Surface estimation of translucent materials: an application to fabric digitization

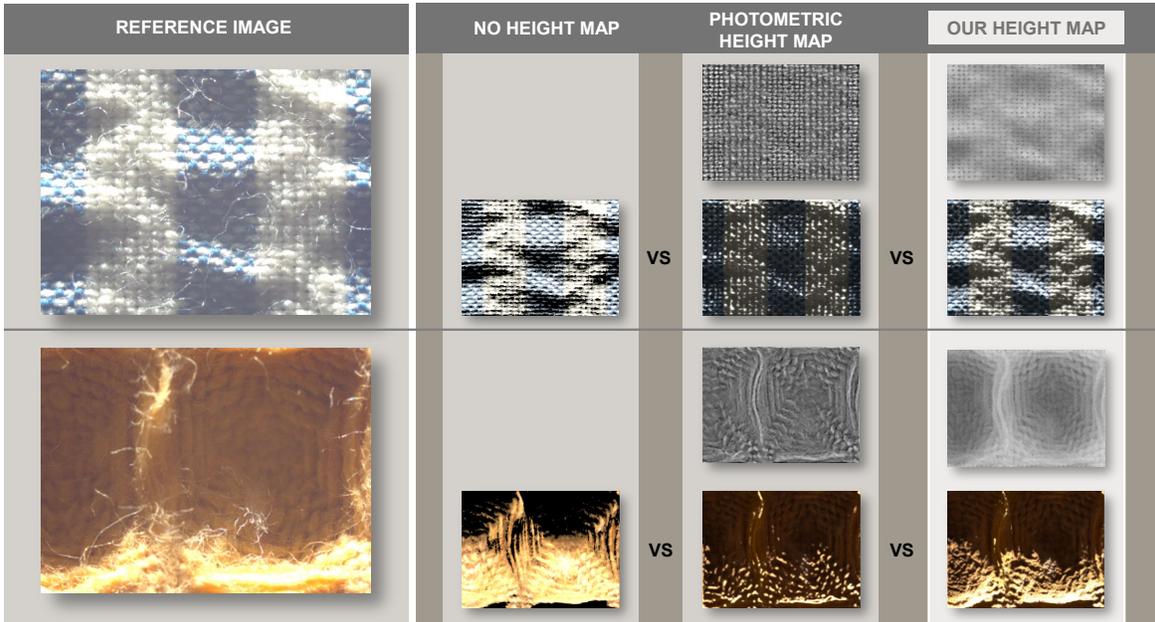Diego Sagredo , Javier Fabre , and Jorge Lopez-Moreno

Fig. 1. Preview of the resulting surface estimation, as well as raster renders employing the obtained height maps, compared to the same renders with no height map and with photometric height maps (from integrated photometric normals).

Abstract—**Estimating the surface of translucent objects from photometric data poses significant challenges due to complex internal light scattering. We introduce a novel method that computes a depth map from single-viewpoint photographs of a material sample, captured under multiple illuminations. Our approach leverages inverse rendering to derive a volumetric representation, including density, albedo, and phase function, from which a surface mesh is reconstructed. Beyond validation with synthetic and 3D-printed physical models, we illustrate our technique's power by successfully applying it to the digitization of fabrics, a notoriously difficult material due to its intricate translucent structure. This work advances the state-of-the-art texture stack acquisition via enhanced surface reconstruction.**

Index Terms—**Differentiable rendering, volumetric, height map**

I. Introduction

Diego Sagredo is with Universidad Rey Juan Carlos (Madrid, Spain). E-mail: diego.sagredo@urjc.es

Javier Fabre is with Seddi Inc (USA) and Universidad Rey Juan Carlos (Madrid, Spain). E-mail: javier.fabre@urjc.es

Jorge Lopez-Moreno is with Universidad Rey Juan Carlos (Madrid, Spain). E-mail: jorge.lopez@urjc.es

DIGITIZING real-world materials is a fundamental topic in computer graphics, particularly for applications involving the digitization of everyday objects such as garments or human skin, where we have very low perceptual tolerance for visual artifacts. Translucent materials pose a unique difficulty due to their complex interaction with the illumination: the light scattering complicates the detection of defined surface geometry features for inverse methods. Accurately capturing the geometry of such materials is critical not only for visual realism, but also for improving downstream tasks like reflectance estimation or relighting.

Traditional geometry acquisition methods often rely on multi-view stereo or depth sensors, which fail in the presence of translucency. Even multi-view photometric techniques, which estimate surface normals, struggle to resolve depth ambiguities without explicit geometry constraints. Moreover, learning-based monocular depth estimation methods, while relatively successful for opaque surfaces, are limited in their generalization to translucent surfaces.

In this work, we propose a novel method to recover surface

depth in the form of height (depth) maps from a single zenithal camera view, using multiple directional lighting conditions. Our approach leverages Physically-Based Differentiable Rendering (PBDR) to optimize a volumetric density field to match the input images, enabling accurate estimation of relative surface geometry, even for challenging cloth samples.

We evaluate our method both synthetically and on real captured data, and demonstrate its effectiveness when integrated into an existing material digitization pipeline. We further investigate the impact of specific design decisions and analyze common failure modes to inform future improvements.

Our contributions are:

- A volumetric optimization method for depth estimation of translucent cloth using differentiable rendering.
- A novel integration of this method into a photometric material capture pipeline, adding support for shadowing and explicit geometry.
- An analysis of depth extraction techniques and inpainting strategies to improve robustness in challenging materials.

II. RELATED WORK

Materials estimation: Estimating geometric and optical properties of real-world materials is a well-established research area in computer graphics and computer vision. Significant research efforts have primarily targeted recovering surface material parameters using methods based in BRDF models [1]–[3], SVBRDF models [4]–[6], and SVBSDF models [7]–[9]. Complementary methods rely on intrinsic image decomposition to separate reflectance and shading components [10].

Comprehensive surveys [11], [12] provide further insight into the current state of material acquisition and inverse rendering, highlighting existing methods and their limitations, setting a clear context for our contribution.

Geometric parameters estimation: Many methods specifically target geometric attributes like albedo, roughness, and normal maps [1], [2], [4]. In contrast, relatively few explicitly recover depth maps. Early examples include the dark-is-deep approximation [13] and learning-based monocular depth estimation [6], [14], predominantly for opaque materials. Some works [15] explore the use of Depth from Focus [16] techniques for fabric captured at the microscale, although it is not accurate enough for this context. Recent research addressing translucent materials often assumes homogeneous subsurface scattering [8], leaving depth estimation in heterogeneous translucent materials largely unexplored.

Scene-level depth estimation using monocular RGB or depth sensors, such as LiDAR (Light Detection and Ranging) has seen notable advancements [17]–[19]. However, such approaches are typically insufficient for capturing detailed depth from translucent objects due to limitations in resolution, precision, or handling of subsurface scattering.

We address these limitations by optimizing detailed height maps for heterogeneous translucent materials, such as cloths.

Inverse rendering and PBDR: The rapid evolution of differentiable rendering has significantly expanded inverse rendering capabilities, enabling gradient-based optimization of sophisticated rendering processes. Surveys [20], [21] detail foundational developments in this field.

Our work leverages PBDR [21], which explicitly differentiates through comprehensive light transport simulations, encompassing complex phenomena such as subsurface scattering and global illumination. Several frameworks support PBDR, including Redner [22], nvdiffrec [23], and Mitsuba [24]. Among them, Mitsuba 3 stands out by combining fully differentiable path tracing, extensive volumetric rendering via voxel grids, and robust material modeling. These capabilities are powered by Dr.Jit [25], a differentiable just-in-time compiler and mathematical framework tailored for high-performance rendering and optimization. Recent advances such as Path Replay Backpropagation (PRB) [26] have significantly accelerated differentiable rendering in Mitsuba via efficient, unbiased gradient computation through complex light paths.

Additionally, some methods [27] explore micro-appearance modeling for real fabrics, which relates closely to our samples and further underscores the applicability of differentiable rendering techniques to cloth-like translucent materials.

In this work, we exploit the advanced PBDR capabilities of Mitsuba and Dr.Jit to achieve robust and accurate depth estimation of translucent materials through joint optimization of volumetric density, albedo, and phase function parameters.

Volumetric inverse rendering: Inverse rendering techniques have increasingly included volumetric representations, enabling the recovery of internal material properties such as density and scattering parameters. Early works, such as inverse volume rendering with material dictionaries [28], demonstrated feasibility using predefined basis representations but lack the differentiability and resolution of modern techniques. Recent works [26] use Mitsuba 2 and achieve effective reconstruction of heterogeneous volumetric parameters through differentiable path tracing. Additional works [29] present a learning-based method for inverse subsurface scattering, capable of inferring scattering profiles from single images. Other methods [30] further optimize appearance in 3D printed translucent volumes using gradient-based approaches, closely aligning with our goals. Inverse scattering methods have been comprehensively evaluated [9], with particular emphasis on the complexities and challenges associated with heterogeneous media. Lastly, jointly estimating surface textures and homogeneous subsurface scattering properties was practiced [31] using an inverse rendering framework [24], [25]. Their method assumes a spatially uniform volume and known geometry, and therefore focuses solely on material reconstruction.

Volumetric cloth modeling and appearance: Volumetric representations have long been recognized as an effective means of modeling the appearance of fibrous and cloth-like materials, whose complex three-dimensional micro-structure and multiple scattering effects are poorly captured by surface-based models. In the past decade, a general radiative transfer framework for anisotropic participating media was introduced [32], enabling physically based forward rendering of materials with oriented micro-structure, including cloth. This formulation established volumetric participating media as a viable

forward model for cloth appearance.

Building on this paradigm, a statistical volumetric approach to predictive rendering of fabrics was proposed [33], approximating yarn-level structure using voxelized density and orientation distributions. Their method demonstrates high-fidelity forward appearance reproduction while avoiding explicit fiber geometry, but assumes known or synthesized volumetric structure and does not address inverse reconstruction.

Closer in spirit to inverse problems, some works [34] use micro-CT imaging to construct volumetric appearance models of fabric. In their approach, volumetric density and fiber orientation fields are extracted from CT scans, while global scattering parameters are estimated by matching image statistics between photographs and rendered images. Although this method solves an inverse problem at the level of appearance parameter estimation, it assumes access to volumetric ground-truth geometry and does not attempt to recover geometric structure from images alone.

Our work advances the state-of-the-art by simultaneously optimizing volumetric density, albedo, and phase function, explicitly deriving detailed and practical height maps for heterogeneous translucent materials.

III. DEPTH ESTIMATION METHOD

Estimating the geometry of translucent materials, such as cloth, using a relatively simple capture setup presents significant challenges, which arise from the inherent ambiguity of interpreting geometry under directional lights, where a single illumination pattern can correspond to multiple plausible surface shapes. Our goal is to recover surface depth, represented as height maps, from a limited set of photographic captures: a single frontal (top-down) camera view under several directional lighting conditions. This sparse capture setup we used [7], provides a series of images from an overhead perspective along with a stack of texture maps generated via differentiable rendering. Among these, we primarily use the albedo map, which contains baked-in ambient occlusion due to the absence of explicit geometry during its derivation. Their texture stack optimization system is also used for evaluation of our method, as described in Section III-B

Depth estimation from 2D images is inherently ill-posed. Conventional methods typically require multiple views or complex lighting arrangements to resolve geometric ambiguities. The challenge is even greater for translucent materials, where light scattering and soft shadowing further increase ambiguity, as structural cues are obscured. Although some machine learning approaches attempt single-image depth estimation, their success is largely limited to opaque materials and constrained scene types, with lower performance for translucent materials.

To overcome these challenges, our method combines optimization with differentiable rendering to estimate depth from multiple images taken from a fixed viewpoint under varying lighting. We recreate the physical capture setup in a virtual scene, replicating the camera parameters and directional lights, and optimize a height map that aligns the rendered outputs with the observed captures. This allows us to reconstruct fine surface detail despite the sparsity of input data.

A. Volumetric parameters optimization

The optimization procedure employs differentiable rendering to backpropagate the derivatives of the volumetric density, albedo and phase function (Henyey-Greenstein g [35]) parameters, to iteratively optimize them by minimizing differences between renders and reference captures through a loss function. The boundaries of the volume are manually specified and not optimized, as Mitsuba does not support differentiable volume bounds as they define scene structure and acceleration data, preventing smooth updates during optimization.

The optimization is divided into three stages, where each stage has the goal of obtaining finer details than the previous one, like a pyramidal multiscale strategy. The exact structure of our method is presented in Figure 2.

The scene parameters are updated with its corresponding gradients at the end of each gradient descent iteration, after all the views have been rendered. These views vary based on differences in light direction, while the camera setup remains consistent across all views.

Following the rendering of each view, the loss is computed and subsequently back-propagated, accumulating gradients for the optimizer, so it can later take a step and update the scene parameters accordingly. The loss calculation employs a modified Mean Squared Error (MSE) objective function:

$$\hat{L}(y) = \log\left(W \cdot \frac{L(y)}{100} + 1\right) \cdot 100 \qquad (1)$$

$$T(y) = \text{LAB2RGB}\left(\hat{L}(y),\ A(y),\ B(y)\right) \qquad (2)$$

$$\mathscr{L} = \frac{1}{N}\sum_{i=1}^{N}\left(T(\hat{y}_i) - T(y_i)\right)^2 \qquad (3)$$

where $\hat{y}$ is the optimized image, y is the ground truth reference image and N is the number of pixels in both images.

Due to the fact that Henyey-Greenstein phase function is unable to accurately represent specular reflections, it becomes difficult to minimize the difference with the reference captures, which show pronounced specular highlights, resulting in significant spikes in the loss function, adversely affecting the optimization process.

Originally, we opted for modifying the loss function with a weight map, applied to the function as follows:

$$\mathscr{L} = \frac{1}{N}\sum_{i=1}^{N}\omega_i\left(\hat{y}_i - y_i\right)^2 \qquad (4)$$

The factor ω is a per-pixel weight map used to weigh down the brighter pixels of the image. Specifically, we classified pixel brightness based on luminosity (L channel) in the CIELAB color space [36] for the reference image $\hat{y}$, which effectively separates brightness from chromaticity. The main limitation of this strategy is that certain cloth samples exhibit a noticeable loss of chromaticity in the final renderings, which may have a negative effect in the accuracy of height map optimization. This issue is primarily observed in materials with bright, saturated albedos, such as the light pink spectrum.
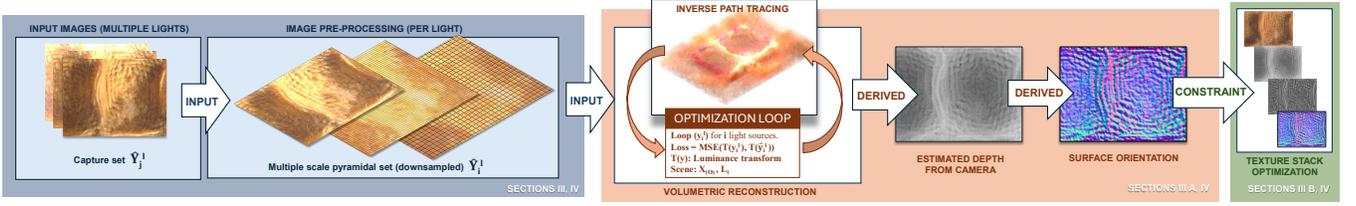
Fig. 2. As this diagram shows, our method consists of 3 main sections: First, we take the capture set and process it to obtain 3 different resolution scales, then we optimize the volume for the sample in 3 stages, one for each scale, acquiring our estimated surface, and finally we perform the texture stack optimization, with our estimated surface.

To overcome this limitation, we introduce the transformation $T(y)$, that operates in the perceptually uniform CIELAB color space. Given an input image y in RGB, we first convert it to LAB, to then apply a nonlinear transformation to the luminance channel $L(y)$ as seen in Equation 1, where $L(y)$ is first normalized to the range $[0, 1]$ and scaled by a weighting factor W, which we set to 1.1 empirically, to ensure the decay is adequate. Then, its value is renormalized to the original range.

Finally, we recombine $\hat{L}(y)$ with the original chromaticity channels $A(y)$ and $B(y)$, and convert the result back to RGB for further processing or output (Equation 2).

This transformation implicitly downweights the brightest pixels, reducing the impact of specular highlights and exposure imbalances, without adversely affecting chromaticity. Qualitatively, this adjustment improves the visual fidelity of the renderings in highly chromatic regions, as well as the height maps consistency with variations in the albedo.

The volumetric optimization yielded two output volumes: density and albedo. From the density volume, we extracted the desired height map, using a method we refer to as **Median-integrated Density**, inspired by NeRFMeshing [37]. More specifically, we defined the interface between empty and solid voxels as the depth at which cumulative voxel densities along a ray reach 50% of the total integrated density.

B. Application on material digitization pipeline

To evaluate the practical impact of our method, we integrated it into a material digitization pipeline [7], which captures appearance under controlled lighting and optimizes a stack of texture maps (e.g., albedo, roughness, normal, specular). Their system operates without explicitly modeling geometry, relying instead on a photometrically estimated normal map, which limits its ability to represent complex surface structure and cast shadows accurately.

We modified their rendering and optimization pipeline to support geometry-aware differentiable rendering. Specifically, we incorporated 3D geometry and shadow mapping capabilities, allowing the renderer to use height maps during texture optimization and shading. Also, we take inspiration from differentiable shadow mapping techniques [38], and make use of Nvdiffrast [39] to maintain differentiability throughout the entire shadow mapping process. Additionally, we soften the shadowmap appearance by applying a Gaussian blur kernel and compute an ambient term $I_{\text{ambient}} = k \cdot A^2$ to be added to the final image, where A is the albedo map and k is a
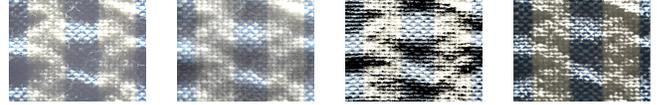

Fig. 3. From left to right: photographic reference capture, a render using the volumetric pipeline, and two renderings using a material digitization raster pipeline; first without geometry-aware differentiable rendering and second with geometry-aware differentiable rendering

constant scaling factor. A visual comparison is shown in Figure 3, where the inclusion of geometry clearly enhances shadow realism and surface structure.

Using the same reference captures provided by their system, which also serve as inputs to our differentiable optimization process, we evaluated the visual fidelity and consistency of material renderings with and without the integration of our height maps. The inclusion of our geometry data improved cast shadows and enhanced the perceived structure of the material, particularly in samples where the surface relief plays a significant role in appearance.

We further evaluated the role of surface orientation by replacing their photometrically estimated normal map with one derived from our height map using a Sobel-based [40] orientation calculation. Interestingly, while our depth-derived normal map appeared sharper and more detailed, it resulted in less realistic shading when used in their raster-based renderer. This is due to the lack of subsurface scattering simulation, which in practice softens surface transitions in real cloth. In this context, the blurred quality of their photometric normals, which implicitly captures that softness, proved to be more visually consistent with the reference captures. A comparison between the two normal maps is shown in Figure 4. Despite this, the combination of our geometry with their photometric normal map still yielded better results than the baseline configuration without explicit geometric information.

In addition to improving visual appearance, the introduced geometry support also enables the extraction of an ambient occlusion map. We approximate this by averaging shadow maps corresponding to the 30 lowest elevation lighting angles. This produces a plausible occlusion effect grounded in the optimized geometry, as it can be seen in Figure 5.

IV. IMPLEMENTATION DETAILS

Renderer setup: Our volumetric optimizations use the standard Path Replay Backpropagation implementation in Mitsuba, while the rasterizer is fully self-implemented.
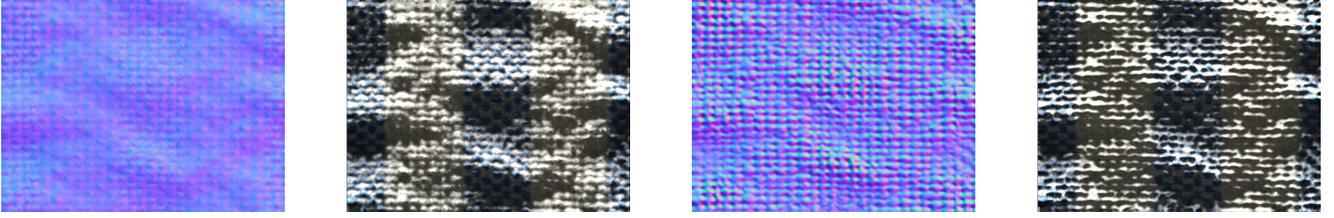
Fig. 4. Comparison between photometric normal map and its corresponding raster render (left two images) versus our depth-derived Sobel normal map and its raster render (right two images), both rendered with our recovered height map. To compute our normal map, we use a Sobel operator with a 17-pixel kernel and a scale factor derived from the kernel size to normalize the weights. The large kernel smooths gradients by attenuating high-frequency components in the height map, which can otherwise introduce visual artifacts in the resulting normal map.
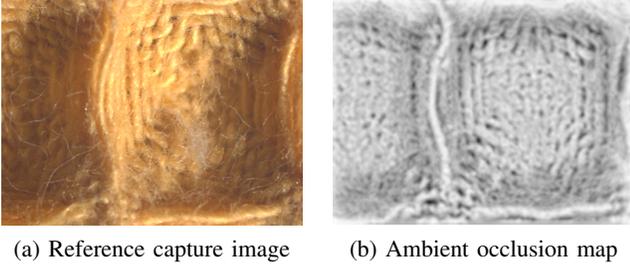

(a) Reference capture image　　　(b) Ambient occlusion map

Fig. 5. Ambient occlusion map, approximating far field illumination, derived from our surface estimation.

Ring	Captures set	Captures subset	Elevation
1	30	4	83°
2	28	2	71°
3	24	0	59°
4	21	0	47°
5	16	0	36°
6	11	2	24°
7	4	0	166° (below)

TABLE I
DISTRIBUTION OF THE FULL CAPTURE SET BY RING

Capture set: We obtained 135 captures per sample using the optical capture system [7]. Each image corresponds to a different directional lighting angle, arranged in a concentric ring formation around the sample. To reduce computational cost during optimization, we selected a representative subset of the captures through a small ablation study, which is described in the supplementary material. This subset maintains angular diversity while keeping iteration time manageable. The lighting distribution is shown in Table I.

Padding and boundary extension: The captured data only covers the rectangular patch of the sample, potentially missing geometric details casting shadows from outside the visible field-of-view, degrading optimization quality. To address this, we extended the optimization volume beyond the sample patch boundaries, by a factor of 2, along the X and Y axes, creating a padding region, and resulting in volume bounds that are twice the size of the area of interest. This prevented edge artifacts by ensuring that shadow casters geometry outside the field-of-view could be defined by the optimization.

Pyramidal upsampling stages: We optimize in three stages, from coarse to fine. The coarsest resolution has a minimum of 30, and the finest, a maximum of 360, to ensure we will not lose quality or run out of memory. The middle

resolution is derived from the known yarn radius from the capture setup, by resolving yarns to have approximately 5 pixels of diameter. Then, the first and third stages resolutions are adapted to the middle one, to ensure the scale factor is an integer, as per the requirements of volumetric upsampling in DrJIT [25]. The following equations implement this strategy:

$$Rx_{\mathrm{mid}} = \left\lfloor \frac{2.5}{r_{\mathrm{yarn}}} \cdot Rx_{\mathrm{ref}} + 0.5 \right\rfloor \tag{5}$$

$$\alpha_{\mathrm{top}} = \left\lfloor \frac{Rx_{\mathrm{max}}}{Rx_{\mathrm{mid}}} \right\rfloor, \qquad \alpha_{\mathrm{bot}} = \left\lfloor \frac{Rx_{\mathrm{mid}}}{Rx_{\mathrm{min}}} \right\rfloor \tag{6}$$

$$Rx_{\mathrm{top}} = \begin{cases} Rx_{\mathrm{max}}, & \text{if } \alpha_{\mathrm{top}} < 1 \\ \alpha_{\mathrm{top}} \cdot Rx_{\mathrm{mid}}, & \text{otherwise} \end{cases} \tag{7}$$

$$Rx_{\mathrm{bot}} = \begin{cases} Rx_{\mathrm{min}}, & \text{if } \alpha_{\mathrm{bot}} < 1 \\ \left\lfloor \frac{Rx_{\mathrm{mid}}}{\alpha_{\mathrm{bot}}} + 0.5 \right\rfloor, & \text{otherwise} \end{cases} \tag{8}$$

where r_{yarn} is the yarn radius (in pixels) and Rx_{ref} is the horizontal resolution of the reference images.

This formulation guarantees that resolutions for all three stages are consistent, while ensuring that each stage operates at an appropriate level of detail. For the resolution values along the Y-axis and Z-axis, we apply the known ratios with respect to the width, to assure that the voxels maintain a regular shape.

There is, however, one exception to this rule: for samples with unusually large physical depth relative to their width, we allow the voxels to become elongated along the Z-axis in order to reduce the total number of required voxels and control memory consumption during optimization. Rather than maintaining perfectly cubic voxels in all cases, we scale the voxel size along the depth direction to cap the volumetric resolution. This is done by enforcing a minimum aspect ratio, so if a sample's physical depth would otherwise require a highly granular Z-axis (leading to excessive memory usage), we instead compute the resolution as if the volume were shallower, preserving a manageable number of layers while allowing voxel stretching along Z.

Metallic masking: Some fabric samples contain metallic strands or flakes, which introduce strong specular reflections that interfere with the optimization process. These reflections result in brightness spikes that the differentiable renderer cannot accurately explain using the underlying scattering model. To address this, we identify and create a mask for the metallic
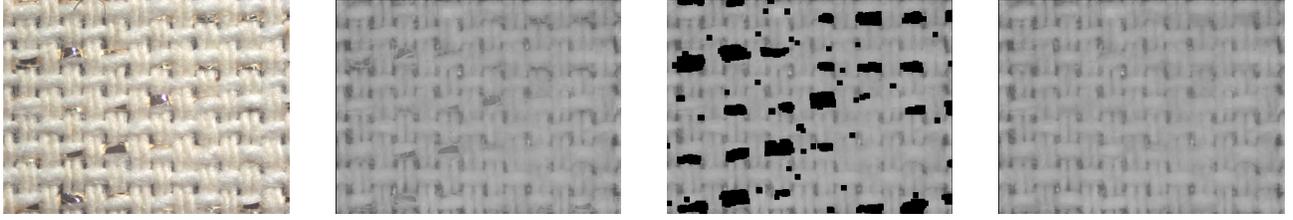
Fig. 6. Results of metallic masking. From left to right: Reference image, height map from optimization, height map with masked metallic regions and height map with inpainting applied to the metallic regions.
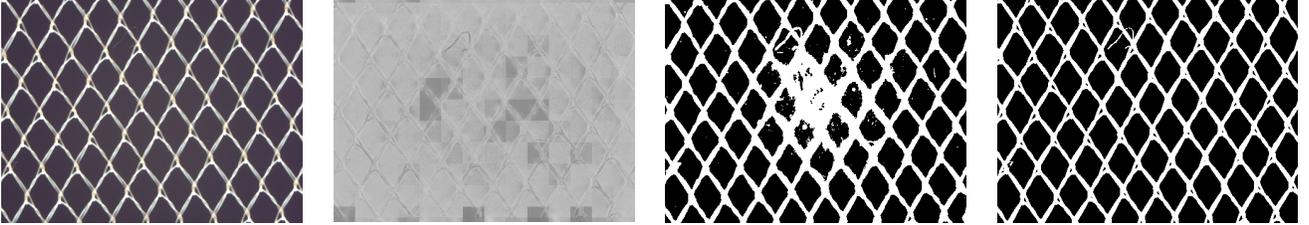

Fig. 7. Results of the opacity estimation refinement process. From left to right: Reference image, height map from optimization, original alpha map and refined alpha map.

regions in the estimated height map, informed by roughness and specular tint textures from the stack. Specifically, we compute the mask by inverting the roughness, adding the specular tint, applying a threshold after normalization (set to 0.75 empirically) and performing a morphological dilation to improve robustness. Finally, we apply an inpainting process on the unmasked regions [41] to fill in the missing geometry. The resulting height maps are more consistent and stable than the ones obtained from our estimation method. A visual comparison before and after masking is shown in Figure 6.

Opacity Refinement: Certain samples might include perforations or holes, which pose challenges during depth estimation due to the absence of observable geometry in those regions. To improve alpha (opacity) masking, we generate a refined binary alpha map based on the captured image stack. Specifically, we select the orthographic view (top-down light direction) from the set of captures and normalize it using the average albedo computed over solid regions of the albedo map. Let I be the selected capture and A the albedo map masked by the original alpha map. The normalized image I' is computed as:

$$I' = \text{ContrastEnhance}\left(\frac{I}{\mu_A + \varepsilon}\right) \tag{9}$$

where μ_A is the average color of valid albedo pixels and ε is a small constant to avoid division by zero. Contrast enhancement is applied using a fixed stretching factor (empirically set to 5.0). The resulting image is converted to grayscale and binarized using Otsu's thresholding method [42], a parameter-free technique that automatically selects the optimal threshold based on histogram variance. This produces a clean binary mask that robustly distinguishes perforations from valid geometry, and is used to mask the densities volume during optimization. A comparison between the original and refined alpha maps is shown in Figure 7.

Accumulated albedo map: In order to facilitate the visualization of the obtained albedo volume, we convert it into

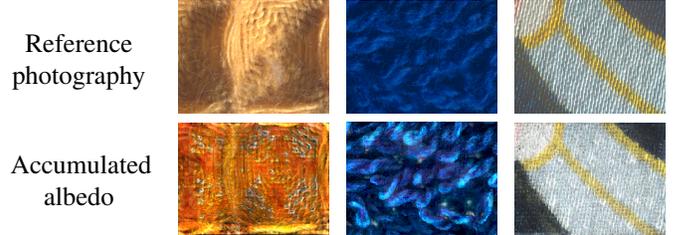Reference photography
Accumulated albedo

Fig. 8. Accumulated albedo maps compared to their reference capture images.

a texture map that represents the accumulated albedo. We accumulate each vertical column of voxels in the volume, while also taking into account the density values in order to obtain the final texture values.

First, let the transmittance at voxel i be defined as:

$$T_i = e^{-\sigma_i \cdot \Delta z} \tag{10}$$

where σ_i is the density at voxel i, and Δz is the voxel thickness, computed as:

$$\Delta z = \frac{d_{\text{cloth}}}{r_z} \tag{11}$$

Here, d_{cloth} denotes the physical depth of the entire cloth sample, and r_z is the resolution of the volume along the Z-axis.

The scattering term at each voxel is:

$$s_i = a_i \cdot \sigma_i \tag{12}$$

with a_i representing the albedo at voxel i. The final accumulated value S for a column is:

$$S = \sum_{i=0}^{N} (1 - T_i) \cdot s_i \tag{13}$$

The maps obtained from this, can be observed in Figure 8.

V. RESULTS AND EVALUATION

In this section, we evaluate the performance and robustness of our height map optimization method across synthetic validation scenes, prior methods, real material captures, and multiple implementation variations. We begin by validating our estimation method using controlled synthetic scenes with known geometry, and comparing the normal map derived from the resulting height map of our method with the results of prior methods that recover SVBRDF normal map. Then, we display an ablation study on key design choices, followed by the final results of our method, and finally, we evaluate alternative optimization approaches and their limitations.

A. Method validation

To asses the accuracy of our method under controlled conditions, we constructed three test environments where ground-truth geometry and material parameters are known. The first setup is fully virtual, and consists of a rectangular plane, positioned and oriented to face the camera directly. Then, we apply a synthetic height map to the plane, and a principled Bidirectional Scattering Distribution Function (BSDF) model [43], obtaining rendered images that serve as reference captures for our height map optimization pipeline.

The second setup differs from the first one in the origin of the captures, which in this case come from the same optical capture system used for actual samples [7]. The test scene consists of a 3D-printed object, made from a translucent material. Similarly to the first setup, we apply our optimization pipeline to these captures and evaluate the results by comparing it with the known geometry of the 3D-print.

The third and final validation setup consists of a participating medium, defined over density and albedo volumes, representing a synthetic cloth sample. This volume is rendered using the same Mitsuba scene configuration employed during optimization, ensuring consistency between reference data and the reconstruction pipeline. The resulting images serve as reference captures for the optimization, yielding a recovered height map that we can quantitatively validate against the known geometry of the underlying synthetic volume.

The results of these experiments are presented in Figure 9. For the first setup, the surface material used for both the reference images and the optimization is defined using a blend BSDF consisting of two principled BSDF lobes, a reflective (75%) and a transmissive (25%) BSDF lobe with the corresponding values for Base color, Roughness, Anisotropic, Specular transmission, Index of refraction and Specular tint, of ([0.1, 0.15, 0.4], 0.7, 0.4, 0.0, 0.5, 0.3) and ([0.15, 0.08, 0.4], 0.8, 0.0, 1.0, 1.78, 0.0) respectively, as defined by Mitsuba's principled BSDF model.

Alongside the recovered height maps, we provide both qualitative and quantitative evaluations against the synthetic reference, computed after alignment of the reconstructed and reference maps. Since single-view height estimation is ambiguous up to an unknown global offset and scale, we first align the reconstructed and reference maps by subtracting the height value of a reference pixel located in a flat, known-height region. This alignment enables a consistent baseline
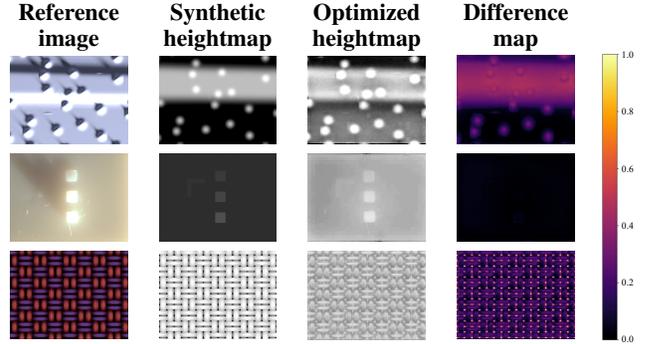

Fig. 9. Optimization results evaluation from synthetic data. Left to right, reference image, synthetic height map, optimized height map and difference map between the synthetic and optimized maps. Top to bottom: Synthetic height map on virtual mesh, optical capture of 3D-printed object and synthetic volumetric cloth definition. Height maps are contrast-enhanced for better visualization.

	FLIP ↓	LPIPS ↓	SSIM ↑	PSNR ↑
(A)	0.262 ± 0.093	0.359 ± 0.075	0.640 ± 0.071	22.59 ± 3.43
(B)	0.181 ± 0.019	0.292 ± 0.047	0.652 ± 0.020	25.12 ± 0.86
(C)	0.524 ± 0.042	0.568 ± 0.018	0.380 ± 0.038	14.42 ± 0.74
(D)	$\mathbf{0.520 \pm 0.033}$	$\mathbf{0.548 \pm 0.043}$	$\mathbf{0.421 \pm 0.035}$	$\mathbf{14.58 \pm 0.69}$
(E)	0.573 ± 0.146	0.444 ± 0.088	0.321 ± 0.095	13.67 ± 3.72

TABLE II
QUANTITATIVE METRICS EVALUATION. EVALUATED OVER 20 SAMPLES (AVERAGED ACROSS 8 NOVEL VIEWS). (A) VOLUMETRIC RENDERING VS GROUND-TRUTH PHOTOGRAPH. (B) VOLUMETRIC VS GROUND-TRUTH SYNTHETIC RENDER. (C)-(E) MATERIAL DIGITIZATION PIPELINE (BASELINE, WITH HEIGHT MAP) AND SVBRDF RECOVERY [44] VS GROUND-TRUTH PHOTOGRAPHS.

for evaluating relative surface structure, with a mean absolute error variation of ±30 μm induced by the choice of reference pixel, assuming the height map values range from 0 to 1 mm. Residual errors are visualized using difference maps, which serve as a qualitative illustration of the spatial distribution and magnitude of deviations. Additionally, in Table II, we report a set of standard quantitative metrics–PSNR, SSIM, FLIP, and LPIPS–computed on the aligned height maps to summarize reconstruction accuracy in terms of pixel-wise fidelity, structural consistency, and perceptual similarity. While more principled methods such as Depth from Focus [16] can, in theory, recover absolute scale, they do not provide sufficient accuracy for our experimental setup [15] and are therefore not considered further.

B. Results on fabric dataset

We used samples from the publicly available SEDDI DOME Dataset [45], which provides micro- and macro-level captures of various fabric materials under controlled illumination. We also include a small number of yet-unpublished samples from the same system and provided by the dataset authors.

Table III shows representative results. The optimization consistently recovers fine-grained surface features at the yarn level, including subtle relief patterns and local depth variations. Experiments on an NVIDIA RTX 4070 Ti, with an
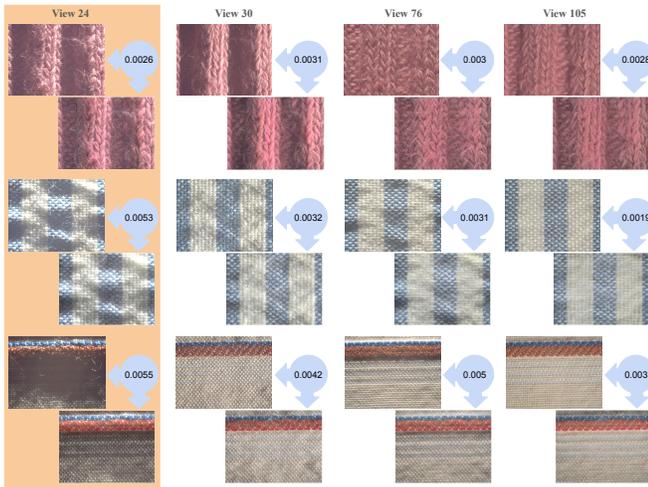
Fig. 10. Left to right, one light directions from the optimization set and three novel light directions, demonstrating versatility across light directions. Top to bottom, 3 different sample cloths, capture above and render below. The circles alongside each pair of images display a numerical comparison between each of the capture-render pairs, using the same loss function from optimization.

Fig. 11. Novel view points for three fabric samples, obtained with volumetric rendering changing the camera angle by 20º, 30º and 50º, pivoting on both axes around the sample. The bottom row shows an example of the surface geometry recovered by our method.

optimization time of ∼3.5h per sample. Due to the limited expressiveness of the volumetric phase function, specular highlights are not reproduced in the renders. To mitigate this, the loss function reduces the influence of specular highlights in the reference images and instead focuses on shadows and shading to avoid biasing the height-map optimization.

Additionally, to demonstrate the reliability and generalization of our method, we have generated path-traced volumetric renderings with novel light directions (Figure 10) and viewing angles (Figure 11), not included in the dataset used for the optimization process. Note that inaccuracies in density estimation produce a blurred output at gracing angles, due to accumulated scattering and absorption by low-density voxels, produced by the inherent ambiguity of having only one view during optimization. As we lack actual ground truth micro-images from different view points, we show a surface recovered with our method to facilitate the visualization.

C. Ablation study

To evaluate the impact of individual components of our pipeline, we conducted an ablation study on optimization configurations (loss function, parameters combinations and learning rate), as well as height map extraction method.

Loss function: We evaluated a logarithmic L2 variant designed to reduce luminosity for bright pixels, which may have a negative effect in the accuracy of height map optimization, while we preserve chromaticity in bright, saturated materials. As illustrated in Figure 12, this approach improves visual consistency for high-albedo samples, although its impact on recovered geometry is marginal.

Optimization configuration: We tested different combinations of jointly optimized parameters, having density optimization enabled in all cases, while albedo and phase function enabled status were combined differently, resulting in four different combinations. Non-optimized parameters were fixed
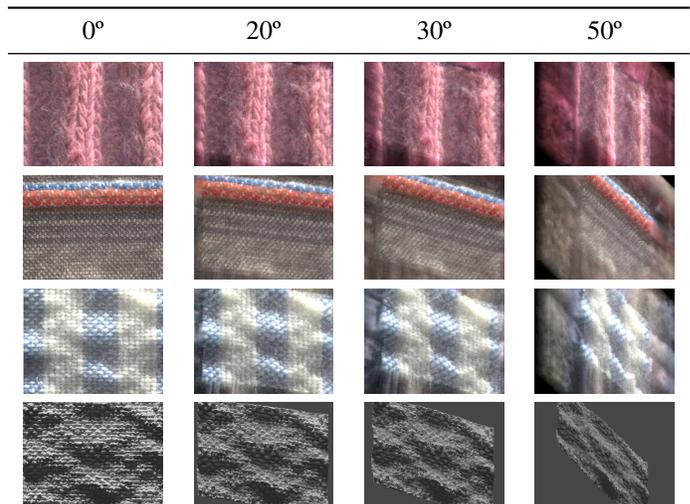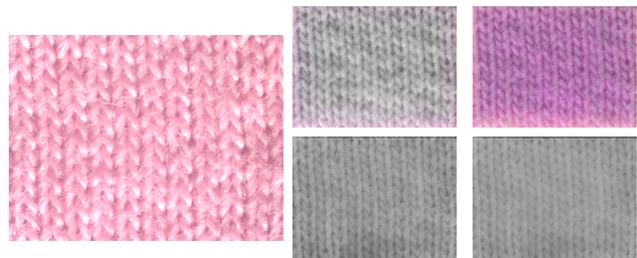

Fig. 12. The reference cloth image is shown on the left. Then, from left to right: Render (top) and height map (bottom) after optimizing with *L2* loss. Render (top) and height map (bottom) with *logarithmic L2* loss

to constant initial values (gray albedo and isotropic phase function). These combinations were evaluated under different learning rates. The most stable and visually consistent results were obtained when optimizing all 3 parameters, with learning rates of 0.06 for the density parameter, and 0.01 for the albedo and phase function parameters.

Qualitative comparisons for the most relevant variants are shown in Figure 13.

Height map extraction: In section III-A we explain how we used the Median-integrated Density method to extract a height map. In contrast to this method, a more naive approach would be to employ an isosurface extraction technique based on the Marching Cubes algorithm [46], and subsequently compute the Z-depth buffer by rasterizing its distance from the camera view. Although these results successfully depict depth information, they suffer from prominent floater-like artifacts related to fly-out threads (disconnected strands extending from the main surface), as illustrated in Figure 14 (a)). The resulting height maps from the Median-integrated Density method demonstrated significant improvements in artifact reduction, offering clearer structural delineation and enhanced overall fidelity compared to the naive approach (Figure 14 (b)).

	Photography	Render	Height map (ours)	Height map (photometric)	Normal map (ours)
(a)					
(b)					
(c)					
(d)					
(e)					
(f)					
(g)					
(h)					
(i)					
(j)					

TABLE III

OPTIMIZATION RESULTS FOR 10 SAMPLES. WE DISPLAY, FROM LEFT TO RIGHT, A REFERENCE CAPTURE FROM THE OPTICAL CAPTURE SYSTEM, A VOLUMETRIC RENDER OF THE OPTIMIZATION RESULT, OUR EXTRACTED HEIGHT MAP, THE HEIGHT MAP INTEGRATED FROM THE PHOTOMETRICAL NORMAL MAP AND THE NORMAL MAP DERIVED FROM OUR HEIGHT MAP. A MORE EXTENSIVE LIST OF SAMPLES IS SHOWCASED IN THE SUPPLEMENTAL MATERIAL ACCOMPANYING THIS PAPER.

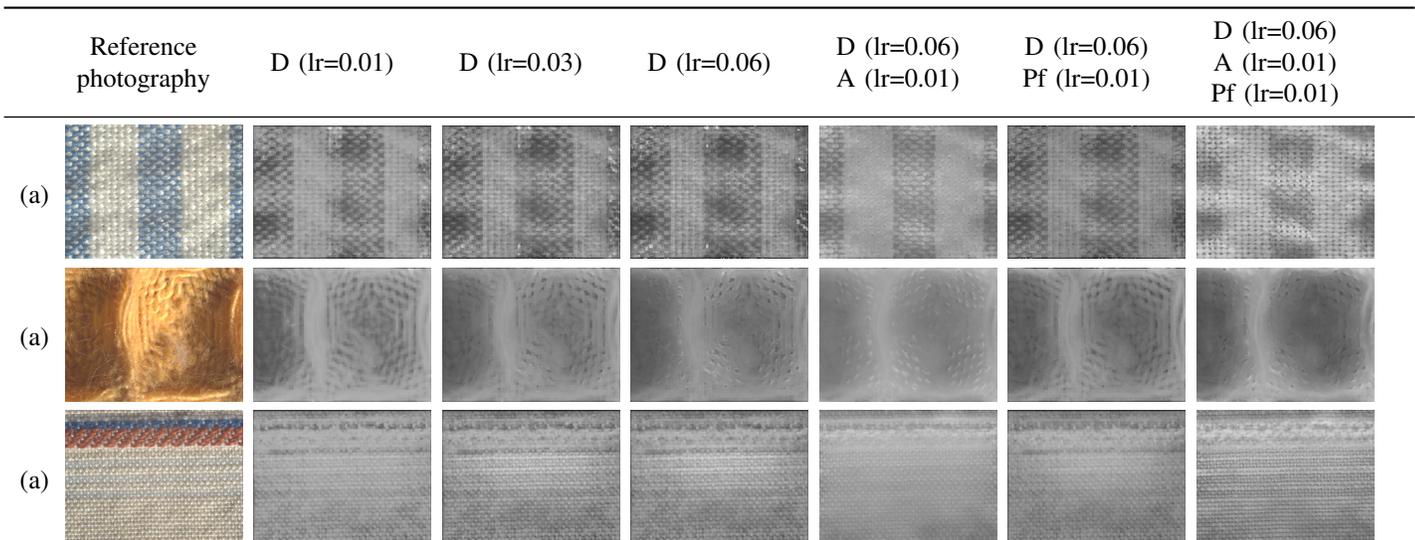|	Reference photography	D (lr=0.01)	D (lr=0.03)	D (lr=0.06)	D (lr=0.06) A (lr=0.01)	D (lr=0.06) Pf (lr=0.01)	D (lr=0.06) A (lr=0.01) Pf (lr=0.01)
(a)							
(a)							
(a)							

Fig. 13. Ablation study for 3 sample clothes and 6 relevant variants, with density (D), albedo (A) and phase function (Pf) parameters enabled or disabled in each variant. We first display the reference photography for each sample, and then, the height maps obtained from the optimization pipeline for each variant.

(a) Thresholding method.

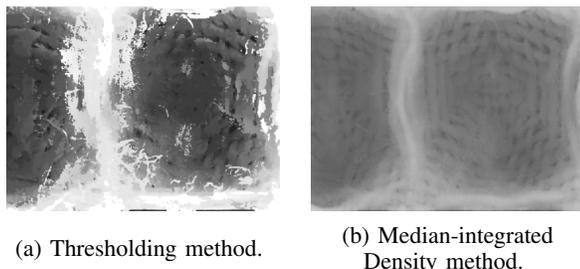(b) Median-integrated Density method.

Fig. 14. Comparison of two methods for extracting the height map from the volume. Note how the thresholding method suffers from floater-like artifacts that the Median-integrated Density method reduces significantly.

D. Comparison with alternative strategies

Surface-based approach: We alternatively explored a surface-based approach, optimizing a mesh of vertices (as described in the supplementary material). However, this method proved inadequate for modeling the complex light transport in volumetric translucent materials. Consequently, we adopted the volumetric representation described in Section III-A, which simulates full subsurface light scattering within the material.

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^{N} \omega_i \left(\hat{y}_i - y_i \right)^2 \tag{14}$$

Prior work: We compare our approach against a previous SVBRDF maps recovery method [44], by comparing the normal maps of both methods. In our method, the normal map is derived from the optimized height map, while the prior method recovers it directly from the image data. While our method is more limited by resolution, intrinsic decomposition approaches, missing depth data, often fail to generalize to novel illuminations and view points, due to lack of occlusion computation. Evaluation includes standard quantitative metrics, with the results summarized in Table II, and qualitative results displayed in Figure 15.
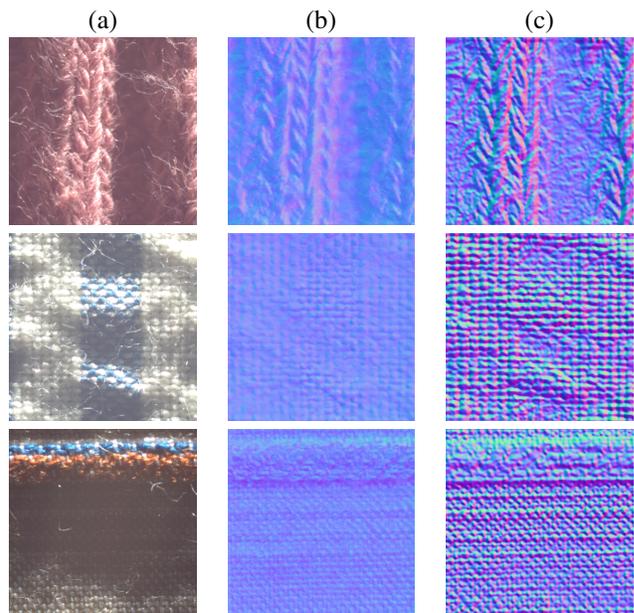
(a) (b) (c)

Fig. 15. We qualitatively compare the normal maps recovered using the SVBRDFrecovery method [44] (b) with the normal map derived from the optimized height map from our method (c). We display three different samples alongside the corresponding photographic capture (a), for reference.

VI. LIMITATIONS AND FUTURE WORK

A. Limitations

While our method demonstrates strong performance for micro-scale depth estimation in translucent materials, some limitations remain.

First, the approach is inherently designed to capture micro-patches and does not capture large-scale geometry or shape. Extending the system to capture larger structures could potentially generalize the method without sacrificing stability or largely modifying the system, but this remains untested.

Second, certain optical artifacts in the capture system, such

as lens shading and vignetting, introduce illumination inconsistencies that can subtly affect the optimization. Although mitigated through normalization and calibration, these effects remain a source of minor error. Additionally, fly-away fibers (thin or detached fibers) may still be partially present in the final height maps, though less prominently than in alternative extraction strategies, as discussed in Section V-C.

On top of that, some grid-like artifacts arise due to the resolution scaling between stages of the optimization. Differences in density between adjacent voxels in early stages are not properly smoothed by subsequent higher resolution stages, specially if the sample has a dark albedo, which provides a very low amount of information for the optimizer to work. The height map derived from the density volume also presents these artifacts, reducing the fidelity of the results.

Another challenge arises when materials exhibit highly specular behavior or extremely dark albedos, like the sample (a) in Table III, where optimization quality degrades significantly due to poor signal-to-noise ratio. Also, the optimizer will often find that lowering the height of a dark yarn, and assigning it a higher albedo produces a similar image to the ground-truth, which might converge to a plausible but incorrect local minimum. Cast shadows from multiple light directions help to reduce this kind of ambiguity, but they are still present in these cases.

B. Future work

We identify several directions for improvement.

- **Phase function improved model:** Although our volumetric optimization inherently models light scattering via the Henyey-Greenstein (HG) phase function, future work could explore more expressive subsurface scattering models, such as SGGX [47], or incorporate spatially varying scattering coefficients to better capture material-specific light transport behavior. However, ensuring the positive definiteness of the SGGX scattering matrix poses a significant numerical challenge, Although the SGGX matrix shares eigenvectors with the covariance matrix, its eigenvalues represent projected areas rather than variances, and the two coincide only in special cases. This distinction is largely benign when SGGX is used as a VNDF, but becomes problematic when used as a phase function, where the mapping from covariance to valid SGGX parameters is nonlinear and poorly conditioned.
- **Subsurface Scattering estimation:** While we compute accumulated volumetric scattering albedo values, we do not use it for optimization purposes, but this data could be exploited to explicitly estimating subsurface scattering (SSS) parameters as part of the material digitization pipeline, enhancing both visual realism in real time rendering and parameter estimation accuracy.
- **Extended use of inpainting:** To address ill-posed areas, such as those affected by extreme reflectivity from metallic threads, we already employ generative inpainting using RePaint [41], guided by masks derived from multiple texture maps. As future work, we propose extending this strategy to also handle other challenging regions, such

as underexposed (very dark) areas, and to explore tighter integration of the inpainting process into the optimization loop itself, potentially improving convergence and spatial consistency.
- **Density-aware ambient and shadow blur kernel:** In the material digitization pipeline used in this work, ambient illumination is approximated via a fixed function of the albedo map, and the shadow map is blurred using a constant Gaussian kernel. A promising direction for future work would be to condition both components on the underlying density volume (for instance, by integrating density along view or light directions to modulate ambient fill, and using local density and its spatial variation to drive adaptive shadow softness.
- **Exploration of additional material models:** While our experiment with a surface BSDF produced poor results with real objects, recent hybrid representations seem very promising, such as the layered BSDF proposed by [48].
- **Seamless tiling:** To support texture-based rendering pipelines, future work could explore tiling or stitching strategies to make the resulting height maps usable across larger surfaces, using SeamlessGAN [49].

VII. CONCLUSIONS

We presented a volumetric optimization translucent cloth geometry using a single-view, multi-light capture setup. By leveraging differentiable rendering and a density-based representation, our approach recovers detailed height maps explaining observed light transport while accounting for scattering.

We validated the method on synthetic data with ground truth geometry, and applied it to real captured cloth samples, showing improved depth quality and enhanced realism when integrated into a material digitization pipeline. Our ablation studies demonstrate the importance of albedo modeling, phase function tuning, and robust loss functions, while additional implementation strategies—such as inpainting reflective regions and refining alpha maps—further improve stability.

While our method does not estimate large-scale geometry beyond the patch level, and remains sensitive to extreme albedo or specularity conditions, it opens the door to scalable acquisition of cloth microstructure and may serve as a foundation for integrating subsurface scattering estimation in future work.

REFERENCES

[1] G. Choe, S. G. Narasimhan, and I. S. Kweon, "Simultaneous estimation of near ir brdf and fine-scale surface geometry," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2452–2460.
[2] R. Vidaurre, D. Casas, E. Garces, and J. Lopez-Moreno, "Brdf estimation of complex materials with nested learning," in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2019, pp. 1347–1356.

[3] A. Meka, M. Maximov, M. Zollhoefer, A. Chatterjee, H.-P. Seidel, C. Richardt, and C. Theobalt, "Lime: Live intrinsic material estimation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 6315–6324.

[4] M. Aittala, T. Weyrich, J. Lehtinen *et al.*, "Two-shot svbrdf capture for stationary materials." *ACM Trans. Graph.*, vol. 34, no. 4, pp. 110–1, 2015.

[5] Z. Li, K. Sunkavalli, and M. Chandraker, "Materials for masses: Svbrdf acquisition with a single mobile phone image," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 72–87.

[6] Z. Li, Z. Xu, R. Ramamoorthi, K. Sunkavalli, and M. Chandraker, "Learning to reconstruct shape and spatially-varying reflectance from a single image," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 6, pp. 1–11, 2018.

[7] E. Garces, V. Arellano, C. Rodriguez-Pardo, D. Pascual-Hernandez, S. Suja, and J. Lopez-Moreno, "Towards material digitization with a dual-scale optical system," *ACM Transactions on Graphics (TOG)*, vol. 42, no. 4, pp. 1–13, 2023.

[8] C. Li, T. T. Ngo, and H. Nagahara, "Inverse rendering of translucent objects using physical and neural renderers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 12 510–12 520.

[9] I. Gkioulekas, A. Levin, and T. Zickler, "An evaluation of computational imaging techniques for heterogeneous inverse scattering," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14*. Springer, 2016, pp. 685–701.

[10] M. Janner, J. Wu, T. D. Kulkarni, I. Yildirim, and J. Tenenbaum, "Self-supervised intrinsic image decomposition," *Advances in neural information processing systems*, vol. 30, 2017.

[11] Y. Dong, "Deep appearance modeling: A survey," *Visual Informatics*, vol. 3, no. 2, pp. 59–68, 2019.

[12] E. Garces, C. Rodriguez-Pardo, D. Casas, and J. Lopez-Moreno, "A survey on intrinsic images: Delving deep into lambert and beyond," *International Journal of Computer Vision*, vol. 130, no. 3, pp. 836–868, 2022.

[13] M. Glencross, G. J. Ward, F. Melendez, C. Jay, J. Liu, and R. Hubbold, "A perceptually validated model for surface depth hallucination," *ACM Transactions on Graphics (TOG)*, vol. 27, no. 3, pp. 1–8, 2008.

[14] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2650–2658.

[15] D. Pascual, E. Garces, C. Aliaga, and J. Lopez-Moreno, "Depth from focus: an application for fabrics captured at microscale," in *CVPR Workshop*, 2019.

[16] A. P. Pentland, "A new sense for depth of field," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-9, no. 4, pp. 523–531, 1987.

[17] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao, "Deep ordinal regression network for monocular depth estimation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 2002–2011.

[18] R. Li, K. Xian, C. Shen, Z. Cao, H. Lu, and L. Hang, "Deep attention-based classification network for robust depth prediction," in *Computer Vision–ACCV 2018: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers, Part IV 14*. Springer, 2019, pp. 663–678.

[19] R. Ranftl, K. Lasinger, D. Hafner, K. Schindler, and V. Koltun, "Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 3, pp. 1623–1637, 2020.

[20] H. Kato, D. Beker, M. Morariu, T. Ando, T. Matsuoka, W. Kehl, and A. Gaidon, "Differentiable rendering: A survey," *arXiv preprint arXiv:2006.12057*, 2020.

[21] S. Zhao, W. Jakob, and T.-M. Li, "Physics-based differentiable rendering: a comprehensive introduction," *ACM SIGGRAPH 2020 Courses*, vol. 14, pp. 1–14, 2020.

[22] T.-M. Li, M. Aittala, F. Durand, and J. Lehtinen, "Differentiable monte carlo ray tracing through edge sampling," *ACM Trans. Graph. (Proc. SIGGRAPH Asia)*, vol. 37, no. 6, pp. 222:1–222:11, 2018.

[23] N. Ravi, J. Reizenstein, D. Novotny, T. Gordon, W.-Y. Lo, J. Johnson, and G. Gkioxari, "Accelerating 3d deep learning with pytorch3d," *arXiv:2007.08501*, 2020.

[24] W. Jakob, S. Speierer, N. Roussel, M. Nimier-David, D. Vicini, T. Zeltner, B. Nicolet, M. Crespo, V. Leroy, and Z. Zhang, "Mitsuba 3 renderer," 2022, https://mitsuba-renderer.org.

[25] W. Jakob, S. Speierer, N. Roussel, and D. Vicini, "Dr.jit: A just-in-time compiler for differentiable rendering," *Transactions on Graphics (Proceedings of SIGGRAPH)*, vol. 41, no. 4, Jul. 2022.

[26] M. Nimier-David, D. Vicini, T. Zeltner, and W. Jakob, "Mitsuba 2: A retargetable forward and inverse renderer," *Transactions on Graphics (Proceedings of SIGGRAPH Asia)*, vol. 38, no. 6, Dec. 2019.

[27] P. Khungurn, D. Schroeder, S. Zhao, K. Bala, and S. Marschner, "Matching real fabrics with micro-appearance models." *ACM Trans. Graph.*, vol. 35, no. 1, pp. 1–1, 2015.

[28] I. Gkioulekas, S. Zhao, K. Bala, T. Zickler, and A. Levin, "Inverse volume rendering with material dictionaries," *ACM Transactions on Graphics (TOG)*, vol. 32, no. 6, pp. 1–13, 2013.

[29] C. Che, F. Luan, S. Zhao, K. Bala, and I. Gkioulekas, "Towards learning-based inverse subsurface scattering," in *2020 IEEE International Conference on Computational Photography (ICCP)*. IEEE, 2020, pp. 1–12.

[30] T. K. Nindel, T. Iser, T. Rittig, A. Wilkie, and J. Křivánek, "A gradient-based framework for 3d print appearance optimization," *ACM Transactions on Graphics (TOG)*, vol. 40, no. 4, pp. 1–15, 2021.

[31] P. Weier, J. Riviere, R. Guseinov, S. Garbin, P. Slusallek, B. Bickel, T. Beeler, and D. Vicini, "Practical inverse rendering of textured and translucent appearance," *Transactions on Graphics (Proceedings of SIGGRAPH)*, vol. 44, no. 4, Aug. 2025.

[32] W. Jakob, A. Arbree, J. T. Moon, K. Bala, and S. Marschner, "A radiative transfer framework for rendering materials with anisotropic structure," *ACM Trans. Graph.*, vol. 29, no. 4, Jul. 2010.

[33] K. Schroder, R. Klein, and A. Zinke, "A volumetric approach to predictive rendering of fabrics," *Computer Graphics Forum*, vol. 30, no. 4, pp. 1277–1286, 2011.

[34] S. Zhao, W. Jakob, S. Marschner, and K. Bala, "Building volumetric appearance models of fabric using micro ct imaging," *ACM Trans. Graph.*, vol. 30, no. 4, Jul. 2011.

[35] L. G. Henyey and J. L. Greenstein, "Diffuse radiation in the galaxy," *Astrophysical Journal, vol. 93, p. 70-83 (1941).*, vol. 93, pp. 70–83, 1941.

[36] O. CIE, "Official recommendations on uniform color space, color difference equations and metric color terms," *CIE Publ*, vol. 15, p. 1976, 1976.

[37] M.-J. Rakotosaona, F. Manhardt, D. M. Arroyo, M. Niemeyer, A. Kundu, and F. Tombari, "Nerfmeshing: Distilling neural radiance fields into geometrically-accurate 3d meshes," *arXiv preprint arXiv:2303.09431*, 2023.

[38] M. Worchel and M. Alexa, "Differentiable shadow mapping for efficient inverse graphics," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023, pp. 142–153.

[39] S. Laine, J. Hellsten, T. Karras, Y. Seol, J. Lehtinen, and T. Aila, "Modular primitives for high-performance differentiable rendering," *ACM Transactions on Graphics*, vol. 39, no. 6, 2020.

[40] I. Sobel, "An isotropic 3x3 image gradient operator," *Presentation at Stanford A.I. Project 1968*, 02 2014.

[41] A. Lugmayr, M. Danelljan, A. Romero, F. Yu, R. Timofte, and L. V. Gool, "Repaint: Inpainting using denoising diffusion probabilistic models," 2022.

[42] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst. Man Cybern.*, vol. 9, pp. 62–66, 1979.

[43] B. Burley, "Extending the disney brdf to a bsdf with integrated sub-surface scattering," *SIGGRAPH Course: Physically Based Shading in Theory and Practice. ACM, New York, NY*, vol. 19, no. 7, p. 9, 2015.

[44] D. Luo, H. Sun, L. Ma, J. Yang, and B. Wang, "Correlation-aware encoder-decoder with adapters for svbrdf acquisition," in *Proceedings of SIGGRAPH Asia 2024*, 2024.

[45] E. Garces, V. Arellano, C. Rodriguez-Pardo, D. Pascual-Hernandez, S. Suja, and J. Lopez Moreno, "SEDDI DOME Dataset," 2023.

[46] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3d surface construction algorithm," *SIGGRAPH Comput. Graph.*, vol. 21, no. 4, p. 163–169, Aug. 1987.

[47] E. Heitz, J. Dupuy, C. Crassin, and C. Dachsbacher, "The sggx microflake distribution," *ACM Transactions on Graphics (TOG)*, vol. 34, no. 4, pp. 1–11, 2015.

[48] B. Wang, W. Jin, M. Hašan, and L.-Q. Yan, "Spongecake: A layered microflake surface appearance model," *ACM Trans. Graph.*, vol. 42, no. 1, Sep. 2022.

[49] C. Rodriguez-Pardo and E. Garces, "Seamlessgan: Self-supervised synthesis of tileable texture maps," *IEEE Transactions on Visualization and Computer Graphics*, 2022.