	10	4.52	3.87
6	10	3.89	5.52	10	1.59	9.82	10	6.57	4.73	10	4.93	2.64
7	2	8.73	2.18	2	2.40	1.62	2	4.85	0.85	2	1.54	0.99
8	2	5.56	1.99	2	3.15	0.31	2	1.76	1.48	2	0.52	1.79
9	2	0.44	1.29	2	4.28	1.23	2	0.95	0.79	2	1.22	0.33

Table 3: Waving parameters from two experiments. Parameters from the experiment shown in bold is selected as the final estimate from this experiment. Legend: 1=bend, 2=stretch, 3=shear, 4=bend damping, 5=stretch damping, 6=shear damping, 7=linear drag, 8=quadratic drag, 9=drag degradation.

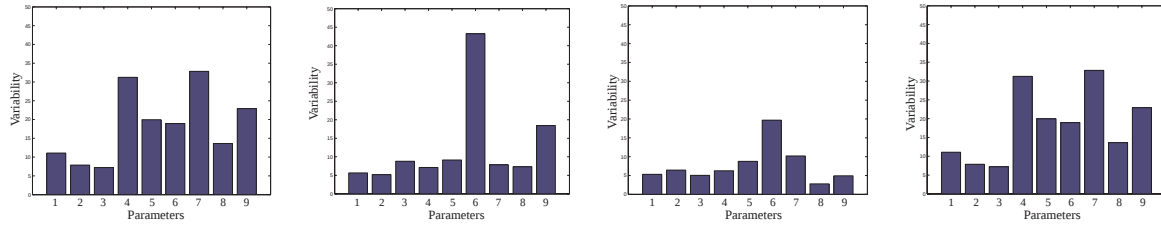


Figure 11: Bar charts showing the variability analysis results for the waving test. From left to right: linen, fleece, satin and knit. Legend: 1=bend, 2=stretch, 3=shear, 4=bend damping, 5=stretch damping, 6=shear damping, 7=linear drag, 8=quadratic drag, 9=drag degradation.

values per parameter). The initial values and final values of the estimated parameters are summarized in Table 1. Figs. 8 and 9 show a very good visual match between the simulations with their counterpart real fabrics. However, there is a significant disparity in the final optimized values from the two trials. In order to understand this disparity, we performed a set of optimizations (on a single fabric) with very similar initial values. Table 2 shows the parameter values for satin from five optimizations where the initial conditions were randomly varied by $\pm 5\%$. From the table, we see that the final error values are very close. We get consistent estimates for parameters that have lower variability (e.g., bend, stretch). Parameters with high variability are estimated poorly, because their values do not contribute sufficiently to the error. This result is consistent with our intuition that static tests cannot be used to estimate dynamic parameters like stretch and shear damping or air drag and motivates the waving test, which excites both the static and waving parameters.

Waving test. We optimize for nine parameters in the waving test: the six cloth stiffness and damping parameters and three air drag parameters (Fig. 10). As with the static test, we initialize the static parameters in this test from a coarse exhaustive search. The dynamic parameters were initialized using a random guess. We optimized on the first 50 frames of the sequence. The initial values and final values of the optimized parameters for two trials are reported in Table 3. The final values of the parameters from the two trials differ in part because the variability of the parameters is still fairly high (Fig. 11). Different motions or larger sequence might further reduce the variability of the parameters. We choose the parameter set that minimizes the sum of the error from the two trials. For instance, in the following example of fleece waving, we choose the parameters from experiment 2.

	Error: Exp 1	Error: Exp 2	Total Error
Pars: Exp 1	4257.2	10913.5	15170.7
Pars: Exp 2	4566.2	7144.3	11710.5

This approach seems to produce plausible results with skirts and other validation experiments. However, we believe that a more general solution for parameter identification using our framework is to simultaneously optimize across multiple trials of different experiments.

Optimization progress. Fig. 12 shows the static shape of the simulation before and after optimization. Fig. 13 shows the corresponding angle map comparison. These two figures show the progress of the optimization and indicate that the minimum corresponds to a visually compelling match.

Metric validation. We compare each of the four optimized angle maps from simulation (corresponding to the four fabrics) with the four angle maps computed from video. In Fig. 14, each curve shows one fabric (e.g., fleece) compared with four simulations, corresponding to each fabric type. We

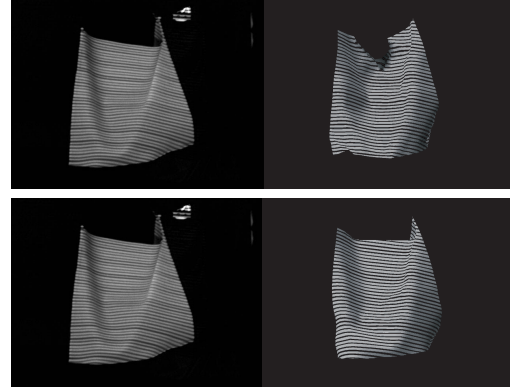


Figure 12: Showing the improvement in shape match after optimization. The top row compares a video frame of fleece with simulation before optimization. The bottom row shows the corresponding video/simulation pair after optimization.

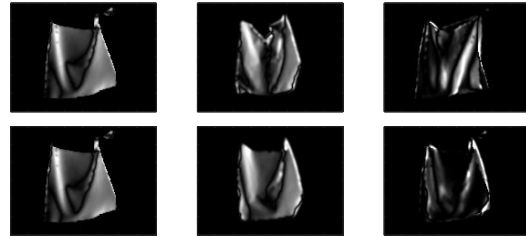


Figure 13: Comparison of angle maps for the shapes shown in Fig. 12 before and after optimization. Top Row (Before Optimization, from left to right): Angle map from video, angle map from simulation, angle map SSD. Bottom Row: The corresponding anglemaps after optimization.

see that each fabric in simulation has a minimum error when compared to its counterpart in reality. Fig. 14 also demonstrates that our approach could be potentially useful for recognizing different types of fabrics in video.

Generalization. We evaluated the parameters obtained from optimization on longer sequences (150 frames). Fig. 10 and the accompanying videos show a good visual match between corresponding frames in simulation and video. All videos are available off our web page and/or included in the DVD. The videos also show that the parameters obtained from optimization generalize well on new sequences. We also validated the estimated parameters on a long sequence actuated by a robot (Fig. 15). We used a Mitsubishi PA-10 robot arm to move the corner point along a simple sinusoidal trajectory, thereby ensuring that we had the same input motion across different fabrics. Finally, we used the optimized parameters to simulate a skipping motion of a human actor wearing a skirt (Fig. 16). Here, the actor repeats the same skipping motion (approximately) for the four different skirts.

We used data from a full body optical motion capture of the actor performing the same skipping motion (in another trial) to drive the character for the cloth simulation. The results show that the parameters obtained from our optimization approach approximately capture the static shape and dynamic properties of skirts of different materials.

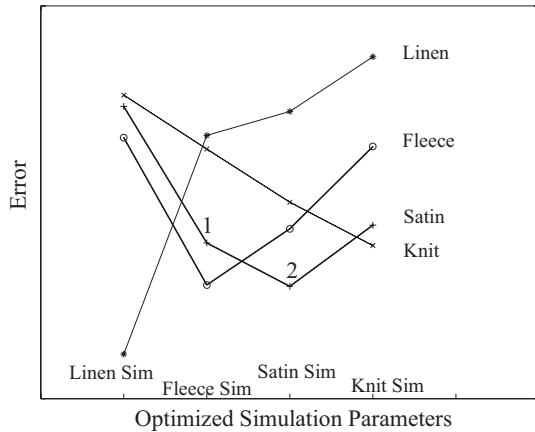


Figure 14: Comparing the optimized parameters in simulation for each fabric with the four real fabrics. For example, point 1 in the graph shows the error when a simulation with fleece parameters is compared with video of satin. Similarly, point 2 is the error when the satin simulation is compared with real satin. The four curves have a minimum when they are compared to their correct counterparts.

9. Discussion

This paper describes an optimization framework for identifying the simulation parameters of cloth from video. We captured the behavior of small swatches of fabric using a set of dynamic and static tests and demonstrated that the optimizer could identify appropriate simulation parameters from those tests. These parameters produced four distinct and recognizable fabrics when applied to a more complex simulation of a skirt as it was driven by motion capture data from a human figure.

The cloth model was not the main focus of this research, yet in early versions of the system it was often the bottleneck in achieving appealing results. To match a video sequence accurately, the cloth physics model as well as the collision algorithms must be chosen carefully. Instabilities in the collision handling will cause perceptible quivering in the motion of cloth. Similarly, extra damping introduced by the integration method makes crisp folds impossible to match. The parameters must also be independent of the resolution of the mesh so that they can be identified on low resolution swatches and applied to higher resolution garments. Progress is being made in these areas, however, and cloth

models are continually improving. For example, Bridson et al.⁷ introduces a scale-independent bend model with encouraging results.

Our cloth model does not diverge significantly from previous models discussed in the literature. Our only major addition was a simple nonlinearity we introduced into the drag model. Hence, our approach should generalize to any parametrized cloth model that produces a sufficiently rich set of physically realistic motions.

Although the skirt is far more complex than the swatches that were used to determine the parameters, it is not as complex as many garments, for example, a form-fitting pair of pants or a tailored blazer. For more complex garments, choosing the parameters via optimization on small, flat swatches may not be sufficient because the shape of the garment is determined by darts, pleats and by the interplay of different fabrics (wool, lining, and interfacing, for example). More complex garments may require the hand design of additional tests that mimic particular behaviors or elements of the garment in isolation. Moreover, the model might need extra parameters to handle anisotropic effects, hysteresis and coupling effects (stretching along one direction causing shrinking in the other direction), all of which would need specialized tests.

En route to the metric used in the experiments described here, we tried a number of other metrics: comparing the overlap of the silhouettes, the distance function between silhouette edges, and using information from internal edges marked on the fabric. The metric that measures folds and silhouettes, in concert with the projector for the light stripes, proved to be a simple and effective metric that far outperformed our earlier attempts. The space of possible metrics is vast, of course, but one class of metrics that we did not experiment with are statistical metrics that compute a function of the shape of the fabric across time rather than evaluating the match on a frame-by-frame basis. The experiments with the swatches were carefully controlled to have initial conditions for the simulation that matched those seen in the video. If instead, we were to optimize on more complicated garments, then such tight control of the initial conditions is unlikely and a statistical metric might be preferable. Such a metric might, for example, compute the average number of folds across a time sequence rather than looking for a fold to appear at a particular location on the swatch.

Our hope is that this work will promote a more rigorous evaluation of various cloth models, especially with respect to how accurately they match reality, and perhaps lead to creation of a standardized set of benchmarks for cloth simulation models.

10. Acknowledgements

We like to thank Jia-Chi Wu for his help in implementing the cloth simulator and the simplified air drag model. We

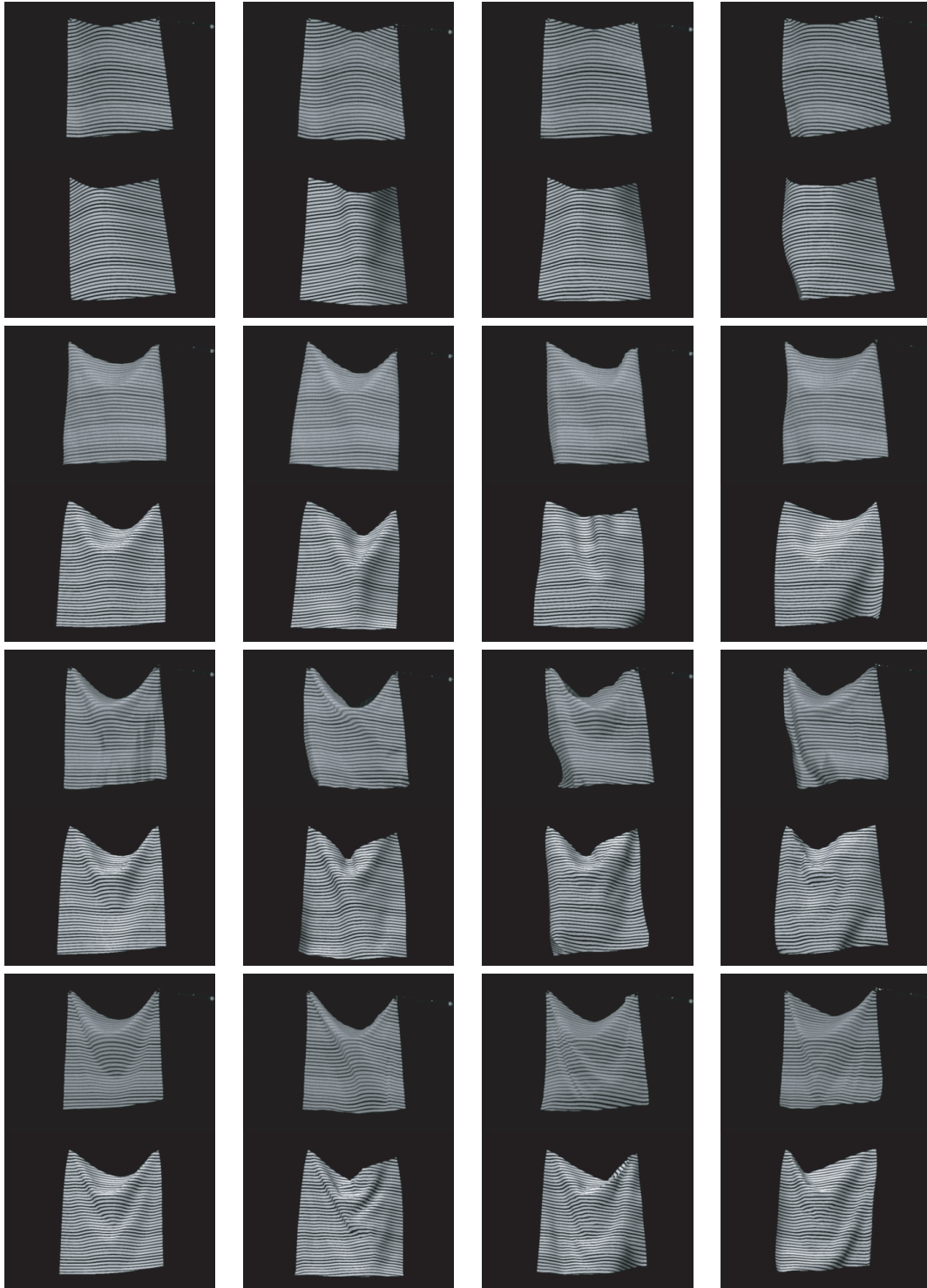


Figure 15: Validating the estimated parameters using the same input excitation. The top right corner of the fabric is actuated using a Mitsubishi PA-10 robot. Each row shows the match between video (top) and simulation (bottom) at four frames chosen from a 100 frame sequence. The fabrics, from top to bottom, are linen, fleece, satin and knit respectively.

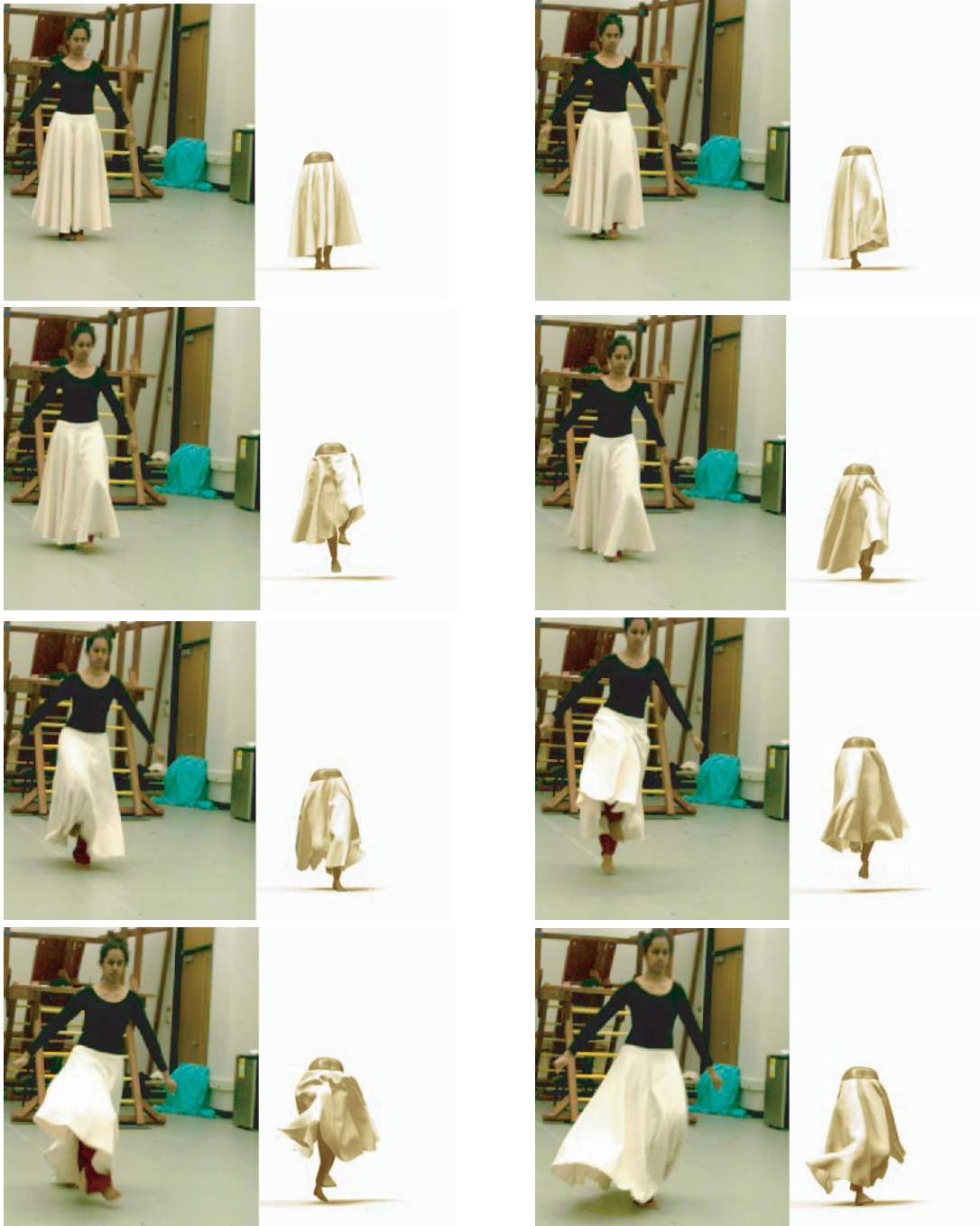


Figure 16: Validating the estimated parameters on a more complicated motion and garment. We show (from left to right, top to bottom) several frames of an actor skipping while wearing a fleece skirt. The corresponding frames of the skirt in simulation shows that our technique captures the approximate shape and dynamics of the real skirt. These frames were equally spaced across the entire sequence (0.5 seconds apart). The videos in the webpage show the validation results on all four skirts.

also thank Roshni Sivasankaran, Bonnie Jang and Priyanka Vaddi for their help with the skirt motion capture experiments, and Mike Stevens for cleaning up the motion capture data. The support of NSF under ITR grant IIS-0113007 and EIA-0196217 is gratefully acknowledged. Finally, we would like to thank the anonymous reviewers for their valuable comments and feedback.

References

1. D. J. Acheson. *Elementary Fluid Dynamics*. Oxford University Press, Oxford, 1990.
2. U. M. Ascher and L. R. Petzold. *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. Society for Industrial and Applied Mathematics, Philadelphia, 1998.
3. D. Baraff, A. Witkin, and M. Kass. Untangling cloth. *ACM Transactions on Graphics*, 22(3), July 2003.
4. D. Baraff and A. P. Witkin. Large steps in cloth simulation. In *Proceedings of SIGGRAPH 98*, Computer Graphics Proceedings, Annual Conference Series, pages 43–54, July 1998.
5. D. E. Breen, D. H. House, and M. J. Wozny. Predicting the drape of woven cloth using interacting particles. In *Proceedings of SIGGRAPH 94*, Computer Graphics Proceedings, Annual Conference Series, pages 365–372, July 1994.
6. R. Bridson, R. P. Fedkiw, and J. Anderson. Robust treatment of collisions, contact, and friction for cloth animation. *ACM Transactions on Graphics*, 21(3):594–603, July 2002.
7. R. Bridson, S. Marino, and R. Fedkiw. Simulation of clothing with folds and wrinkles. In *ACM SIGGRAPH Symposium on Computer Animation*, July 2003.
8. R. L. Carceroni and K. N. Kutulakos. Multi-view scene capture by surfel sampling: From video streams to non-rigid 3d motion, shape & reflectance. *Proc. International conference on Computer Vision*, pages 60–67, 2001.
9. K.-J. Choi and H.-S. Ko. Stable but responsive cloth. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 604–611. ACM Press, 2002.
10. G. C. DeAngelis, I. Ohzawa, and R. D. Freeman. Spatiotemporal organization of simple-cell receptive fields in the cat's striate cortex. i. general characteristics and postnatal development. *J Neurophysiol.*, 69(4):1091–1117, 1993.
11. D. J. Field. Relations between the statistics of natural images and the response properties of cortical cells. *Journal of the Optical Society of America A*, 4(12):2379–2394, December 1987.
12. D. J. Field. What is the goal of sensory coding? *Neural Computation*, 6(4):559–601, July 1994.
13. W. T. Freeman and E. H. Adelson. The design and use of steerable filters. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13(9):891–906, 1991.
14. C. Frohlich. Aerodynamic drag crisis and its possible effect on the flight of baseballs. *American Journal of Physics*, 52(4):325–334, April 1984.
15. J. Haddon, D. Forsyth, and D. Parks. The appearance of clothing. <http://http.cs.berkeley.edu/haddon/clothingshade.ps>, 1998.
16. J. Haddon and D. A. Forsyth. Shading primitives: finding folds and shallow grooves. *Proc. International conference on Computer Vision*, pages 236–41, 1998.
17. D. H. House and D. E. Breen, editors. *Cloth Modeling and Animation*. A.K. Peters, Ltd., Natick, Massachusetts, July 2000.
18. D. H. House, R. W. DeVaul, and D. E. Breen. Towards simulating cloth dynamics using interacting particles. *International Journal of Clothing Science and Technology*, 8(3):75–94, 1996.
19. D. H. Hubel and T. N. Wiesel. Receptive fields and functional architecture of monkey striate cortex. *J. Physiol. (Lond.)*, 195(1):215–243, 1968.
20. S. Huh, D. Metaxas, and N. Badler. Collision resolutions in cloth simulation. In *Computer Animation*, pages 122–127. IEEE, 2001.
21. N. Jovic and T. S. Huang. Estimating cloth draping parameters from range data. In *International Workshop on Synthetic-Natural Hybrid Coding and 3-D Imaging*, pages 73–76, Rhodes, Greece, 1997.
22. S. Kawabata. The standardization and analysis of hand evaluation. *The Textile Machinery Society of Japan*, 1980.
23. T. J. Lahey. Modeling hysteresis in the bending of fabrics. Master's thesis, University of Waterloo, 2002.
24. T. Larsson and T. Akenine-Möller. Collision detection for continuously deforming bodies. In *Eurographics 2001*, pages 325–333, 2001.
25. L. Ling. Aerodynamic effects. In D. H. House and D. E. Breen, editors, *Cloth Modeling and Animation*. A.K. Peters, Ltd., Natick, Massachusetts, July 2000.
26. T. Möller. A fast triangle-triangle intersection test. *Journal of Graphics Tools*, 2(2):25–30, 1997.
27. F. T. Peirce. The geometry of cloth structure. *Journal of the Textile Institute*, 28(T45–T97), 1937.
28. W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes: The Art of Scientific*

- Computing*, chapter 10.9, pages 444–455. Cambridge University Press, Cambridge (UK) and New York, 2nd edition, 1992.
29. X. Provot. Deformation constraints in a mass-spring model to describe rigid cloth behavior. In *Graphics Interface '95*, pages 147–154, May 1995.
30. X. Provot. Collision and self-collision handling in cloth model dedicated to design. In *Computer Animation and Simulation '97*, pages 177–190, September 1997.
31. D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal on Computer Vision*, 47(1):7–42, 2002.
32. S. Sclaroff and A. Pentland. Generalized implicit functions for computer graphics. In *Computer Graphics (Proceedings of SIGGRAPH 91)*, volume 25, pages 247–250, July 1991.
33. D. Terzopoulos, J. Platt, A. Barr, and K. Fleischer. Elastically deformable models. In *Computer Graphics (Proceedings of SIGGRAPH 87)*, volume 21, pages 205–214, July 1987.
34. L. Torresani, D. Yang, G. Alexander, and C. Bregler. Tracking and modelling non-rigid objects with rank constraints. *Proc. Computer Vision and Pattern Recognition Conference*, pages 493–500, 2001.
35. R. L. D. Valois, E. W. Yund, and N. Hepler. The orientation and direction selectivity of cells in macaque visual cortex. *Vision Research*, 22(5):531–544, 1982.
36. J. H. van Hateren and D. L. Ruderman. Independent component analysis of natural image sequences yields spatio-temporal filters similar to simple cells in primary visual cortex. *Proc. Royal Soc. Lond. B*, 265(1412):2315–2320, 1998.
37. P. Volino, M. Courchesne, and N. Magnenat-Thalmann. Versatile and efficient techniques for simulating cloth and other deformable objects. *Computer Graphics*, 29(Annual Conference Series):137–144, 1995.
38. P. Volino and N. Magnenat-Thalmann. Efficient self-collision detection on smoothly discretised surface animations using geometrical shape regularity. In *Computer Graphics Forum (Eurographics Proc.)*, volume 13, pages 155–166, 1994.
39. P. Volino and N. Magnenat-Thalmann. Accurate collision response on polygonal meshes. In *Computer Animation 2000*, pages 154–163, May 2000.
40. L. Zhang, B. Curless, and S. M. Seitz. Rapid shape acquisition using color structured light and multi-pass dynamic programming. In *Proc. Symposium on 3D Data Processing Visualization and Transmission (3DPVT)*, 2002.