Learning-based Cloth Material Recovery from Video

Shan Yang, Junbang Liang, Ming C. Lin
Department of Computer Science
University of North Carolina at Chapel Hill
{alexyang,liangjb,lin}@cs.unc.edu

Abstract

Image and video understanding enables better reconstruction of the physical world. Existing methods focus largely on geometry and visual appearance of the reconstructed scene. In this paper, we extend the frontier in image understanding and present a method to recover the material properties of cloth from a video. Previous cloth material recovery methods often require markers or complex experimental set-up to acquire physical properties, or are limited to certain types of images or videos. Our approach takes advantages of the appearance changes of the moving cloth to infer its physical properties. To extract information about the cloth, our method characterizes both the motion and the visual appearance of the cloth geometry. We apply the Convolutional Neural Network (CNN) and the Long Short Term Memory (LSTM) neural network to material recovery of cloth from videos. We also exploit simulated data to help statistical learning of mapping between the visual appearance and material type of the cloth. The effectiveness of our method is demonstrated via validation using both the simulated datasets and the real-life recorded videos.

1. Introduction

Recent advances in virtual reality (VR) make it possible to recreate a vivid virtual world that can be captured as a collection of images or a video sequence. Better understanding of the physical scene can further assist in the virtual reconstruction of the real world by incorporating more realistic motion and physical interaction of virtual objects. With the introduction of the deep neural network and advances in image understanding, object detection and recognition have achieved an unprecedented level of accuracy. Capturing the physical properties of the objects in the environment can further provide a more realistic human-scene interaction. For example, in a virtual try-on system for clothing, it is critical to use material properties that correctly reflect the garment behavior; physical recreation of the fabric not only gives a compelling visual simulacrum of the cloth, but also affects how the garment feels and fits on the body. In this paper, we propose a novel method of extracting physical information from videos in a way analogous to how humans perceive physical systems in an image or a video using "mental simulations" [11].

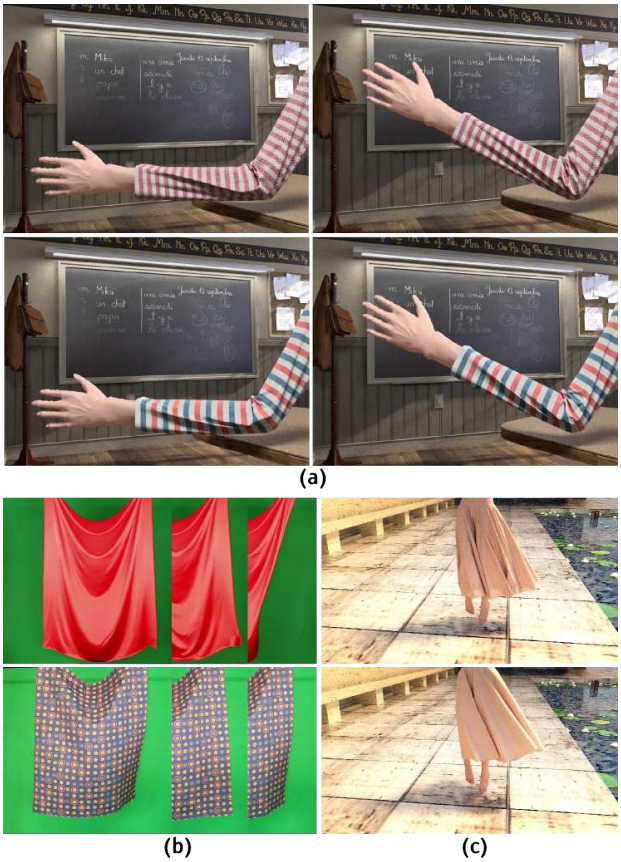

Figure 1. **Learning-based cloth material prediction and material cloning results.** (a) learning samples generated using the state-of-art physically-based cloth simulator Arcsim[38] (b) example real-life cloth motion videos presented in[6] (c) simulated skirt with the material type predicted from the real-life video in (b) using the learned model from samples presented in (a).

The key intuition behind our method is that the visual appearance of a piece of moving cloth encodes the intrinsic material characteristics. We use the parameters of the

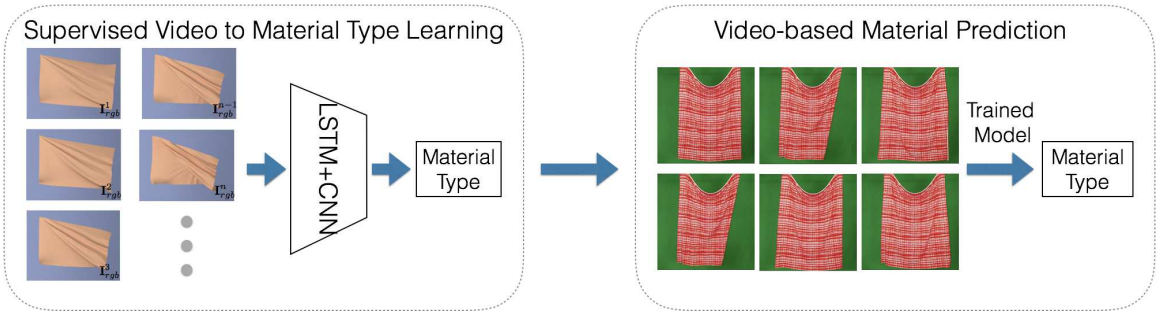

Figure 2. **An overview of our method.** Our cloth material recovery method learns an appearance-to-material mapping model from a set of synthetic training samples. With the learned mapping model, we perform material-type prediction given a recorded video of cloth motion.

material model to represent the cloth's material properties for the recorded fabrics. We adopt the cloth material model proposed by Wang et. al. [51], which encodes the stretching and the bending of the cloth. To quantify the parameter space, we first find a parameter sub-space which discretizes the cloth material type into 54 classes. Each class defines a range of the stretching and bending parameters in the original continuous parameter space. To recover these stretching and bending parameters from the target video, we use machine learning to define the mapping between the "visual features" and the physics properties.

The visual features we use consist of the RGB information of each frame of the video. We assume that the videos are taken in controlled lighting conditions. Furthermore, we take advantages of simulated data from high-fidelity, physically-based cloth simulator to generate a very large set of videos that would be either difficult to obtain or too time-consuming and tedious to capture. With the recovered and tracked moving cloth, we can create a virtual world that uses fabrics with physical properties similar to those of the actual fabrics items in the captured video. The key contributions of this work are: a deep neural network based parameter-learning algorithm and the application of physically-based simulated data of cloth visual-to-material learning. Our dataset as well as the code are available online[1].

2. Related Work

Material Understanding: One of the fundamental problems in computer vision is image and video understanding. It includes the key processes, such as object segmentation [24, 39, 12, 21, 40, 35], object detection [42, 5, 20, 15, 49, 23], object recognition [18, 34, 60, 1, 46, 43], scene understanding [10, 50], human activities and behavior understanding [45, 44, 47], traffic pattern analysis [19, 59], and surface material recognition [14, 3].

Our proposed cloth material understanding is one sub-process of image/video understanding. More recently, "physical scene understanding," which focuses on understanding the intrinsic properties of moving objects [2, 52] has emerged as the next frontier of scene understanding. It is known that human brain can perceive dynamic systems in an image or a video. Inspired by human cognition, our method presents a computational framework that perceives the material properties of cloth in ways similar to how humans perceive dynamical systems in the physical world.

Deep Neural Network for Temporal Pattern Learning: With the advance in the artificial intelligence area, the deep neural network has been used for a vast number of tasks, especially the use of the recurrent neural network in the temporal sequence pattern learning tasks such as activity recognition [17, 27] and video captioning [58]. Our proposed neural network structure is inspired by the LRCN [17].

Use of Synthetic Data-set: The time and the energy needed to label captured data means that there is a limited amount of real-world data for training deep neural networks. Increasingly, researchers are starting to explore the use of synthetic databases to assist a variety of computer vision tasks. For example, Chen et.al. [8] proposed a synthetic human-body data-set to help with 3D pose estimation; Keskin et.al. [29] make use of synthetic hand images to train a hand-pose estimator; and many synthetic pedestrian data-sets [25, 9] have been generated to study computer detection of humans in real-life images/videos.

Recovery of Physical Properties: Recovering the physical properties of a dynamical system has been a challenging problem across computer graphics, medical imaging, robotics, and computer vision for decades. And recovering physical properties of dynamical systems has become especially important with the rise of interest in VR research; the recovered physics properties from a real-life scene can be used in a virtual world or a synthetic environment to recreate a realistic animation of the given dynamical system. For example, in medical image analysis, accurately recreating the physical properties of patient tissues in virtual systems can increase diagnostic accuracy for certain kinds of diseases [54, 55, 53].

Previous methods of recovering physical properties can be classified into to three key categories: measurement-based methods [48, 36, 51], which estimate the physical properties by sampling various physical quantities of the dynamical system; statistically based methods [56, 6, 13], which learn the physical properties by observing

[1] http://gamma.cs.unc.edu/VideoCloth

the statistical parameters of the observed data; and iterative simulation-optimization techniques [4, 54, 57, 33, 37], which recover physical properties by simultaneously simulating the dynamical phenomena and identifying its physical properties. Our method is a hybrid of these three methods. We take advantage of simulations of the dynamical phenomenon for more robust prior computations, and use the statistical method to better learn the intrinsic parameters characterizing the dynamical system, i.e. the moving cloth, in this paper.

Cloth Simulation: Simulation of cloth and garments has been extensively studied in computer graphics [7, 22, 38]. Methods for cloth simulation can be divided into two classes: one focuses on the accuracy of the simulation, and the other tackles the problem of real-time performance [30]. This work takes advantage of the state-of-art cloth simulator, ArcSim [38], which has a high degree of accuracy and visual fidelity.

3. Overview of Our Method

In this section we give a formal definition of the problem.

Problem Statement: Given a sequence of RGB images $\mathcal{V} = \{\Omega_1, \Omega_2, \ldots, \Omega_N\}$, determine the type of material of the recorded cloth.

Figure 2 presents an overview of our approach. To constrain both our input and solution space, we first find the suitable material and the motion sub-space that can best represent the cloth material and motion in real life. Then, we exploit physically based cloth simulations to generate a much larger number of data samples within these subspaces that would otherwise be difficult or time-consuming to capture. The "appearance feature" of the cloth is represented by the pixel $\mathbf{I}_{rgb}$. With the data samples, we combine the image signal feature extraction method, Convolutional Neural Network (CNN), with the temporal sequence learning method, Long Short Term Memory (LSTM), to learn the mapping from visual "appearance" to "material". We list the notations used throughout the paper in Table. 1.

Table 1. **Notations and definition of our method.**

NOTATION	DEFINITION
$\mathcal{V}$	input sequence of images
N	input sequence length
$\mathbf{I}_{rgb}$	RGB channels of the pixel $\mathbf{I}$
$\mathbf{v}$	output of the CNN
W	weights to be learned in the neural network
$\mathcal{M}$	cloth 3D triangle mesh
$\mathcal{P}$	material parameter sub-space
(p, k)	(stretching,bending) parameter in the sub-space

In the following sections, we present details of how our method learns the mapping between the visual appearance of cloth and its physical properties, and information on the generation of synthetic data-sets.

4. Visual, Material and Motion Representation

We first describe the visual appearance feature representation, material parameter space discretization and the motion sub-space of cloth.

4.1. Appearance Representation

We use the convoluted RGB color ($\mathbf{I}_{rgb}$) in the video as the appearance representation. We apply 5 layers of Convolutional Neural Network (CNN) to the RGB channels to extract both low and high-level visual features.

$$\mathbf{v}(\mathbf{I}_{rgb}) = W[\text{CNN}(\mathbf{I}_{rgb})] + b, \tag{1}$$

with W as the weights and b as the bias to be learned. The output of the final fully connected layer (fc6 layer) is the input to the LSTM as the appearance encoding.

4.2. Material Representation

Before we introduce our material representation, we first describe the material model we applied in our physically-based cloth simulator. Instead of using the types of manufacturing material of fabric from the physical world, we use the parameters of the material model of the physically-based simulator as the basis for representing the types of fabric material. Manufacturing fabric material, such as cotton, polyester, and linen, alone does not sufficiently define the material of the cloth. Other factors, such as the weaving patterns and thread count, also affect the material properties of a piece of cloth. Furthermore, since the driving application of this work is virtual try-on for e-commerce, our goal is to automatically determine the set of material parameters required for the physics-based cloth simulator that would reproduce the cloth dynamics observed in the video. The material model in the physically-based cloth simulator defines the cloth behavior under different external forces. The parameters of the material model thus appropriately defines the material type of the cloth under simulation. Therefore, we use the parameters of the material model of the physically-based cloth simulator to represent the types of fabric material in this paper.

4.2.1 Material Model

The choice of material models defines the number of material types that can be approximated using a physically based simulator. In this paper, we use a cloth material model proposed by Wang et al. [51], which can be used to model most of the cloth materials in the real world.

A material model, in general, defines the relation between the stress $\boldsymbol{\sigma}$ and the strain $\boldsymbol{\varepsilon}$. The cloth material consists of two sub-models, stretching and bending models. The stretching model describes how much the cloth would stretch, when subject to a certain amount of planar external forces. Similarly, the bending model defines how much the

cloth would bend, when subject to out-of-plane forces. A linear stress-strain relation can be expressed using a constant stiffness tensor matrix $\mathbf{C}$ as: $\boldsymbol{\sigma} = \mathbf{C}\boldsymbol{\varepsilon}$. To better approximate the stretching physics of a piece of cloth, Wang et al. [51] proposed a stiffness tensor matrix that is not constant but depends on the in-plane-strain tensor $\mathbf{C}(\boldsymbol{\varepsilon})$. We refer readers to our supplementary file for detailed explanation on this material model.

4.2.2 Parameter Space Discretization

In the cloth material model [51], there are 24 and 15 variables in stretching and bending models. This continuous space makes this problem intractable. And individual value of these 24 or 15 variables does not lead to visually perceptible impact. The magnitude of all variables, instead, produces visually perceptible simulation result. To constrain our input/solution space, we discretize the original material parameter space and choose the "quantized" parameter subspace as our material parameter sub-space. Our output will be in this sub-space. To discretize the continuous material parameter space, we choose one of the material presented in the paper [51], called "camel-ponte-roma", as the basis. The material sub-space is constructed by multiplying this material basis with a positive coefficient. We further quantize the coefficients in continuous space to a discrete set of numbers. The size of this discrete set of numbers is the number of material types we used to represent the cloth material in real life. Using this mechanism, we discretized both the stretching and the bending parameter space.

To construct an optimal material parameter sub-space $\mathcal{P}$, optimal in the sense that the size of the coefficient set is minimized and the number of different real-life cloth materials that can be represented is maximized, we first conduct a material parameter sensitivity analysis. The material parameter sensitivity analysis examines the sensitivity of the material parameters κ with respect to the amount of deformation $D(\kappa)$. The sensitivity is measured as: $\frac{\partial D(\kappa)}{\partial \kappa}$, which is the slope of the curve shown in Fig. 3. For the stretching parameter p analysis, we hang a piece of cloth and measure the maximum amount of stretching $D(\mathcal{M})$ as in the length changes, when subjected to gravity. And, for bending parameter k sensitivity analysis, we fold a piece of cloth and keep track with the maximum curvature $C(\mathcal{M})$. The maximum amount of stretching $D(\mathcal{M})$ and the maximum curvature $C(\mathcal{M})$ are measured from the 3D mesh $\mathcal{M}$ as follows:

$$D(\mathcal{M}) = \max_{\mathbf{u} \in V} \|\mathbf{u} - \mathbf{u}_0\|, \tag{2}$$

$$C(\mathcal{M}) = \max_{f_1, f_2 \in F, f_1 \cap f_2 = \mathbf{e}_0} \frac{\|\mathbf{e}_0\| arccos(\mathbf{n}_1 \cdot \mathbf{n}_2)}{A_1 + A_2}, \tag{3}$$

where $\mathcal{M} = V, F, E$ is the 3D triangle mesh, which has a vertex set V, a face set F and an edge set E, of the cloth, $\mathbf{u}$ is a vertex of the cloth's mesh $\mathcal{M}$ and $\mathbf{u}_0$ is position of that vertex in rest configuration, f_1, f_2 are two adjacent faces with shared edge $\mathbf{e}_0$, $\mathbf{n}_1, \mathbf{n}_2$ are their normals and A_1, A_2 are the area of those two faces.

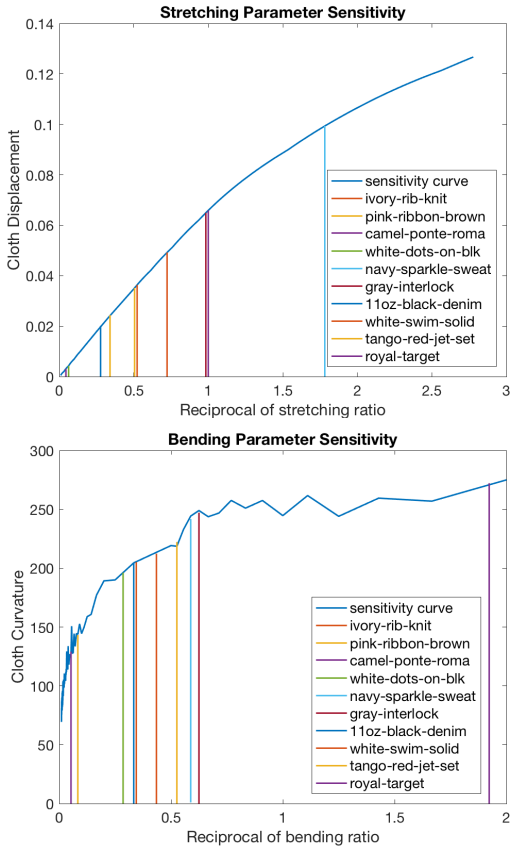

Figure 3. **Stretching and bending parameters sensitivity analysis results. (best view in color)** The x-axis is the reciprocal of parameter ratios to the basis material. The reciprocal of stretching ratio for the "camel-ponte-roma" material is 1. The y-axis is the maximum amount of deformation of the cloth, i.e., maximum amount of stretching or maximum curvature, respectively. We use the vertical lines with different colors to represent the 10 types of materials presented in [51]. The jittering in the bottom figure is due to the adaptive remeshing.

The analysis results are shown in Fig. 3. The slope of the sensitivity curve (light blue) in Fig. 3 is positively related to how sensitive the cloth deformation/curvature is with respect to the stretching/bending coefficient. The jittering in the bending parameter sensitivity analysis is due to the remeshing scheme. We further divide the x-axis in Fig. 3 into a set of segments based on the slope of the sensitivity curve. We divide the x-axis into more discrete sets when the slope of the sensitivity curve is large and vice versa. The discrete set segments of the x-axis are the stretching/bending coefficients set. Based on our analysis, the stretching parameter sub-space is $\mathcal{P}_s = \{0.5, 1, 2, 3, 10, 20\}$ and the bending parameter sub-space is $\mathcal{P}_b = \{0.5, 1, 2, 3, 4, 5, 10, 15, 20\}$. Combining the two sub-spaces $\mathcal{P} = \{(p, k) | p \in \mathcal{P}_s, k \in \mathcal{P}_b\}$, our discretized sub-space can represent 54 types of material.

Table 2. **Material parameter sub-space validation.** The floating point numbers show the estimated stretching/bending parameter coefficients $(\tilde{p}, \tilde{k})$, while the numbers in the parenthesis are the corresponding stretching/bending parameter (p, k) in our defined subspace $\mathcal{P}$.

Material [51]	Stretching Ratio $\tilde{p}(p)$	Bending Ratio $\tilde{k}(k)$
ivory-rib-knit	1.3817(1)	2.3(2)
pink-ribbon-brown	2.9343(3)	12(10)
camel-ponte-roma	1(1)	0.52(0.5)
white-dots-on-blk	15.8108(20)	3.5(4)
navy-sparkle-sweat	0.5613(0.5)	1.7(2)
gray-interlock	1.0164(1)	1.6(2)
11oz-black-denim	3.6079(3)	3(3)
white-swim-solid	1.9126(2)	2.9(3)
tango-red-jet-set	1.9784(2)	1.9(2)
royal-target	22.2857(20)	19(20)

To prove the validity of our material parameter sub-space, we illustrate that our material types have the ability to represent some of the commonly encountered real-life fabric material classes. We use the ten material types presented in the paper [51] for the validation experiment. Firstly, we estimate the parameters (the floating point numbers $(\tilde{p}, \tilde{k})$ in Table 2). And then we discretize them into our subspace (the numbers in the parenthesis(p, k) in Table 2). As shown in Table 2, our discretized material types can represent these 10 types of cloth with a limited amount of error.

4.3. Motion Sub-space

To further make our problem tractable, we constrain the motion space of the cloth by controlling the external forces of the cloth. Under controlled external forces, the cloth moves in a motion sub-space. In addition, we need to make sure that the motion subspace is spanned in a way to capture the relation between the motion and the material properties of the cloth. We choose two types of external forces: constant-velocity wind blowing and fixed-size arm bending. The constant-velocity wind blowing can stretch the cloth to its maximum amount of stretching deformation, while the fixed-size arm bending can bend the cloth to its highest curvature.

5. Learning Method

In this section, we explain how to establish the mapping between the visual appearance of a moving cloth and its physical properties using deep neural network.

5.1. Deep Neural Network Structure

Design Rationale: We propose to combine CNN with LSTM (similar to the LRCN [16] structure) for our appearance-to-material learning (network structure shown in Fig. 4). CNN is used to extract both low- and high-level visual features. LSTM part of the network focuses on temporal motion pattern learning. In the following sections, we will briefly introduce our network structure.

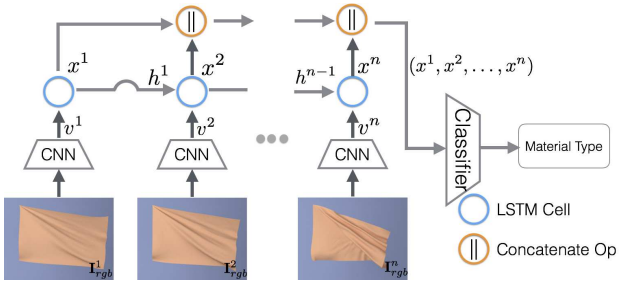

Figure 4. **Appearance-to-material learning method.** We apply CNN and LSTM (the original LRCN design presented in [16]) to learn the mapping between appearance and material.

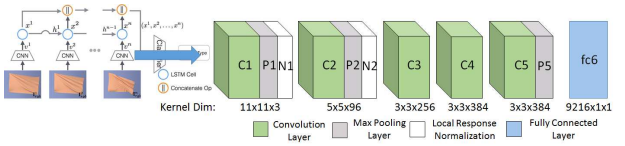

Figure 5. **The five-layer CNN structure.** The original design is presented in [31]

Convolutional Neural Network for Hierarchical Visual Feature Learning: Convolutional neural network was first proposed by LeCun et. al. [32] for digit recognition. The basis of the convolutional neural network is the convolution operation. The convolution operation serves as a filtering operation on an image. Layers of convolutional neural network (CNN) with convolution kernels of different dimensions extract features at various levels of details.

We applied a five-layer CNN (shown in Fig. 5) for its ability in hierarchical visual feature selection. This part of the network structure is similar to the AlexNet [31]. The fifth convolution layer is followed by one fully connected layer. The output of the fully connected layer (fc6) is the input to each LSTM cell.

To demonstrate the effectiveness of our CNN design, we visualize the activation of the fifth convolution layer (the "conv5" layer). In Fig. 6, we overlay the real-life cloth moving images with the "conv5" layer activation which is visualized using the "jet" color map. The model is trained with our simulated wind-blowing data set. It is shown that we successfully trained the neural network in paying attention to the cloth area (highlighted in yellow-red) and the cloth moving edges (highlighted in red) of real-life images.

Recurrent Neural Network for Sequential Pattern Learning: A single image contains a limited amount of information concerning the physics properties of a piece of cloth. But a video can be more powerful to demonstrate how the physics properties, such as the material properties of a piece of cloth, can affect its motions. To approximate this mapping between the material properties of the cloth

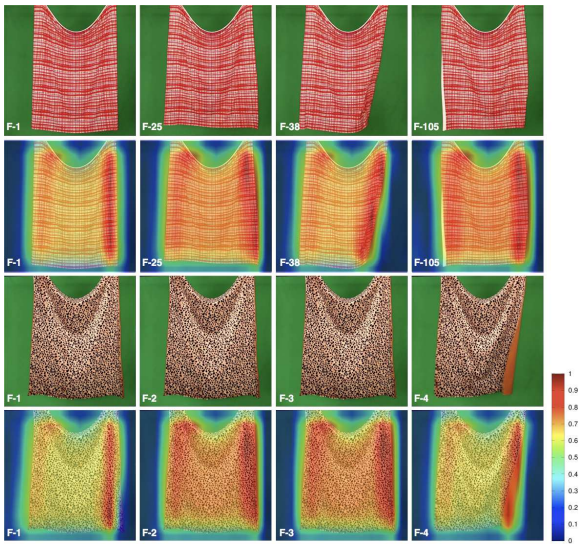

Figure 6. **Learned CNN conv5-layer activation visualization. (best view in color)** In 2nd and 4th rows, we overlay the conv5 layer activation using the "jet" color map with the original image in 1st and 3rd rows. The columns are different frames, with the frame number shown in the bottom left of each image. The model is trained with simulated data set from the wind-blowing motion.

and its sequential movement, we apply the recurrent neural network. Unlike the feed-forward neural network, the recurrent neural network has a feedback loop. The loop connects the output of the current cell to the input of the cell at the next step. The feedback loop act as the "memory" of the recurrent neural network. With the "memory", the recurrent neural network has the ability to gradually extract the pattern of the input sequence.

Following the intuition behind the recurrent neural network, we choose the LSTM [26] instead of the traditional recurrent neural network architecture for its ability to deal with vanishing/exploding gradient and fast convergence to learn the pattern in temporal sequence of data.

6. Physics-based Synthetic Data-sets

To learn the mapping between the visual appearance of a moving cloth and its material characteristics using a statistical method, we require a large number of data samples. Instead of using limited number of real-life recorded videos of cloth moving, we use simulation data as training samples. Our synthetic data generation exploits physically based cloth simulation. This approach enables us to automatically generate a large number of data samples in a short amount of time without any manual recording or labeling.

In the following section, we will introduce our learning data samples generation pipeline.

6.1. Data Generation

Our data generation pipeline (shown in Fig. 7) consists of two steps: cloth simulation and image rendering. The cloth

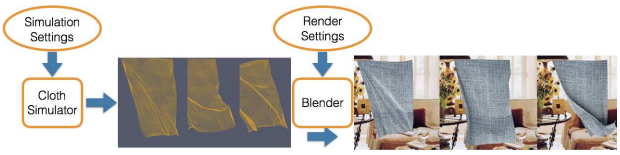

Figure 7. **Data generation pipeline.** The pipeline consists of two steps: cloth simulation and image rendering.

Figure 8. **Simulated data showcase.** The first three rows are example frames from our Wind-blowing data set with the cloth in pose-1. The bottom two rows are example frames from our Wind-blowing simulated data set with the cloth in pose-2 consisting of two different types of material.

meshes are generated through physically based simulation (ArcSim [38]). The cloth is subject to external forces such as gravity, wind and arm bending. Those external forces will drive the movement of the cloth. We vary the external force magnitude, stretching and bending parameters to simulate a number of sequences of cloth motion. Since we discretize the magnitude of 24 and 15 variables of the stretching and bending model into 6 and 9 bins, we jitter around the discretized values to generate the training dataset. For each set of magnitude of external forces, stretching, bending parameters we generate a sequence of 3D cloth meshes. The sequence is divided into sub-sequences as temporal training samples. Then we render each frame of the 3D cloth meshes to 2D images using Blender. The images are rendered under controlled lighting conditions and camera settings. Instead of rendering the cloth as uniform colored, for each sequence of 3D mesh, we randomly assign them with a texture image. We further composite the foreground cloth with a random background image (as shown in Fig. 8) to make the scene more complicated and to train the network to pay attention to the cloth area (as shown in Fig. 6). Our background images are chosen from the indoor scene image dataset [41].

7. Experiments

We implemented our method using the Caffe [28] deep neural network framework. The training process takes around 12 hours with a NVDIA-Titanx^{TM} GPU. It takes up to 40,000 iterations to converge.

7.1. Data Preparation

Our data set is generated using physically-based cloth simulation. By changing the simulation parameters, we obtain cloth with different material properties. We also observed two key factors that can affect the learning process: the remeshing scheme (adaptive vs. uniform resolution) and the texture of the cloth. The remeshing scheme affects the wrinkle formulation of the simulated cloth, while the texture affects the visual feature that the CNN can extract. For each motion, remeshing scheme, and texture type, we generated 2,592 sequences of cloth motion, using the method we introduced in Sec. 6.1. Among the 2,592 sequences, 2,106 of them were used for training and the rest 432 were used for testing. Each sequence consists of 10 frames. We tested our learned model on both the simulated data set and the real-life videos.

7.2. Results

Our training data consists of two different types of motion: arm bending and wind blowing, with 54 material types (by varying a combination of 6 bending and 9 stretching parameters).

7.2.1 Baseline Results

To validate our network structure, we constructed two baseline tests. Our first baseline test excludes the sequential pattern learning part (LSTM). We fine-tune the pre-trained AlexNet [31] with all the frames (210,600 images) of our training videos. Then we test our fine-tuned model on simulated data (43,200 images). Test results are shown in Table 3. Our first baseline framework achieves 53.6% of accuracy for predicting 54 classes of materials for arm bending motion and that of 56.9% for wind blowing motion when testing on simulated data. For the second baseline, we fix our CNN part of the network but train the LSTM part. The second test aims to validate the effectiveness of our CNN sub-network. This framework obtains 56.9% of accuracy for predicting 54 classes of materials for arm bending motion and that of 57.0% for wind blowing motion when testing on simulated data. As is shown in Table 3 and Table 4, the accuracy for cloth material type prediction from both simulated images/videos and real-life images/videos for both baseline frameworks is lower than our (CNN+LSTM) model. Our third baseline replace the LSTM with vanilla RNN. When trained on simulated data and tested on simulated data, the accuracy of the vanilla RNN is about 20% lower. When tested on real-life videos, the r-value is around 0.2 lower than our model (refer to supplementary document).

Table 3. **Testing results.** The models are trained with the arm bending motion and wind blowing motion. Then they are tested on 432 simulated arm bending/wind blowing videos, where the ground truth is known. Our method achieved up to 71.8% of accuracy for predicting 54 classes of materials for arm bending motion and up to 66.7% for wind blowing motion.

Data Setting \ Method			Base-1	Base-2	**CNN+LSTM**
Type	Re-mesh	Texture	RGB-I	RGB-V	RGB-V
Arm	Adapt	Grid	56.0	54.0	63.3
		Color	52.9	50.2	66.0
		Rand	53.0	54.3	71.1
	Unif-1	Rand1	54.0	56.8	62.9
	Unif-2	Rand2	51.9	57	62.7
	Unif-3	Rand3	53.6	56.9	**71.8**
Wind	Adapt	Grid	50.4	48.0	63.4
		Color	54.0	51.2	68.0
		Rand	53.7	53.2	67.7
	Unif-1	Rand1	53.6	53.3	64.7
	Unif-2	Rand2	58.9	57.0	64.5
	Unif-3	Rand3	56.9	53.0	**66.7**

7.2.2 Validation of Our Method

To validate our method, We first test the accuracy of the model trained with only the simulated arm bending motion for predicting material type of the arm bending videos. We achieve up to 71.8% of accuracy for predicting from the 54 classes of material types when using only the three-channel RGB video. The model that has the best accuracy is the one trained with the texture randomly assigned and the mesh uniformly remeshed three times. And the second best model is the one trained with the adaptive remeshing scheme and randomly assigned texture. The main reason behind this is that the meshes that are uniformly remeshed three times contain more details than the adaptive remeshed ones.

Next, we train the deep neural network with the wind blowing motion data set and test the learned model on the simulated wind blowing videos. The results are shown in Table 3. Similar to the arm bending results, the best performing model is the one that is trained with the texture randomly assigned and the mesh uniformly remeshed three times. We achieve up to 66.7% of accuracy for predicting among 54 material types when using only the three-channel RGB Video.

Finally, we test our learned model on 90 real-life videos [6]. The 90 videos record the wind blowing motion of 30 kinds of cloth with three different wind strength. We correlate our predicted material type with both the ground truth stiffness value and the ground truth density value. Our material types are 54 discrete numbers range from 0 to 53. The higher the number generally means the cloth is stiffer. Among the models we trained, the one which is trained with the wind blowing motion, uniformly remeshed three times,

Table 4. **Stiffness/density correlation r values for [6] vs. Ours** Our method outperforms both [6] and human perception, achieving the highest correlation value of 0.77 and 0.84 respectively for stiffness and density, undergoing large motion due to stronger wind (W3-video). W1, W2, W3 indicates different wind strength. The larger the number, the stronger the wind.

Method	Input	Stiffness	Density
Human [6]	Image	0.48	0.45
Human [6]	Video	0.73	0.83
AlexNet (baseline1)	Image	0.04	0.06
preCNN+LSTM (baseline2)	30 W3-Videos	0.12	0.13
CNN+LSTM (ours)	30 W1-Videos	0.47	0.55
CNN+LSTM (ours)	30 W2-Videos	0.43	0.62
CNN+LSTM (ours)	30 W3-Videos	0.50	0.64
K. Bouman et. al. [6]	23 W1-Videos	0.74	0.77
K. Bouman et. al. [6]	23 W2-Videos	0.67	0.85
K. Bouman et. al. [6]	23 W3-Videos	0.70	0.77
CNN+LSTM (ours)	23 W1-Videos	0.71	0.75
CNN+LSTM (ours)	23 W2-Videos	0.69	0.80
CNN+LSTM (ours)	23 W3-Videos	**0.77**	**0.84**

texture randomly assigned performs the best on both simulated data according to Table 4.

The prediction from this model also correlates the best with both the ground truth stiffness value and the ground truth density value. We achieve up to 0.50 and 0.64, respectively, as of the R value which is close to the one when human predicting material from a single image presented in [6] for the correlation test. Our experiment results also show that our prediction results is sensitive to the cloth motion as the predicted material type correlate better with the ground truth values as the strength of the wind increases. The wind intensity (indicated as W1, W2, W3) affects the prediction results by influencing motion of the cloth. Our trained model performs best on videos taken with the maximum wind strength (W3) shown in Table 4. Further comparison analysis is given in the following section.

7.2.3 Comparison

In Table 4, we compare our method with the other cloth material recovery methods [6] that addresses the same problem as ours. Inspired by the feature selection proposed in [6], we propose a more general feature extraction method based on deep neural network. To make fair comparison with K. Bouman et.al. [6], we also removed 7 videos which lack of texture or of high specularity. After excluding those 7 videos, our correlation coefficient R value for predicting cloth stiffness is 0.77 which is higher than those presented in [6]. We demonstrated in experiments that our learned model can predict material type from videos more accurately than using features in [6] and human perception.

7.3. Application

We further demonstrate our proposed framework with the application of "material cloning". It is a 2-step process: fist identify the material type from the video, then apply the identified material type to physically-based cloth simulation. With our trained deep neural network model, we can predict the material type from a video recording the motion of the cloth in a fairly small amount of time. The recovered material type can be "cloned" on another piece of cloth or a piece of garment as shown in Fig. 9. We refer readers to our supplementary file for video demos.

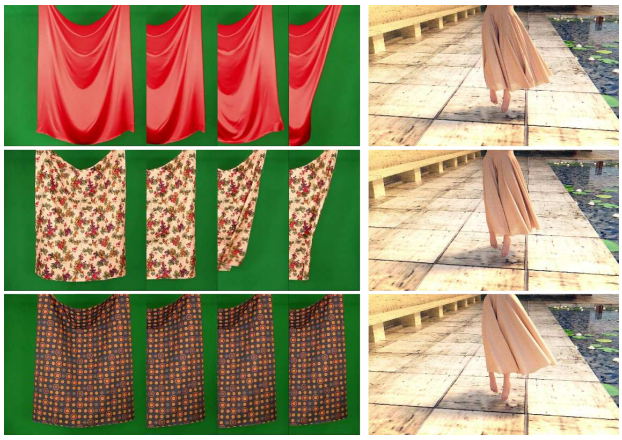

Figure 9. **Material cloning results.** The first column are the input cloth motion videos[6]. We predict the material type of the cloth in these input videos and clone those material onto the skirt. The simulated skirt are shown in the second column.

7.4. Discussion and Limitations

Our current learning samples are generated using physics simulator. There are differences between the simulated data and real-life recorded videos, due to the numerical errors in the cloth simulation and also the quality of the rendered images. Our experiments show great promise of our learned model using data from simulator in predicting material types of cloth in the real-life videos. With a more accurate simulator and more photorealistic rendering, the proposed framework can learn a better model from sampled simulation data. The neural network structure can also be further improved for cross-domain learning.

8. Conclusion and Future Work

We have presented a learning-based algorithm to recover material properties from videos, using training datasets generated by physics simulators. Our learned model can recover physical properties (e.g. fabric material) of the cloth from a video. Our training videos contain only a single piece of cloth and the recorded cloth is not interacting with any other object. While this is not always the case in real-world scenarios, this method provides new insights to a challenging problem. A natural extension would be to learn from videos of cloth directly interacting with the body, under varying lighting conditions and partial occlusion.

Acknowledgment: This research is supported in part by National Institute of Health, National Science Foundation, and UNC Arts & Science Foundation.

References

[1] M. Aubry, D. Maturana, A. A. Efros, B. C. Russell, and J. Sivic. Seeing 3d chairs: exemplar part-based 2d-3d alignment using a large dataset of cad models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3762–3769, 2014. 2

[2] P. W. Battaglia, J. B. Hamrick, and J. B. Tenenbaum. Simulation as an engine of physical scene understanding. *Proceedings of the National Academy of Sciences*, 110(45):18327–18332, 2013. 2

[3] S. Bell, P. Upchurch, N. Snavely, and K. Bala. Material recognition in the wild with the materials in context database. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3479–3487, 2015. 2

[4] K. S. Bhat, C. D. Twigg, J. K. Hodgins, P. K. Khosla, Z. Popović, and S. M. Seitz. Estimating cloth simulation parameters from video. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 37–51. Eurographics Association, 2003. 3

[5] A. Borji, M.-M. Cheng, H. Jiang, and J. Li. Salient object detection: A benchmark. *IEEE Transactions on Image Processing*, 24(12):5706–5722, 2015. 2

[6] K. L. Bouman, B. Xiao, P. Battaglia, and W. T. Freeman. Estimating the material properties of fabric from video. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1984–1991, 2013. 1, 2, 7, 8

[7] R. Bridson, R. Fedkiw, and J. Anderson. Robust treatment of collisions, contact and friction for cloth animation. In *ACM Transactions on Graphics (ToG)*, volume 21, pages 594–603. ACM, 2002. 3

[8] W. Chen, H. Wang, Y. Li, H. Su, D. Lischinsk, D. Cohen-Or, B. Chen, et al. Synthesizing training images for boosting human 3d pose estimation. *arXiv preprint arXiv:1604.02703*, 2016. 2

[9] E. Cheung, T. K. Wong, A. Bera, X. Wang, and D. Manocha. Lcrowdv: Generating labeled videos for simulation-based crowd behavior learning. *arXiv preprint arXiv:1606.08998*, 2016. 2

[10] W. Choi, Y.-W. Chao, C. Pantofaru, and S. Savarese. Understanding indoor scenes using 3d geometric phrases. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 33–40, 2013. 2

[11] K. J. W. Craik. *The nature of explanation*, volume 445. CUP Archive, 1967. 1

[12] J. Dai, K. He, and J. Sun. Convolutional feature masking for joint object and stuff segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3992–4000, 2015. 2

[13] A. Davis, K. L. Bouman, J. G. Chen, M. Rubinstein, F. Durand, and W. T. Freeman. Visual vibrometry: Estimating material properties from small motion in video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5335–5343, 2015. 2

[14] A. DelPozo and S. Savarese. Detecting specular surfaces on natural images. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007. 2

[15] P. Dollár, R. Appel, S. Belongie, and P. Perona. Fast feature pyramids for object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(8):1532–1545, 2014. 2

[16] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2625–2634, 2015. 5

[17] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*, 2015. 2

[18] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *ICML*, pages 647–655, 2014. 2

[19] A. Geiger, M. Lauer, C. Wojek, C. Stiller, and R. Urtasun. 3d traffic scene understanding from movable platforms. *IEEE transactions on pattern analysis and machine intelligence*, 36(5):1012–1025, 2014. 2

[20] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014. 2

[21] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Region-based convolutional networks for accurate object detection and segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 38(1):142–158, 2016. 2

[22] N. K. Govindaraju, I. Kabul, M. C. Lin, and D. Manocha. Fast continuous collision detection among deformable models using graphics processors. *Computers & Graphics*, 31(1):5–14, 2007. 3

[23] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Simultaneous detection and segmentation. In *European Conference on Computer Vision*, pages 297–312. Springer, 2014. 2

[24] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Hypercolumns for object segmentation and fine-grained localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 447–456, 2015. 2

[25] H. Hattori, V. Naresh Boddeti, K. M. Kitani, and T. Kanade. Learning scene-specific pedestrian detectors without real data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3819–3827, 2015. 2

[26] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 6

[27] M. S. Ibrahim, S. Muralidharan, Z. Deng, A. Vahdat, and G. Mori. A hierarchical deep temporal model for group activity recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1971–1980, 2016. 2

[28] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014. 7

[29] C. Keskin, F. Kıraç, Y. E. Kara, and L. Akarun. Real time hand pose estimation using depth sensors. In *Consumer Depth Cameras for Computer Vision*, pages 119–137. Springer, 2013. 2

[30] W. Koh, R. Narain, and J. F. O'Brien. View-dependent adaptive cloth simulation. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 159–166. Eurographics Association, 2014. 3

[31] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 5, 7

[32] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 5

[33] H.-P. Lee, M. Foskey, M. Niethammer, P. Krajcevski, and M. C. Lin. Simulation-based joint estimation of body deformation and elasticity parameters for medical image analysis. *IEEE transactions on medical imaging*, 31(11):2156–2168, 2012. 3

[34] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, pages 740–755. Springer, 2014. 2

[35] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015. 2

[36] E. Miguel, D. Bradley, B. Thomaszewski, B. Bickel, W. Matusik, M. A. Otaduy, and S. Marschner. Data-driven estimation of cloth simulation models. In *Computer Graphics Forum*, volume 31, pages 519–528. Wiley Online Library, 2012. 2

[37] D. Mongus, B. Repnik, M. Mernik, and B. Žalik. A hybrid evolutionary algorithm for tuning a cloth-simulation model. *Applied Soft Computing*, 12(1):266–273, 2012. 3

[38] R. Narain, A. Samii, and J. F. O'Brien. Adaptive anisotropic remeshing for cloth simulation. *ACM transactions on graphics (TOG)*, 31(6):152, 2012. 1, 3, 6

[39] H. Noh, S. Hong, and B. Han. Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1520–1528, 2015. 2

[40] P. O. Pinheiro and R. Collobert. From image-level to pixel-level labeling with convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1713–1721, 2015. 2

[41] A. Quattoni and A. Torralba. Recognizing indoor scenes. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 413–420. IEEE, 2009. 6

[42] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015. 2

[43] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. 2

[44] J. Shao, C. Change Loy, and X. Wang. Scene-independent group profiling in crowd. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2219–2226, 2014. 2

[45] J. Shao, K. Kang, C. C. Loy, and X. Wang. Deeply learned attributes for crowded scene understanding. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4657–4666. IEEE, 2015. 2

[46] A. Sharif Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. Cnn features off-the-shelf: an astounding baseline for recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 806–813, 2014. 2

[47] B. Solmaz, B. E. Moore, and M. Shah. Identifying behaviors in crowd scenes using stability analysis for dynamical systems. *IEEE transactions on pattern analysis and machine intelligence*, 34(10):2064–2070, 2012. 2

[48] C. Syllebranque and S. Boivin. Estimation of mechanical parameters of deformable solids from videos. *The Visual Computer*, 24(11):963–972, 2008. 2

[49] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2015. 2

[50] H. Wang, S. Gould, and D. Roller. Discriminative learning with latent variables for cluttered indoor scene understanding. *Communications of the ACM*, 56(4):92–99, 2013. 2

[51] H. Wang, J. F. O'Brien, and R. Ramamoorthi. Data-driven elastic models for cloth: modeling and measurement. *ACM Transactions on Graphics (TOG)*, 30(4):71, 2011. 2, 3, 4, 5

[52] J. Wu, I. Yildirim, J. J. Lim, B. Freeman, and J. Tenenbaum. Galileo: Perceiving physical object properties by integrating a physics engine with deep learning. In *Advances in Neural Information Processing Systems*, pages 127–135, 2015. 2

[53] S. Yang, V. Jojic, J. Lian, R. Chen, H. Zhu, and M. C. Lin. Classification of prostate cancer grades and t-stages based on tissue elasticity using medical image analysis. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 627–635. Springer, 2016. 2

[54] S. Yang and M. Lin. Materialcloning: Acquiring elasticity parameters from images for medical applications. *IEEE transactions on visualization and computer graphics*, 2015. 2, 3

[55] S. Yang and M. Lin. Simultaneous estimation of elasticity for multiple deformable bodies. *Computer animation and virtual worlds*, 26(3-4):197–206, 2015. 2

[56] S. Yang and M. C. Lin. Bayesian estimation of non-rigid mechanical parameters using temporal sequences of deformation samples. In *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, pages 4036–4043. IEEE, 2016. 2

[57] S. Yang, Z. Pan, T. Amert, K. Wang, L. Yu, T. Berg, and M. C. Lin. Physics-inspired garment recovery from a single-view image. *arXiv preprint arXiv:1608.01250*, 2016. 3

[58] H. Yu, J. Wang, Z. Huang, Y. Yang, and W. Xu. Video paragraph captioning using hierarchical recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4584–4593, 2016. 2

[59] H. Zhang, A. Geiger, and R. Urtasun. Understanding high-level semantics by modeling traffic patterns. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3056–3063, 2013. 2

[60] N. Zhang, J. Donahue, R. Girshick, and T. Darrell. Part-based r-cnns for fine-grained category detection. In *European Conference on Computer Vision*, pages 834–849. Springer, 2014. 2