Task 4. Cloth Rendering

Garoe Dorta-Perez
CM50245: Computer Animation and Games II

May 28, 2015

1 Introduction

Rendering realistic images is a challenging task, specially if there are memory or time constrains for the computation. Cloth is a complex material composed of interwoven threads of different types. Moreover, its appearance can vary from diffuse to highly specular.

2 Previous work

Solving for global illumination generally involves providing a solution to the rendering equation [6]. A simple approach is to focus on simulating the micro-geometry of the cloth in conjunction with global illumination. The light scattering is simulated for each fibre in the thread and the fibres are modelled as perfect cylinders. To be able to model the the complete scattering effects on a surface, Bidirectional Scattering-Surface Reflectance Distribution Function (BSSRDF) [10] have been used. Under this paradigm, complex light phenomena like subsurface scattering can be modelled with sufficient accuracy. One of the earliest approaches was based on simple empirical shading models [18]. The main objective was to accomplish believable shading, disregarding physical accuracy. Ever since, an increasing number of methods have been proposed, this techniques can be broadly divided into two groups, *data-based* models and *procedurally-based* models.

The *data-based* approach focuses on collecting reflectance information that will later be used to model the cloth. Bidirectional Texture Function (BTF) [3] is a function that is often used to sample BRDF information from the fabric. Sattler et al. [12] captured BTF from a rectangular probed and generated view-dependant texture-maps for the cloth. Wang et al. [16] proposed a variation of the previous technique, the authors modeled the captured data as a microfacet-based BRDF to which they fitted a normal distribution function for each facet. Within the data-based models, a volumetric method was proposed by Zhao et al. [20]. The authors capture a X-ray computed tomography (CT) scanner of the fabric, which is later matched to images of the same material.

Procedurally-based techniques emphasise on improving previous BRDF models in order to simulate cloth appearance realistically. Kang [7] proposed a procedural method which models the reflectance of woven fabric using an alternating anisotropy and deformed microfacet distribution. A weft or warp anisotropic function is computed for each fragment which has been previously classified into one of the two groups. A model with focus on accurately handling specular reflections was proposed by Irawan and Marschner [4]. The authors' model does not required image data and it relies on textures for close up views. Additionally, they are able to reproduce both the small scale and large scale appearance of woven cloth. The limitations of this model include a costly numerical data fit step, decreased accuracy for views at grazing angles and a restricted number of specular highlights.

A survey of physically-based methods in cloth simulations was presented by Schröder et al. [14], we also refer the readers to Yuen and Wünsche's [19] survey for example-based and procedural-based techniques.

3 Methodology

In this section we will discus in detail the theoretical and practical aspects of the chosen paper.

3.1 Light scattering model

We have chosen to implement a recent paper [11] which presents a microcylinder based model for fast and realistic cloth rendering. This model overcomes the limitations of Irawan and Marschner's work [4] and it can also be implemented in GPU where it achieves real time performance. The authors propose a model of fabric based on two microcylinders oriented in orthogonal directions, as shown in Figure 1. The reflectance model for a single thread in the fabric is defined as

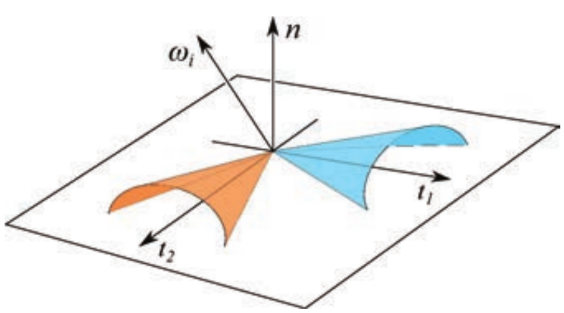


Figure 1: Sadeghi et al. shading model [11], where $\boldsymbol{\omega_i}$ is the incident light direction, $\mathbf{n}$ is the surface normal and $\mathbf{t_1}, \mathbf{t_2}$ are the orthogonal thread directions.

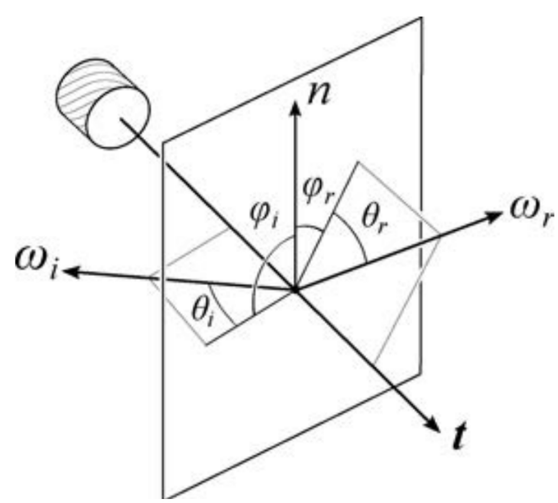


Figure 2: Sadeghi et al. shading model [11], showing θ and ϕ angles given a pair of directions $\boldsymbol{\omega_i}$ and $\boldsymbol{\omega_r}$.

$$L_r = \int \frac{\left(f_{r,s}(\mathbf{t}, \boldsymbol{\omega_i}, \boldsymbol{\omega_r}) + f_{r,v}(\mathbf{t}, \boldsymbol{\omega_i}, \boldsymbol{\omega_r})\right) L_i(\boldsymbol{\omega_i}) \cos(\theta_i) \delta \boldsymbol{\omega_i}}{\cos^2(\theta_d)}, \tag{1}$$

where $\mathbf{t}$ is the thread direction, $\boldsymbol{\omega_i}$ is the ray incoming direction, $\boldsymbol{\omega_r}$ is the ray outgoing direction, θ_i, θ_r, ϕ_i and ϕ_r are angles as shown in Figure 2, $\theta_d = \theta_i - \theta_r$ and L_i is the incoming radiance in the evaluated point. Note that radiometric notation [8] is used to define L_r, which represents the outgoing radiance over a infinitesimal arc length of the cylinder and how the integral extends over the entire sphere instead of the typical hemisphere.

The surface reflection term in Equation 1 is defined as

$$f_{r,s}(\mathbf{t}, \boldsymbol{\omega_i}, \boldsymbol{\omega_r}) = F_r(\eta, \boldsymbol{\omega_i}) \cos(\phi_d/2) g(\gamma_s, \theta_h), \tag{2}$$

where $\theta_h = (\theta_i + \theta_r)/2$, $\phi_d = \phi_i - \phi_r$, F_r is a Fresnel reflection term that is computed using Schlick's approximation [13] $F_r(\eta, \boldsymbol{\omega_i}) = \eta + (1-\eta)(1 - \mathbf{h} \cdot \boldsymbol{\omega_i})^5$ where $\cdot$ is the vector dot product operator, $\mathbf{h} = (\boldsymbol{\omega_i} + \boldsymbol{\omega_r}) / |\boldsymbol{\omega_i} + \boldsymbol{\omega_r}|$ is the normalized halfway vector, η is the reflectance for $\mathbf{h} \cdot \boldsymbol{\omega_i} = 1$, $g(\gamma, \theta) = \gamma e^{\mathbf{j} \cdot \mathbf{q} - 1}$ is a Gaussian lobe [17] where $\mathbf{q}$ is the lobe axis, the direction $\mathbf{j}$ is the spherical parameter in the resulting function and γ is the amplitude. The $\cos(\phi_d/2)$ term controls the intensity given the incoming and outgoing rays, and the Gaussian term is added to simulate surface roughness in the previously smooth cylinders, this Gaussian term is therefore an approximation to microfacets on the cylinders.

The volume scattering term in Equation 1 is defined as

$$f_{r,v}(\mathbf{t}, \boldsymbol{\omega_i}, \boldsymbol{\omega_r}) = F_t(\eta, \boldsymbol{\omega_i}) F_t(\eta', \boldsymbol{\omega_r'}) \frac{(1 - k_d) g(\gamma_v, \theta_h) + k_d}{\cos(\theta_i) + \cos(\theta_r)} \mathbf{k_a}, \tag{3}$$

where k_d is a scattering constant, $\mathbf{k_a}$ is an rgb albedo constant vector, $F_t = 1 - F_r$ is a Fresnel transmission term, $\boldsymbol{\omega_r'}$ is a projection of $\boldsymbol{\omega_r}$ into a plane that contains the normal $\mathbf{n}$ and η' is computed using the Bravais index [8] as $\eta'(m) = \sqrt{\eta^2 - sin^2(m)} / \cos(m)$ where m is the angle between $\boldsymbol{\omega_r}$ and its projection $\boldsymbol{\omega_r'}$. The Gaussian term $g(\gamma_v, \theta_h)$ in introduced to control the width of the scattering cone shown in Figure1, and the division by $\cos(\theta_i) + \cos(\theta_r)$ is added as a normalization factor [2].

The outgoing radiance in the path shown in Figure 2 is

$$L_r(\boldsymbol{\omega_r}) = a_1 L_{r,1}(\boldsymbol{\omega_r}) + a_2 L_{r,2}(\boldsymbol{\omega_r}), \tag{4}$$

where a_1 and a_2 are the area coverage ratio for the first and second thread within the patch, in our implementation we assumed a watertight pattern of equally size threads, leading to $a_1 = a_2 = 0.5$, L_1 and L_2 are the outgoing radiances for the first and second thread computed as shown in Equation 1.

In order to compute a thread direction $\mathbf{t}$ without using external data structures, we follow a fixed texture, axis as shown in Figure 3. Given an intersection point in a triangle $\mathbf{p} = [x_p, y_p, z_p]$ and its texture coordinates $\mathbf{u} = [u_p, v_p]$. We define $\mathbf{t}$ vector as the vector $\mathbf{t} = (\hat{\mathbf{p}} - \mathbf{p}) / |\hat{\mathbf{p}} + \mathbf{p}|$ such that $\hat{\mathbf{p}}$ texture coordinates are $\hat{\mathbf{u}} = [u_p + 1, v_p]$. In our shader we can easily compute the texture coordinates of the triangle vertices, however we can not immediately get a world position from a new texture coordinate. We assume that the 3D world space to texture space transformation is an affine matrix T, that is constant for each triangle and it can be written as

$$\mathbf{u}T = \mathbf{p} \rightarrow \begin{pmatrix} u & v \end{pmatrix} \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{pmatrix} = \begin{pmatrix} x & y & z \end{pmatrix}. \tag{5}$$

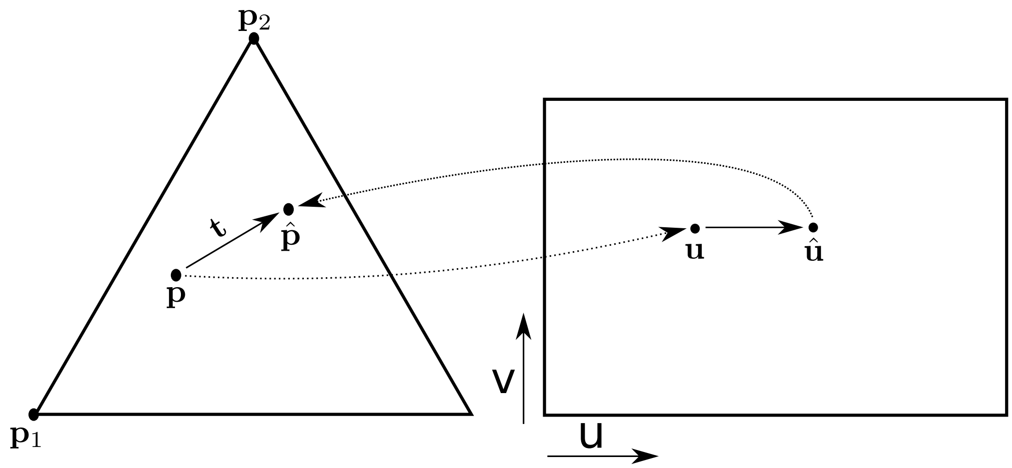


Figure 3: To compute the vector $\mathbf{t}$, we follow the texture coordinates of the intersection point $\mathbf{p}$.

The system in Equation 5 has six unknowns and three equations, therefore we need two pairs of world points - texture coordinates to solve T. Writing out the terms for two known points $\mathbf{p}_1$ and $\mathbf{p}_2$ in the triangle,

$$\begin{aligned} x_1 &= u_1 a_{11} + v_1 a_{21}, \quad y_1 = u_1 a_{12} + v_1 a_{22}, \quad z_1 = u_1 a_{13} + v_1 a_{23}, \\ x_2 &= u_2 a_{11} + v_2 a_{21}, \quad y_2 = u_2 a_{12} + v_2 a_{22}, \quad z_2 = u_2 a_{13} + v_2 a_{23}. \end{aligned} \tag{6}$$

Solving for each term analytically,

$$a_{21} = \frac{u_2 x_1 - u_1 x_2}{u_2 v_1 - u_1 v_2}, \qquad a_{22} = \frac{u_2 y_1 - u_1 y_2}{u_2 v_1 - u_1 v_2}, \qquad a_{23} = \frac{u_2 z_1 - u_1 z_2}{u_2 v_1 - u_1 v_2}, \tag{7}$$

$$a_{11} = \frac{x_1 - v_1 a_{21}}{u_1}, \qquad a_{12} = \frac{y_1 - v_1 a_{22}}{u_1}, \qquad a_{13} = \frac{z_1 - v_1 a_{23}}{u_1}. \tag{8}$$

Notice that the term of $1/(u_2 v_1 - u_1 v_2)$ is shared, so it can be computed only once per triangle in order to increase performance. The aforementioned method will give the thread direction $\mathbf{t}_1$, for $\mathbf{t}_2$ the process is equivalent with the exception that $\hat{\mathbf{u}}$ will be incremented on the v direction, $\hat{\mathbf{u}} = [u_p, v_p + 1]$.

3.2 Shading model

In order to render cloth fabrics the authors evaluate the outgoing radiance from each patch, which is assumed to be locally flat and smaller than a pixel in the image. This patch is defined as the smallest thread weaving patter such that the fabric can be constructed by repeating this patch, as shown in Figure 4. For each thread in the patch a tangent curved is defined, this curve will be sampled at fixed positions giving the normal direction in that position, as shown in Figure 4. The BRDF will be evaluated for each sample and the total outgoing radiance will be computed as follows,

$$L_r(\boldsymbol{\omega_r}) = \frac{1}{N_j}\sum_{\mathbf{t}}\int L_i(\boldsymbol{\omega_i}) f_s(\mathbf{t}, \boldsymbol{\omega_i}, \boldsymbol{\omega_r}) \cos\theta_i d\boldsymbol{\omega_i}, \tag{9}$$

where N_j is the number of samples.

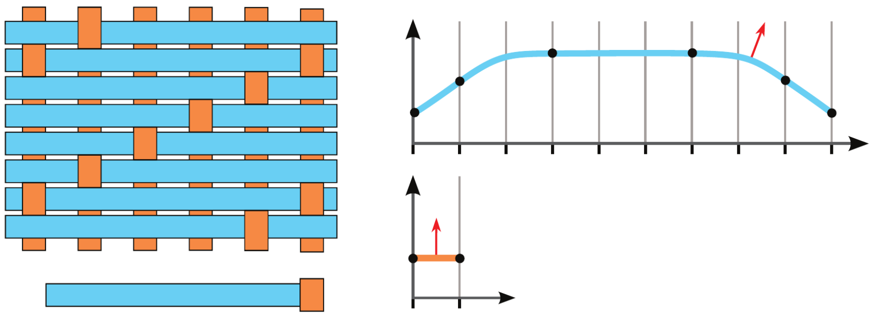


Figure 4: Weaving pattern and tangent curves: (top left) a weaving pattern example, (bottom left) the smallest patch, (top right) tangent curve for the blue thread in the smallest patch, (bottom right) tangent curve for the orange thread in the smallest patch, arrows indicate a normal sampled at that point, image taken from [11].

Sadeghi et al. also adds a shadowing and masking term to improve the quality of their results, the new term will account for threads occluding each other depending on the viewing direction. The masking effect will be specially relevant at grazing angle viewing and lighting directions. In Equation 9, $f_s(\mathbf{t}, \boldsymbol{\omega_i}, \boldsymbol{\omega_r})$ will be multiplied by $M(\mathbf{t}, \boldsymbol{\omega_i}, \boldsymbol{\omega_r})$ such that,

$$\begin{aligned} M(\mathbf{t}, \boldsymbol{\omega_i}, \boldsymbol{\omega_r}) =& (1 - u(\phi_d)) \max(\cos(\phi_i), 0) \times \max(\cos(\phi_r), 0) + \\ & u(\phi_d) \min(\max(\cos(\phi_i), 0), \max(\cos(\phi_r), 0)), \end{aligned} \tag{10}$$

where $\times$ is the vector cross product operator, u is a unit height Gaussian function with standard deviation between 15°and 25°; an in depth discussion on how to compute u can be found in Ashikmin et al. [1] Section 4.2 New Shadowing Term.

The authors add a reweighting factor that improves the results when $\boldsymbol{\omega_i}$ and $\boldsymbol{\omega_r}$ are far from n. By projecting the thread length in the direction of the viewer, threads that are more visible for the viewer will have a greater contribution. The reweighting factor $P(\mathbf{t}, \boldsymbol{\omega_i}, \boldsymbol{\omega_r})$ will be computed as follows,

$$\begin{aligned} P(\mathbf{t}, \boldsymbol{\omega_i}, \boldsymbol{\omega_r}) =& (1 - u(\Psi_d)) \max(\cos(\Psi_i), 0) \times \max(\cos(\Psi_r), 0) + \\ & u(\Psi_d) \min(\max(\cos(\Psi_i), 0), \max(\cos(\Psi_r), 0)), \end{aligned} \tag{11}$$

where Ψ are the angles between $\mathbf{n}$ and the projection of $\boldsymbol{\omega}$ into the plane that contains $\mathbf{t}$ and $\mathbf{n}$, as shown in Figure 5.

There is also a normalization factor Q to account for the number of samples that were introduced in Equation 9 and for weaving patterns that are not water tight,

$$Q = \frac{a_1}{N_1}\sum_{\mathbf{t}} P(\mathbf{t}, \boldsymbol{\omega_i}, \boldsymbol{\omega_r}) + \sum_{\mathbf{t}} P(\mathbf{t}, \boldsymbol{\omega_i}, \boldsymbol{\omega_r}) + (1 - a_1 - a_2)(\boldsymbol{\omega_r} \cdot \mathbf{n}), \tag{12}$$

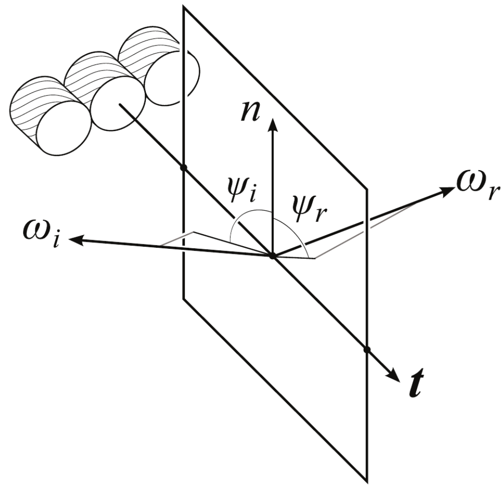


Figure 5: Ψ angles in the thread for a given $\boldsymbol{\omega_i}$ and $\boldsymbol{\omega_r}$, image taken from [11].

where N_1 and N_2 are the number of samples of each thread direction.

Finally, the complete model for the outgoing radiance is

$$L_r(\boldsymbol{\omega_r}) = \frac{1}{QN_j} \sum_{\mathbf{t}} \int L_i(\boldsymbol{\omega_i}) f_s(\mathbf{t}, \boldsymbol{\omega_i}, \boldsymbol{\omega_r}) M(\mathbf{t}, \boldsymbol{\omega_i}, \boldsymbol{\omega_r}) P(\mathbf{t}, \boldsymbol{\omega_i}, \boldsymbol{\omega_r}) \cos\theta_i d\boldsymbol{\omega_i}. \tag{13}$$

3.3 Other implementation considerations

In order to compute more easily quantities such as the θ or ϕ angles introduced in the previous section, we will defined a new coordinate system with the vectors $\{\mathbf{t}, \mathbf{n}, \mathbf{s}\}$, whose centre will be the triangle ray-intersection point $\mathbf{p}$, where $\mathbf{s}$ is a normalized vector $\mathbf{s} = \mathbf{n} \times \mathbf{t}$. The matrix that will transform from world coordinates to this system is

$$M = M_{trans} M_{rot} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -p_x & -p_y & -p_z & 1 \end{pmatrix} \begin{pmatrix} t_x & n_x & s_x & 0 \\ t_y & n_y & s_y & 0 \\ t_z & n_z & s_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \tag{14}$$

such that $\mathbf{x}M = \mathbf{x}'_{new}$, where $\mathbf{x}$ is the original row vector in homogeneous coordinates and $\mathbf{x}'_{new}$ is the transformed vector.

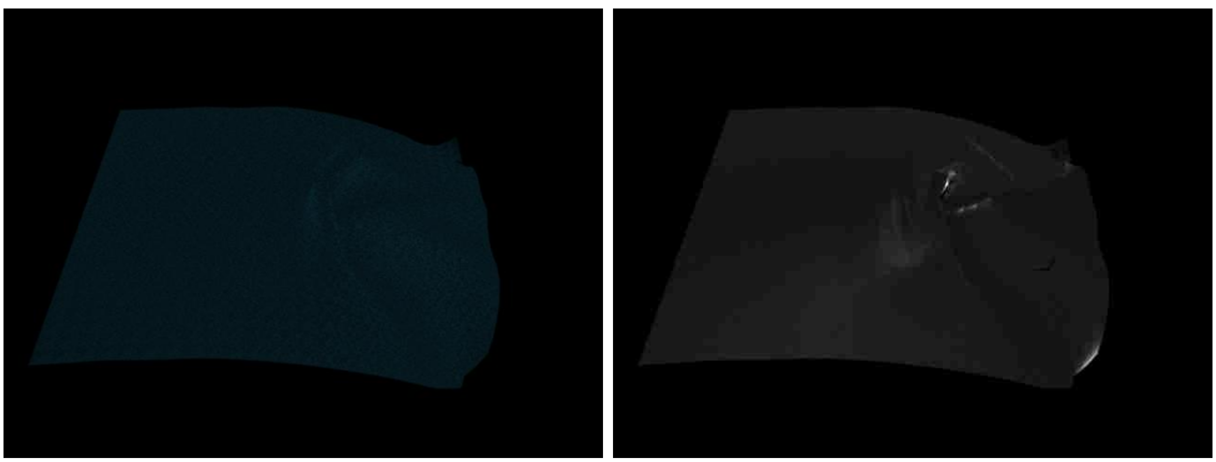
We decided to implement the shader in Maya's®Mental Ray®rendering software. The rationale under this choice lays in the advantages of integrating the work into a established framework, which allows us to easily use the shader for dynamic cloth simulations. Mental Ray™approximates Kajiya's equation [6] using photon mapping [5]. Our approach is to compute an initial estimate of the outgoing radiance L_r for each direct hit during the raytracing path. In the photon map construction stage each photon hit will locally sample L_r using Equation 1, from this samples an average irradiance L_i will be evaluated, which will then be added to the initial estimate as shown below

$$L_r = L_{ri} + \sum_{i=1}^{I} \psi_i L_{ri}, \tag{15}$$

where I is the total number of photons inside a fixed radius around $\mathbf{p}$ and ψ_i is the normalized flux of the i^{th} photon, which is computed from an initial arbitrary flux shared by all photons and decreases with an absorption rate per bounce.

4 Results

We have implemented the basic light scattering model in Mental Ray®. In order to provide a deeper insight of the effects of the different terms in the BRDF, results for the different parts are evaluated separately. The cloth rendered using only the volume scattering term in shown in Figure 6a, using only the reflection term is shown in Figure 6b, note that for visualization purposes the reflection term has been increased by a factor of 5. Lastly, a rendered image for the fabric using both terms is shown in Figure 7.



(a) Effect of the volume scattering term.

(b) Effect of the reflection term, scaled by 5 improve visualization.

Figure 6: Evaluating the effects of the different terms of the BRDF.

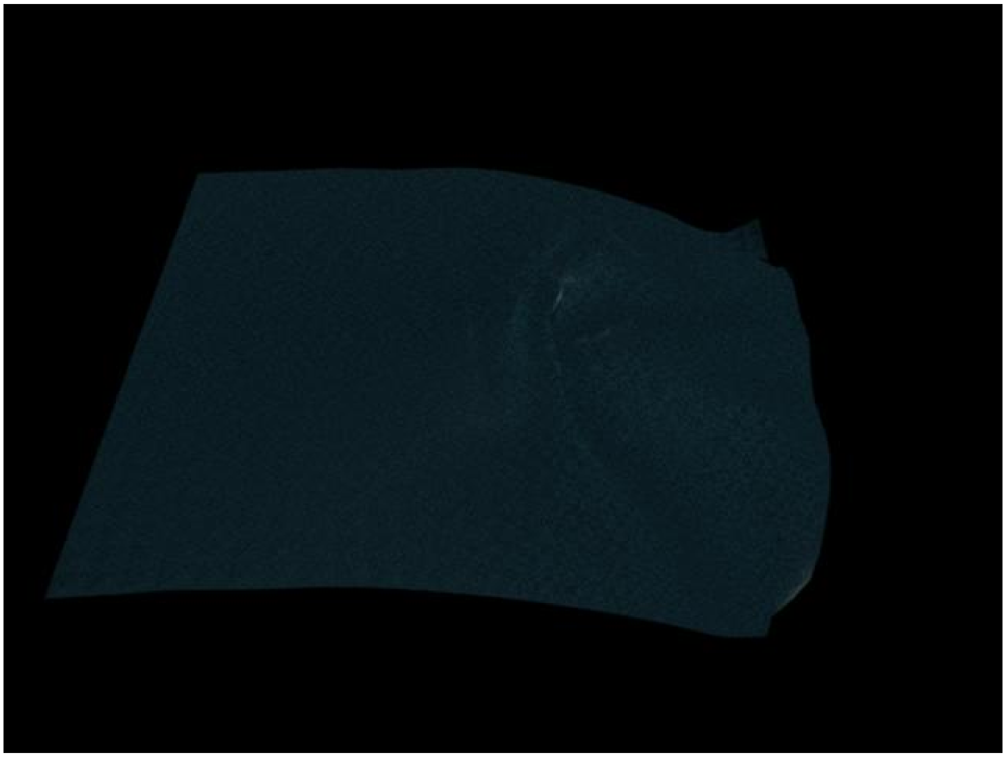


Figure 7: Final result.

5 Conclusion and Future Work

Our implementation only includes the basic light scattering proposed by the authors [11]; so a evident line of future work would be to add the remaining features from the paper into our code. Sadeghi et al. model works under the assumption that the cloth patches are smaller than a pixel in the image, this leads to incorrect rendered images at close ups. Another drawback of the model is the optimization of parameters, being so complex that they have to be adjusted manually. How to efficiently apply importance sampling to reduce the rendering times and increase quality has been addressed with several extensions [15, 9].

References

[1] M Ashikmin, S Premože, and Peter Shirley. A Microfacet-based BRDF Generator. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 65–74, 2000.

[2] Subrahmanyan Chandrasekhar. *Radiative transfer*. Courier Corporation, 2013.

[3] K.J. Dana, S.K. Nayar, B. Van Ginneken, and J.J. Koenderink. Reflectance and texture of real-world surfaces. *ACM Transactions on Graphics*, 18(1):1–34, 1999.

[4] Piti Irawan and Steve Marschner. Specular reflection from woven cloth. *ACM Transactions on Graphics*, 31(1):1–20, 2012.

[5] Henrik W. Jensen. Global illumination using photon maps. In *Rendering Techniques' 96*, pages 21–30. Springer-Verlag, 1996.

[6] James T. Kajiya. The rendering equation. *SIGGRAPH Computer Graphics*, 20(4): 143–150, 1986.

[7] Young-Min Kang. Realtime rendering of realistic fabric with alternation of deformed anisotropy. In *Motion in Games*, pages 301–312. Springer, 2010.

[8] Stephen R. Marschner, Henrik Wann Jensen, Mike Cammarano, Steve Worley, and Pat Hanrahan. Light scattering from human hair fibers. *ACM Transactions on Graphics*, 22(3):780–791, 2003.

[9] Kazutaka Mizutani and Kei Iwasaki. Importance Sampling for Cloth Rendering under Environment Light. In *Mathematical Progress in Expressive Image Synthesis I*, pages 81–88. 2014.

[10] Fred E. Nicodemus, Joseph C. Richmond, Jack J. Hsia, Irving W. Ginsberg, and Thomas Limperis. *Geometrical considerations and nomenclature for reflectance*. National Bureau of Standards Washington, DC, USA, 1977.

[11] Iman Sadeghi, Oleg Bisker, Joachim de Deken, and Henrik Wann Jensen. A practical microcylinder appearance model for cloth rendering. *ACM Transactions on Graphics*, 32(2):1–12, 2013.

[12] Mirko Sattler, Ralf Sarlette, and Reinhard Klein. Efficient and realistic visualization of cloth. In *Proceedings of the 14th Eurographics Workshop on Rendering*, EGRW '03, pages 167–177, 2003.

[13] Christophe Schlick. An Inexpensive BRDF Model for Phsyically-based Rendering. In *Computer Graphics Forum*, volume 13, pages 233–246, 1994.

[14] Kai Schröder, Shuang Zhao, and Arno Zinke. Recent Advances in Physically-Based Appearance Modeling of Cloth. In *SIGGRAPH Asia 2012 Courses*, pages 12:1–12:52, 2012.

[15] Chunpo Wang, Feng Xie, and Parashar Krishnamachari. Importance Sampling for a Microcylinder Based Cloth Bsdf. In *ACM SIGGRAPH 2014 Talks*, pages 41:1—-41:1, 2014.

[16] Jiaping Wang, Shuang Zhao, Xin Tong, John Snyder, and Baining Guo. Modeling anisotropic surface reflectance with example-based microfacet synthesis. *ACM Trans. Graph.*, 27(3):41:1–41:9, 2008.

[17] Jiaping Wang, Peiran Ren, Minmin Gong, John Snyder, and Baining Guo. All-Frequency Rendering of Dynamic, Spatially-Varying Reflectance. *ACM Transactions on Graphics*, 28(5):133:1–133:10, 2009.

[18] Jerry Weft and Murray Hill. The Synthesis of Cloth Objects. *SIGGRAPH Computer Graphics*, 20(4):49–54, 1986.

[19] Wallace Yuen and B Wünsche. An evaluation on woven cloth rendering techniques. In *Proceedings of the 26th International Image and Vision Computing New Zealand Conference (IVCNZ 2011)*, pages 7–12, 2011.

[20] Shuang Zhao, Wenzel Jakob, Steve Marschner, and Kavita Bala. Building volumetric appearance models of fabric using micro ct imaging. *ACM Trans. Graph.*, 30(4):44:1–44:10, 2011.