# Coupled Visual-Physical Learning for Cloth Parameter Estimation Using Single-View Images

**William B.S. Chambers**[*]
Department of Computer Science
University of Maryland
College Park, Maryland
will1996@umd.edu

**Caroline Horsch**[*]
Department of Computer Science
University of Maryland
College Park, Maryland
chorsch@umd.edu

**Junbang Liang**
Department of Computer Science
University of Maryland
College Park, Maryland
liangjb@umd.edu

**Ming C. Lin**
Department of Computer Science
University of Maryland
College Park, Maryland
lin@umd.edu

## Abstract

Material parameter estimation for fabric is essential for virtual try-on, cloth simulation, and fashion design. Previous works often approached the problem from a physics perspective, requiring multiple lab measurements, images, videos, or simulated data to estimate cloth physical parameters using optimization techniques. These techniques are not practically achievable in the real world. In this paper we present a novel method which uses a visual approach to achieve cloth parameter estimation from a single photograph. The key insight of our method is that the intrinsic cloth properties of fiber material, thread thickness, and weave/knit pattern dictate both the physics and the appearance of the cloth, and so should be obtainable from visual means, removing the need for dynamics information. To learn these properties, we use a differentiable renderer to optimize the parameters of a target rendering model, extracting shine, roughness, and a normal map via gradient descent. By leveraging the periodic nature of cloth structure, we use Fourier analysis to extract general information about thread thickness, weave/knit pattern, and thread density from the normal map. This information is used to construct a classifier that discriminates between physical cloth types in a manner independent of variations in color, rotation, scale, and cloth deformation. We demonstrate the robustness of our method by testing the classifier on several different colors, orientations, and deformations of cloth. Our method provides comparable accuracy in material recovery to previous methods, while using only single-view images, which can be easily acquired from commodity hand-held devices.

## 1 Introduction

Lifelike cloth materials are in high demand for VR, Animation, and E-commerce. All of these domains require that fabrics act as they do in our common experience to maintain immersion, or to provide accurate feedback to the customer about the fit/look of their garment before purchase. Cloth parameter estimation is required to bring real materials into simulation accurately, however it remains an open problem due to the substantial complexity of cloth simulation, and further the solutions that

[*]Equal Contribution

Preprint. Under review.

exist are impractical for use by industry because they require lab-environments to function [1, 2], need labeled 3D meshes, or work only on simulated data [10]. A practical solution should require no lab setup, function on extremely little data, and provide satisfactory results. Inspiration for a solution to this problem came from two observations. 1. that there are only on the order of 100 categories of cloth, which are visually and physically distinct from one another. 2. The Fine structural details which identify the type of cloth are readily obvious from the cloths appearance [9, 8]. Due to the first observation, our method takes a classification approach which, due to the second observation, operates on the underlying structure of the cloth. Working at this level allows us to classify the intrinsic material of the cloth itself. This is essential as a naive image classification approach would require an unbounded amount of data to handle orientation, deformation, lighting changes, and fabric dye colors. Our method can be viewed as a three stage pipeline. First, we use a differentiable renderer to extract color, cloth deformation, specular reflectance, roughness, and a normal map. Second, we perform signal analysis on the normal map to extract measures of thread level features in a rotation and scale invariant manner, while being general enough to measure knitted, and woven fabrics. Finally we construct a Quadratic Discriminant Analysis model to provide good classification performance on the low volume of non-linearly correlated feature data. Our key contributions include:

- A differentiable rendering scheme which re-constructs 3D cloth meshes from single images at extremely high quality
- A Signal Analysis procedure for extracting thread-level cloth detail from normal maps
- A Quadratic Discriminant Analysis model for classifying cloth based on single views
- A dataset of labeled single-view real-world cloth images

## 2 Related Work

### 2.1 Cloth Parameter Estimation

Cloth parameter estimation is a method of re-creating real-world cloth in simulation by accurately defining the parameters of the simulation to capture real physics behavior. [1] used video clips to compare sequences of cloth, optimizing cloth parameters to fit frames. [12] Used a similar technique updated with learning based methods to track edges, and further classifying the cloth into 1 of 54 classes of cloth determined by a sensitivity study, this approach is data hungry, and uses simulated data to pad out its datasets, it also requires video. [5] created a differentiable cloth simulator capable of performing gradient descent optimization on a set of physical parameters using 3D mesh loss. This approach is limited by the deformations of the provided meshes in the dataset, parameters exercised by motions not in the dataset receive zero gradient, and cannot be optimized for. Also the approach requires labeled 3D meshes as input. [2] seeks to fit cloth parameters such that the simulated cloth matches lab measured results via complicated optimization methods, requires lab measurements for each desired fabric type, and suffers from degenerate solutions. In contrast our method requires only static, single view, RGB images. [10] uses a differentiable renderer to backpropagate image loss to a differentiable physics system. It jointly optimizes the optical parameters and physics parameters, and provides good results for sequences of simulated data. However it doesn't attempt to work on real-world data, requires generous initialization of meshes and parameters for success, and requires videos. Our method by contrast requires only single view images, which are easily acquirable using commodity smart phones.

### 2.2 Differentiable Rendering

Our inverse renderer is enabled by differentiable rendering, a fairly new field which makes the operations required to render 3D geometry to an image differentiable, thus enabling back-propagation of image loss to scene parameters such as geometry, colors, and textures. Performing gradient descent with this information allows optimization of these parameters, which in effect inverts the process of rendering. Differentiable rendering was recently improved to allow for differentiable path tracing, which substantially improves the ability of the renderer to fit real-world images by removing the need for approximations to handle discontinuities in the rendering process [6], and by handling global illumination. Along with great advances in the differentiability of renderers there has been great work on the optimization of meshes via differentiable rendering from [11], which provides an optimization technique that greatly improves the quality of mesh recovery.

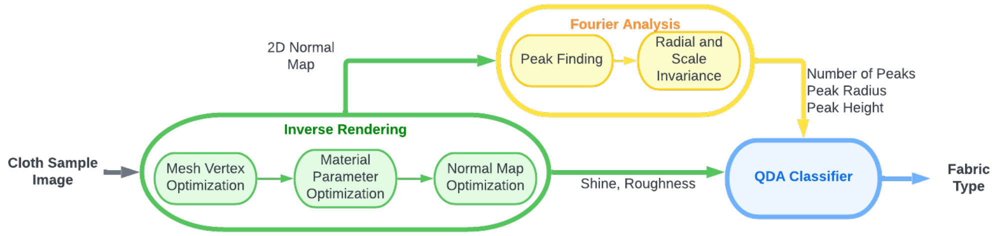



Figure 1: **Overview of the visual parameter extraction, analysis, and classification pipeline**. Beginning with a single cloth image sample, we start by performing inverse rendering to extract the shine, roughness, and normal map. The normal maps are passed to the Fourier analysis which extracts information about the thread structure of the cloth sample by finding number of peaks, their peak height, and peak radius in the Fourier transform. The information extracted from both the inverse rendering and the Fourier analysis are features used to classify the cloth sample using QDA classification. The final output of our pipeline is the fabric type we have determined the cloth sample to be. This fabric type can then be mapped to physical material parameters in various ways.

### 2.3 Cloth Rendering

Cloth rendering is the process of realistically representing cloth. Unlike other materials, e.g solid structures, the microstructure of the cloth plays a huge role in its macro appearance [9, 7, 8, 14, 13]. This has inspired a huge amount of work in the study of how to represent these structures visually. These works inspire our method, because it would not be necessary to represent the cloth micro structure for rendering if it wasn't very visible in images. Thus we should be able to extract information about the micro structure from the images alone.

## 3 Methods

### 3.1 Pipeline Overview

The pipeline starts with an inverse renderer which extracts color, specular reflectance, roughness, and a normal map from a single-view image of cloth. These parameters are then processed into a feature vector containing magnitude of specular reflectance, roughness, and information characteristic of the cloth structure extracted from the normal map.

### 3.2 Inverse Rendering

The first part of our pipeline is the extraction of all visual characteristics relevant to fabric identification. This is achieved by using inverse rendering to reconstruct the cloth as a 3D mesh with a visually accurate material. Using a differentiable renderer allows for optimization via gradient descent of parameters that contribute to a final rendered image. To perform the inverse rendering, we used the Redner differentiable renderer which supports optimization of the vertex and material parameters needed for our extraction [4]. We perform a three-step optimization for cloth reconstruction: vertex position optimization to deform the mesh to match the deformation of the cloth sample, material optical parameter optimization to match the color, shine, and roughness of the sample, and normal map optimization to capture the fine, thread-level details. This visual property extraction process is completed in about 2 hours per image. This process is described in more detail by 2. The resulting final mesh and material create a very similar image to the sample when rendered.

**Vertex Optimization** The first step in reconstructing the mesh is to optimize vertex position. It is essential to perform this optimization before any visual material optimization. If the mesh was not first deformed to match the sample, any optimization of the normal map would cause structural features, such as folds and shadows, to appear in the normal map. Additionally, proper deformation is necessary to have highlights in the correct location when determining shine and color.

Our goal in this step is to find the vertex positions that minimize the loss between the rendered image and the sample image we wish to reconstruct. The formulation of the loss and optimization objective are presented equations 1 and 2. We initialize the vertex optimization with a 64 by 64 2D plane mesh and position a camera and light source facing the center of the plane using Redner's automatic camera positioning. We then render the image with path tracing and compute the loss as seen in 1 between the rendered image and the sample image. The differentiable renderer allows for differentiation with respect to vertex position of this loss, which we use with gradient descent to produce optimized vertex positions.

$$\mathbf{L_{image}} = \mathbf{\Sigma_{i,j}(I_{i,j}Y_{i,j})^2} \tag{1}$$

$$\arg\min_{\mathbf{V}} \mathbf{L_{image}} \tag{2}$$

$$\arg\min_{\mathbf{C,S,R}} \mathbf{L_{image}} \tag{3}$$

$$\arg\min_{\mathbf{N}} \mathbf{L_{image}} \tag{4}$$

An issue with this process is that it does not consider the physical constraints that a piece of cloth has. For example, a cloth cannot intersect itself, and the distance between two points on a cloth cannot stretch past a certain limit. For this reason, single-view reconstruction is very difficult and does not generally give desirable results. In their paper, Baptiste et al. address this issue by using preconditioned gradient descent to bias gradient steps towards smooth solutions [11]. Adopting this technique, we are able to accurately reconstruct the cloth's deformation.

**Color, Shine, and Roughness Optimization** Following the reconstruction of the cloth sample as a 3D mesh, a similar optimization is done with respect to a few of the mesh material properties: color, specular reflection, and roughness. For color and specular reflection, we choose one RGB value for each to represent the entire cloth (as opposed to having multiple colors or levels of shine across the cloth). Roughness is also represented by a single value that is constant across the cloth. We use the same process as vertex optimization with the new objective 3 to optimize these values with the objective together using the differentiable renderer.

**Normal Map Optimization** A normal map defines the direction of the cloth surface in a way that is only considered during lighting. In application, they are used to add detail to the surface of a mesh. We use the normal maps to capture the finer details that are too small and difficult for a mesh, especially a mesh of our dimensions, to capture. We expect the normal map to primarily show the individual threads where they are visible. Our analysis of a normal map versus the image itself allows for cloth deformation invariant results. Depending on how a cloth is deformed, the appearance of the threads in the image itself may appear bent, rotated in many directions, or have differing relative scale. The normal map is able to show a clear thread mapping as if the fabric has been un-deformed. This enables the use of Fourier analysis on the repeating thread structure. To extract this map, we follow the same optimization process as with the other visual parameters. We use the mesh that is output from the color, shine, and roughness optimization and again use the image loss with gradient descent to produce the optimized normal map. The results of all three optimization steps are shown in Table 1.

### 3.3 Normal Map Analysis

Cloth comes in two main structural categories, woven, and knitted. Woven cloth has two distinct thread directions forming a lattice. Knitted cloth is knotted together in columns, forming a regular, but non-lattice structure. The normal map of the inverse renderings samples the underlying thread structure in the z-direction, since the normals of the cloth surface oscillate with the weave or knit pattern. These periodic structures show up in the 2D Fourier transform of the normal map as distinct peak patterns. In the case of woven cloth the lattice structure creates two distinct peaks, offset orthogonal to each other, and mirrored along the $x = -y$ axis due to the mirroring of real fields by the Fourier transform. For knitted cloth the picture is less clear, we still expect to see frequencies corresponding to the knit columns, but the relationship between peaks will lack the orthogonality of woven fabrics in general. Our peak finding approach searches for these peaks by performing peak finding in the 2D Fourier spectrum.

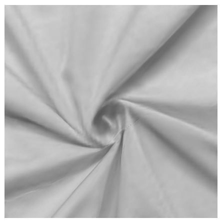
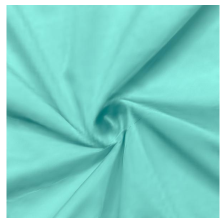
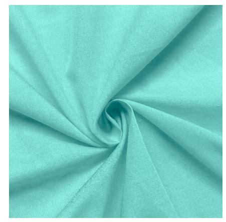
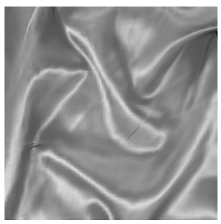
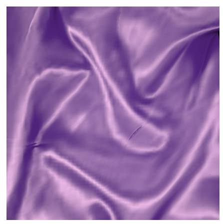
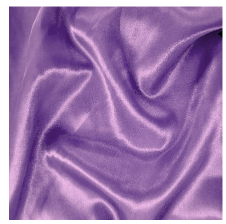
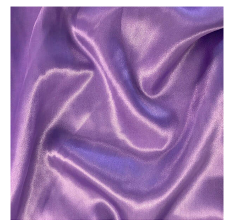
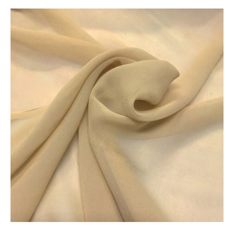
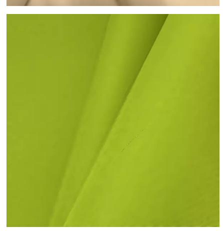
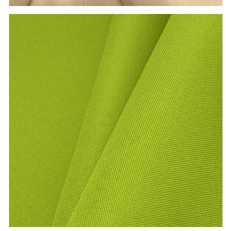
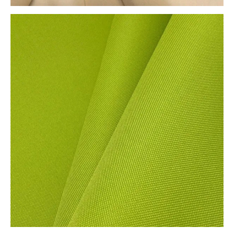
| | Vertex Optimization | Color, Shine, Roughness Optimization | Normal Map Optimization | Original Sample |
|---|---|---|---|---|
| Broad Cloth |  |  |  |  |
| Satin |  |  |  |  |
| Chiffon |  |  |  |  |
| Canvas |  |  |  |  |

Table 1: **Examples of the inverse rendering process on different fabric samples**. For each of them, we show the result after each step in the inverse rendering process. The vertex optimization deforms the mesh to match the geometry of the cloth seen in the image. This mesh is passed to the material parameter optimization, which adjusts the color, shininess, and roughness of the mesh material to recreate the visual properties of the sample image. This mesh and material are then passed to the normal map optimization, which is able to shape the normal map to show the thread-level details of the cloth sample. See Fig. 2 on the details shown for the surface of Canvas fabric using normal map optimization. This step reconstructs the fine details of the texture and allows for further signal analysis due to the periodic nature of the thread pattern.

(a) Reconstructed cloth after visual material property optimization

(b) Reconstructed cloth after normal map optimization.

(c) The learned normal map for this cloth sample

Figure 2: **Importance of the normal map**. In (a), the reconstructed mesh with optimized shininess, roughness, and color show none of the underlying thread structure of the cloth. In (b), after normal map optimization, the details of the thread count, thread density, thread size, and weave pattern are very clear. The learned normal map in (c) contains all of this thread information in a deformation invariant form. This allows us to analyze the repeating thread pattern for any cloth sample image.

---

**Algorithm 1** Inverse Rendering Visual Feature Extraction

---

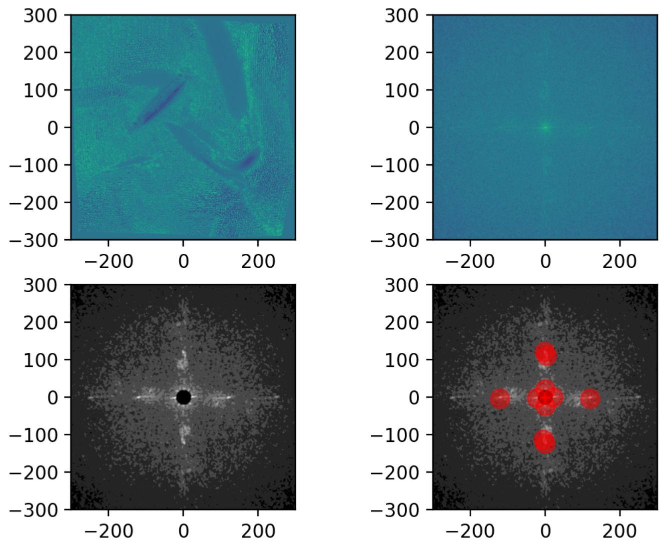
$\mathbf{O}$ ←2D Plane Object
$\mathbf{Y}$ ←RGB Sample Image

$\mathbf{U}$ ← Large Steps initialization
**for** i **do** in 0, 1, ... n
  $\mathbf{I}$ ← Render($\mathbf{O}$)
  $\mathbf{L} \leftarrow (\mathbf{I} - \mathbf{Y})^{\mathbf{2}}$
  $\mathbf{U} \leftarrow \mathbf{U} + \alpha\nabla_{\mathbf{UL}}$
  $\mathbf{V}$ ← Large Steps transformation of $\mathbf{U}$
**end for**

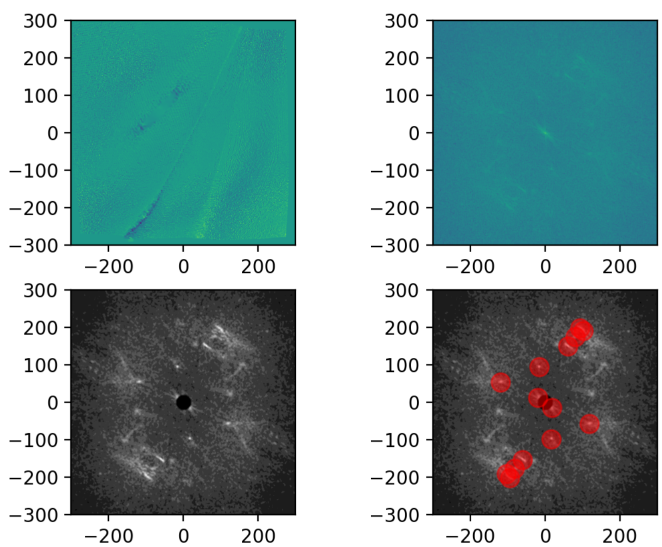
**for** i **do** in 0, 1, ... n
  $\mathbf{I}$ ← Render($\mathbf{O}$)
  $\mathbf{LL} \leftarrow (\mathbf{I} - \mathbf{Y})^{\mathbf{2}}$
  $\mathbf{C} \leftarrow \mathbf{C} + \alpha\nabla_{\mathbf{C}}\mathbf{L}$
  $\mathbf{S} \leftarrow \mathbf{S} + \alpha\nabla_{\mathbf{S}}\mathbf{L}$
  $\mathbf{R} \leftarrow \mathbf{R} + \alpha\nabla_{\mathbf{R}}\mathbf{L}$
**end for**

$\mathbf{N}$ ← initial normal map
**for** i **do** in 0, 1, ... n
  $\mathbf{I}$ ← Render($\mathbf{O}$)
  $\mathbf{L} \leftarrow (\mathbf{I} - \mathbf{Y})^{\mathbf{2}}$
  $\mathbf{N} \leftarrow \mathbf{N} + \alpha\nabla_{\mathbf{N}}\mathbf{L}$
**end for**
return $\mathbf{M}, \mathbf{C}, \mathbf{S}, \mathbf{R}, \mathbf{N}$

---




(a) Normal map processing sequence for Gauze Cotton




(b) Normal map processing sequence for Canvas

Figure 3: **Normal map analysis pipline**: (a) Gauze Cotton, and (b) Canvas. The figures have the z component of the normal map in the top left, the 2D Fourier transform in the top right in log scale, the processed normal map after DC removal, and mean filtering, the bottom right shows the results of peak finding before rotational invariance transformations, and normalization. The peaks in (a) show the characteristic orthogonally offset frequency peaks expected from axis aligned woven cloth. The peaks in (b) show noisier peaks generated from a non-axis aligned sample of canvas. The rotation, and noise issues are addressed by Algorithm 2.

**Normal Map Analysis Camera Orientation Invariance** Rotations and translations of the camera introduce rotation, scaling artifacts in the spectrum of the fabric as seen in figure 2b. To address rotational artifacts we convert from Cartesian frequency coordinates to polar, and discard the $\theta$ component. To address scale artifacts we normalize the $r$ component.

**Normal Map Analysis Noise Compensation** Noise in the Fourier spectrum comes from three main sources: Large scale ripples left in the normal map as artifacts from the cloth distortion, Edges of the normal map optimization region beyond which loss was never propagated, and aliasing caused by the resolution of the normal map being too low for the cloth type. The first category of noise is handled by simply zeroing all frequencies in a circle of fixed radius in the center of the cloth. The second type of noise is addressed by applying a mean kernel at a pre-guessed peak size to the spectrum, then selecting peaks a fixed number of standard deviations above the noise floor. During peak selection we track the radial distance of peaks, and the height of each peak. Having removed the zero center of the spectrum, we expect real peaks to be more likely in the middling frequency range which remains, so we sort peaks by radial distance, then by height, and select peaks in sorted order.

### 3.4 Classifier Feature and Model Selection

Features for the classifier were Roughness, $||\texttt{Specular Reflectance Vector}||$, $R_s$..., $Z_s$..., $N_{peaks}$. Where $R_s$ is the list of r coordinates for the selected peaks, and $Z_s$ is the normalized height of each peak, and $N_{peaks}$ is the total number of peaks that the peak finding algorithm selected. The L2 norm of specular reflectance was chosen over using each component independently for color invariance. QDA was chosen for our model since it functions well on low-data, and handles correlated feature data, unlike other low-data methods like Naive Bayseian models. We expect correlation in our features because our shininess and roughness, as well as peak radii will be pairwise correlated within each image class. The supplementary material includes an analysis of feature correlation properties.

**Algorithm 2**

Normal Map Analysis Pipeline

$\mathbf{N} \leftarrow$NormalMap
$\mathbf{Z} \leftarrow \mathbf{N} \cdot [\mathbf{0}, \mathbf{0}, \mathbf{1}]$
$\mathbf{F} \leftarrow$FFT2D with DC centered of $Z$
$\mathbf{F} \leftarrow \mathbf{F}$ with center DC peak removed
$\mathbf{F} \leftarrow \mathbf{F}$ mean filtered at peak size
$P \leftarrow$ Peaks of $\mathbf{F}$ at $\alpha$ standard deviations above noise floor
$S \leftarrow$ set of $(r, z)$ pairs
**for** p **do** Peaks
    $r, \theta \leftarrow$ polar coordinates of $p$ with $r$ integer truncated
    $z \leftarrow$ height of peak at $r, \theta$
    $S \leftarrow r, z$ if $(r, z) \notin S$, or $z > S[(r, z)] \cdot z$
**end for**
$R_m \leftarrow$ max radius in $S$
$O \leftarrow S$ sorted by $z$, then by $r$ distance normalized by $R_m$
return $O$

## 4 Experiments

### 4.1 Classification Performance

For small-dataset classifier experiments we use the popular method of Leave One Out Cross Validation. We apply this scheme to our data set of 30 multi-color cloth samples selected across 5 different cloth categories with 6 different cloth colors per category. Each iteration of the cross validation scheme trains a QDA classifier while leaving out one of the 6 example images from a category, then classifies the image into one of the 6 available categories. The QDA classifier achieves 60% accuracy with the peak finding configured at $n = 2$ peaks, $\alpha = 2$ standard deviations, DC radius = 20 spatial frequency, peak size = 15 spatial frequency.

Figure 4: simulated instances of broadcloth, satin, canvas, chiffon dresses simulated with cloth category parameters

### 4.2 Results

Our QDA model achieves a LOOCV accuracy of 60% on a data set of 30 images, with 5 classes. this represents a slight decrease in reconstruction accuracy compared to [12] who achieved 66.7% accuracy on a larger dataset and class range. However, their method requires video of cloth in a constrained environment, followed by simulation to expand the data available to their classification pipleine , and the classes they are classifying into do not represent real world cloth categories, but instead a parameter set specific to the [3] Arcsim cloth simulator. By contrast our method requires only easily obtainable single view images, and can generalize to any set of physics paramters for which there is a database indexed by farbic type, as available in [2]. Additionally our rotation, color, and scale invariance allow images to be taken at any angle, and any zoom level where the thread level detail is visible. This practicality enables easy extension, and use of our model.

The supplementary materials provide additional images and animation results.

## 5 Conclusion

In summary, we have presented a pipeline for robustly extracting and identifying cloth material parameters (for both rendering and simulation) from single view images using a differentiable rendering pipeline and digital signal processing techniques on normal maps. The pipeline is capable of operating on very few training samples and produces reasonably good accuracy. This method uses a small amount of data easily acquirable via commodity smartphones and is orthogonal to other techniques, such as differentiable physics optimization. As such, this method could be used as the initialization for a larger and more complex learning pipeline, while at the same time offering good approximation solutions for most common scenarios where there is a limited amount of time and image data from commodity devices and fashion catalog.

**Limitations** Our method is limited to work on single colored cloth, as its impossible for our single-view reconstruction algorithm to differentiate between a color change, and a surface feature or shadow. Further our classification scheme only allows for categorical cloth correctness, this is likely sufficient for entertainment domains in which high precision is not required, however for domains in which the physics parameters of a specific piece of cloth are required like cloth control, our method cannot account for deviations from a standard cloth of that material.

**Generalization and Future Work:** We speculate that this work will provide the initial step toward a more integrated learning framework that couples visual-physical learning for parameter identification using limited data, like single-view images most widely available. A fully integrated differentiable framework can offer a more automated pipeline for real-world reconstruction, including the dynamics, though such reconstruction would likely also require more extensive and limited data (e.g. multi-view videos) that would be difficult to capture and not widely available through commodity devices.

## References


[1] K. S. Bhat, C. D. Twigg, J. K. Hodgins, P. Khosla, Z. Popovic, and S. M. Seitz. Estimating cloth simulation parameters from video. 2003.

[2] D. Clyde, J. Teran, and R. Tamstorf. Modeling and data-driven parameter estimation for woven fabrics. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 1–11, 2017.

[3] W. Koh, R. Narain, and J. F. O'Brien. View-dependent adaptive cloth simulation. 2014.

[4] T.-M. Li, M. Aittala, F. Durand, and J. Lehtinen. Differentiable monte carlo ray tracing through edge sampling. *ACM Trans. Graph. (Proc. SIGGRAPH Asia)*, 37(6):222:1–222:11, 2018.

[5] J. Liang, M. Lin, and V. Koltun. Differentiable cloth simulation for inverse problems. *Advances in Neural Information Processing Systems*, 32, 2019.

[6] S. Liu, T. Li, W. Chen, and H. Li. Soft rasterizer: A differentiable renderer for image-based 3d reasoning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7708–7717, 2019.

[7] Z. Montazeri. *Modeling and Rendering Cloth*. University of California, Irvine, 2021.

[8] Z. Montazeri, S. Gammelmark, H. W. Jensen, and S. Zhao. A practical ply-based appearance modeling for knitted fabrics. *arXiv preprint arXiv:2105.02475*, 2021.

[9] Z. Montazeri, S. B. Gammelmark, S. Zhao, and H. W. Jensen. A practical ply-based appearance model of woven fabrics. *ACM Transactions on Graphics (TOG)*, 39(6):1–13, 2020.

[10] J. K. Murthy, M. Macklin, F. Golemo, V. Voleti, L. Petrini, M. Weiss, B. Considine, J. Parent-Lévesque, K. Xie, K. Erleben, et al. gradsim: Differentiable simulation for system identification and visuomotor control. In *International Conference on Learning Representations*, 2020.

[11] B. Nicolet, A. Jacobson, and W. Jakob. Large steps in inverse rendering of geometry. *ACM Transactions on Graphics (TOG)*, 40(6):1–13, 2021.

[12] S. Yang, J. Liang, and M. C. Lin. Learning-based cloth material recovery from video. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4383–4393, 2017.

[13] S. Zhao. Micro-appearance modeling of fabrics. In *Mathematical Insights into Advanced Computer Graphics Techniques*, pages 7–33. Springer, 2016.

[14] S. Zhao, F. Luan, and K. Bala. Fitting procedural yarn models for realistic cloth rendering. *ACM Transactions on Graphics (TOG)*, 35(4):1–11, 2016.

# A Appendix

## A.1 Cloth Variety

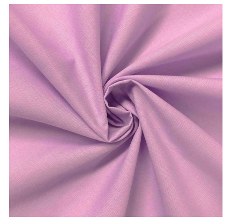
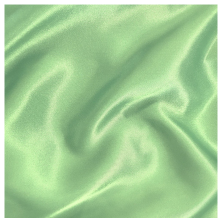
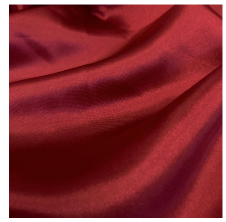
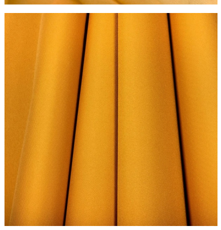
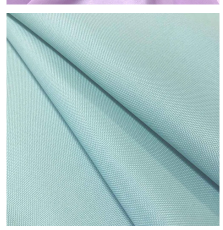
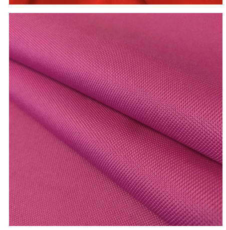
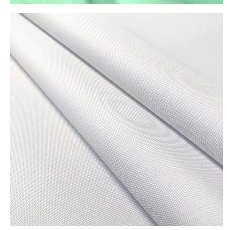
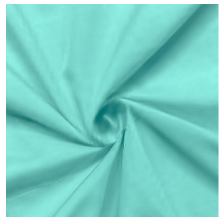
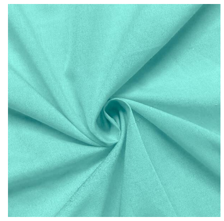
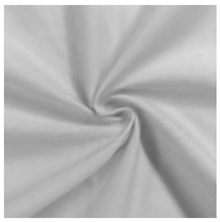
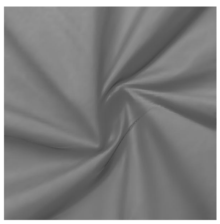
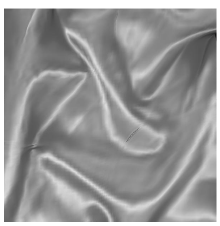
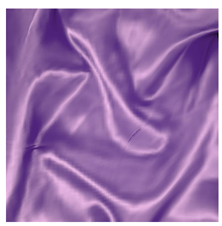
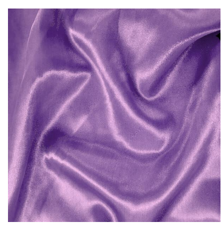
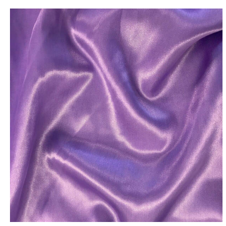
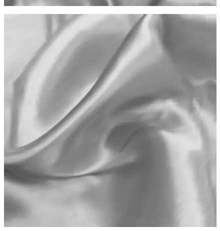
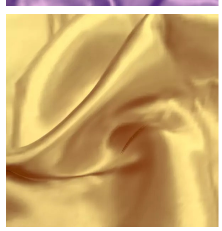
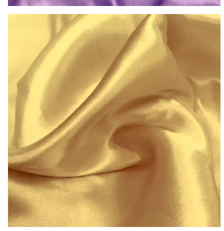
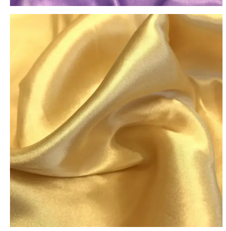
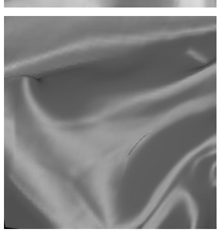
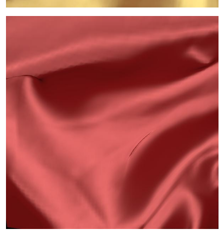
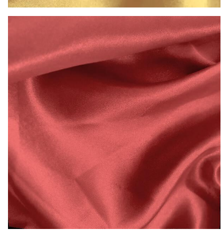
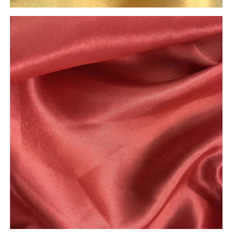
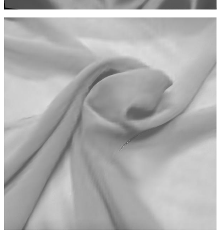
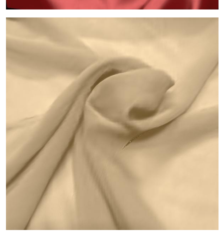
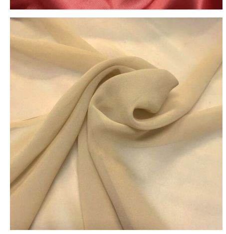
| | | | | |
|---|---|---|---|---|
| Broad Cloth |  |  |  |  |
| Satin |  |  |  |  |
| Chiffon |  |  |  |  |
| Canvas |  |  |  |  |

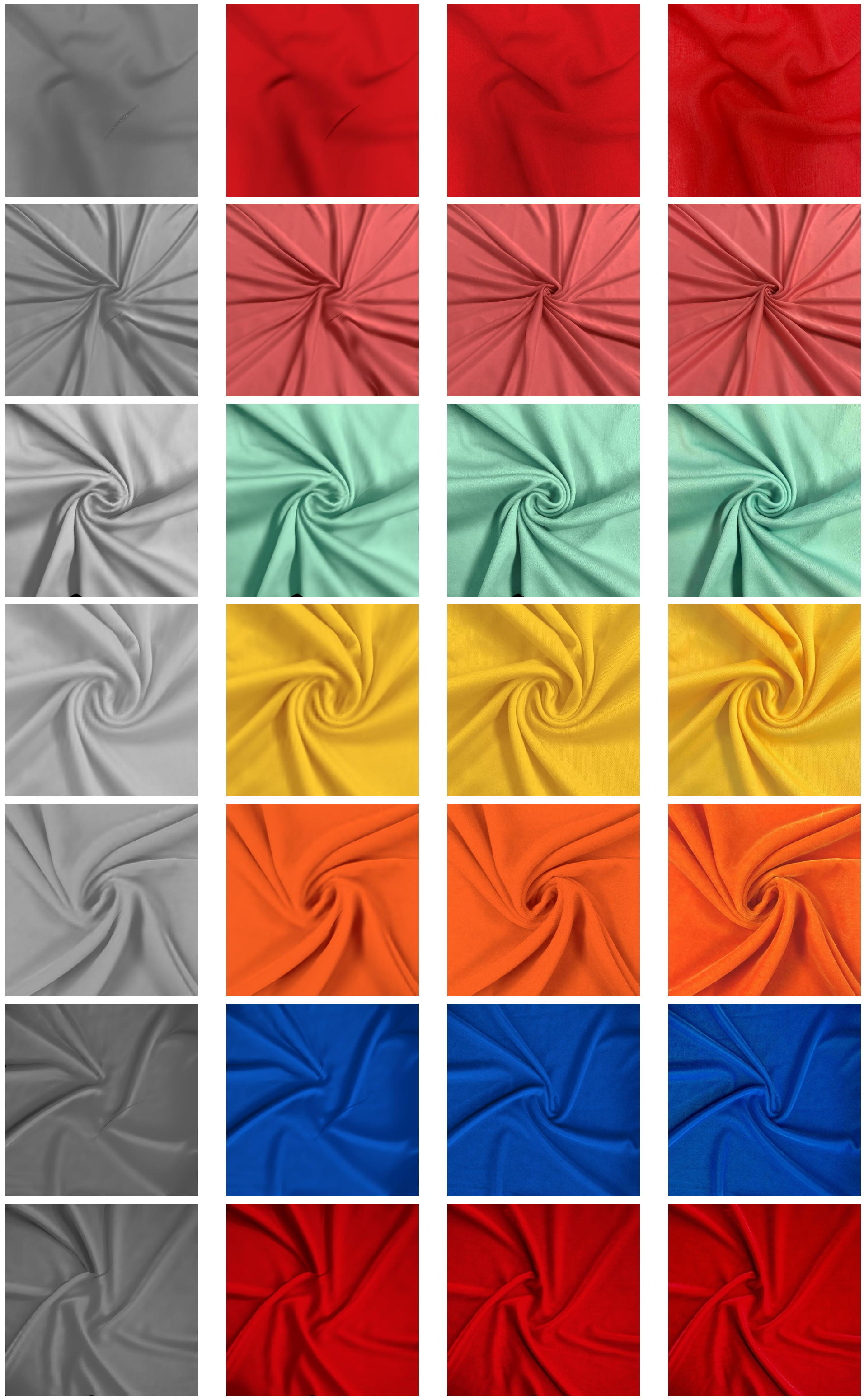
Table 2: **A sample of the image data set used for training and testing, note the wide variety in cloth deformations and colors.**

### A.2 Extended Cloth Reconstruction Table

| | Vertex Optimization | Color, Shine, Roughness Optimization | Normal Map Optimization | Original Sample |
|---|---|---|---|---|
| Broad Cloth |  |  |  |  |
| |  |  |  |  |
| |  |  |  |  |
| Satin |  |  |  |  |
| |  |  |  |  |
| |  |  |  |  |
| Chiffon |  |  |  |  |

Canvas

Gauze



Interlock

Velvet

Table 3: **Examples of the inverse rendering process on different fabric samples**. For each, we show the result after each step in the process. The vertex optimization deforms the mesh to match the geometry of the cloth seen in the image. This mesh is then passed to the material parameter optimization, which adjusts color, shininess, and roughness of the fabric material to recreate the visual properties of the sample image. This mesh and material are then passed to the normal map optimization, which shapes the normal map to reflect the thread-level details of the cloth sample.

## A.3 Feature Correlation Analysis


Figure 5: **Normal map analysis pipline**: Covariance matrices for Broadcloth (top) and CottonGauze (bottom) categories. Note the strong off-diagonal components indicating correlations between z0 and z1, r0 and r1, and Shininess and Roughness. These correlations provide the basis for our choice of Quadratic Discriminant Analysis as they violate the naive Bayes assumption

### A.4 Experiments on expanded dataset including knit and velvet samples

We perform experiments adding two new cloth categories, including non-woven cloths to assess the performance of our method outside woven fabrics. This dataset also includes Velvet, which poses a unique challenge due to its non-periodic surface. With these cloth types added, we achieve a LOOCV accuracy of 38% on 7 cloth categories with hyper parameters configured at $n = 2$ peaks, $\alpha = 3$ standard deviations, DC radius = 20 spatial frequency, peak size = 25 spatial frequency. If we remove velvet, leaving us with 6 categories including woven and knitted fabrics, we achieve 58.3% accuracy with $n = 4$ peaks, $\alpha = 3$ standard deviations, DC radius = 20 spatial frequency, peak size = 15 spatial frequency.