Volumetric Skin and Fabric Shading at Framestore

Nathan Walster

1 INTRODUCTION

On *Guardians of the Galaxy Vol. 2* and *Alien: Covenant*, we were faced with the problem of how to shade and render complex translucent creatures. We will present our approach, which uses a volumetric Monte-Carlo path-tracing framework, in the following sections. We will also discuss how we worked with production artists to parameterise the system, and extensions that allowed us to render volumetric fabric.

1.1 Existing Workflow

In previous projects, Framestore had used a variety of techniques to render materials with a subsurface appearance. Our approach had been to try to maintain a balance between the adoption of current technology and ensuring we had a system that can work in a production environment. Like any studio, this required considering factors such as artist control, image quality and render time cost.

Prior to the method described in these course notes, our existing workflow was based around a ray-traced BSSRDF importance sampling (IS) technique [Kin+13], utilising a normalised diffusion approximation [Chr15; CKB16]. For media with prominent anisotropic scattering, we often incorporated a single-scattering component [Jen+01]. We then combined this, in an energy conserving manner, with a microfacet specular model [Hd14].

Although we were careful to ensure energy conservation between the reflective and transmissive components, the subsurface scattering lobes were usually combined in an ad hoc manner. This approach was artist led, and usually driven by textures and blended weight factors. Although this often achieved compelling results, it was time consuming and unintuitive. Artists often found that they needed to add an extra diffuse component, or mix in further subsurface scattering with a different diffusion profile. This led to a complex layering and blending of shading models that became difficult to control, and in turn could become restrictive when responding to supervisor and client feedback. It was also a challenge for developers to offer artists this level of freedom whilst still guaranteeing efficiency at render time.

1.2 Motivation

In early 2016, we began work on *Guardians of the Galaxy Vol. 2* and *Alien: Covenant*. Both of these shows contained complex translucent creatures that we realised would present difficult challenges to the look development team. A particular issue with our existing approach had always been the rendering of internal structure. Translucent skin, muscles, organs and a skeleton — all shaded with believable attenuation and scattering — was beyond the scope of our ad hoc method. We had previous experience with this on *Guardians of the Galaxy Vol. 1*, and although artists had combined the scattering components in inventive ways, it had been difficult to achieve the results that were required. A particular observation had been that characters behaved unreliably in differing lighting environments. Render times had also been significant.

In order to address these issues, we began a development effort focused around using Monte-Carlo volumetric shading techniques. The goal was to develop a common framework that could be used in many areas of our shading library. This would provide us with the ability to create compelling visual results for these new creatures, but also give us something to extend and develop in the future. We also aimed to provide artists with a unified interface and a simplified set of parameters. We hoped that by removing the reliance on ad hoc blending and complex interaction between many models, the underlying light transport would also be simplified and would offer an elegant way to scatter between multiple overlapping objects. Another important requirement was that the system needed to support a variety of input data, whether homogeneous, 2D textured or volumetric.

1.2.1 Abilisk

The Abilisk (Figure 1) was a hero creature that appeared in the opening sequence of the *Guardians* film. Its outer skin varied in thickness from a thick, dense appearance to a thin membrane, and the director was very keen to be able to see multiple layers of translucent organs and an internal structure to its body.

There was also a need to portray the scale of this creature, with ridges and wrinkles on his surface. These are typically areas where a diffusion approximation approach to subsurface scattering is inaccurate.

1.2.2 Chest Burster

The Chest Burster (Figure 2) was an iconic character that we needed to recreate for *Alien: Covenant*. It was composed of translucent layers of membranes, veins and liquid.

Figure 1: Abilisk.

Figure 2: Chest Burster.

2 SKIN SHADING

2.1 Path-Traced Subsurface Scattering

Path-traced Monte-Carlo subsurface scattering offers a simple and robust approach to solving the kinds of volumetric integration problems we were facing, but is often considered too computationally expensive for production use, particularly for highly scattering media. We felt that, given the inherent per-sample cost of BSSRDF importance sampling and the complexity of solutions our artists had been inclined to use, the overhead might not actually be too high. Furthermore, we had already been using some of the required techniques in our conventional volume rendering, so we had some understanding of how these could be applied in a production environment.

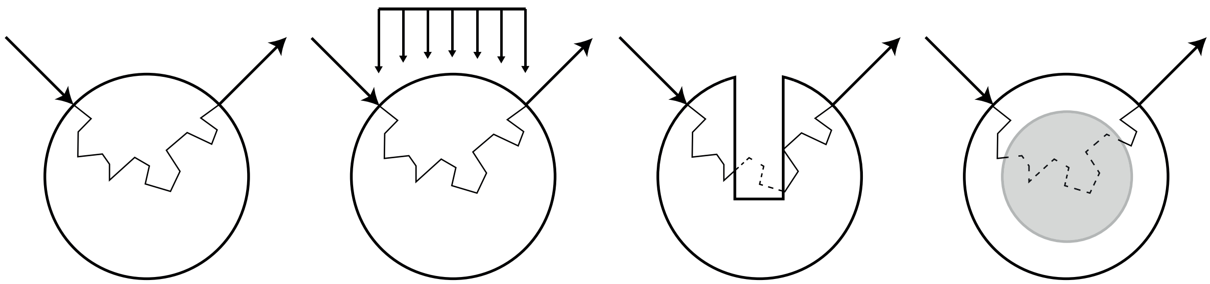

Figure 3: Left to right: a subsurface light transport path; projecting rays using BSSRDF IS; non-trivial geometry; internal geometry.

2.2 Theory

Monte-Carlo subsurface scattering relies on sampling volumetric space using an unbiased estimator. There are two key considerations: the sampling of direction and distance when scattering through a medium, and the computation of transmission when computing next event estimation. In the following sections, we will briefly describe some of the common distance sampling approaches, as well as phase functions for directional sampling.

2.2.1 *Homogeneous vs Heterogeneous*

The simplest volumetric scattering case describes a medium that is homogeneous, i.e. one that has uniform optical properties throughout. These mediums are very convenient for sampling as extinction can be computed analytically. In contrast, a heterogeneous medium is one where the optical properties — such as absorption and scattering coefficients and phase function parameterisation — can change throughout the volume.

2.2.2 *Ray-Marching*

Ray-marching is a classic volume integration technique where (often) fixed sized steps are taken through a medium. This technique suffers unpredictable levels of bias and often requires many small steps to resolve areas of fine detail. It is non-trivial for artists to choose a step size that offers a good balance between visual quality and performance.

2.2.3 *Delta/Woodcock Tracking*

Delta (or Woodcock) tracking [Woo+65] is an unbiased, free-flight distance sampling method. Unlike ray-marching, it can adaptively skip over thin regions and concentrate on dense regions where a scattering event is likely to occur. It relies on having an estimation of the maximum extinction coefficient of the medium, and the evaluation cost is highly dependent on how close this is to the true extinction coefficient. If the medium if heterogeneous and the extinction is varying, it may be necessary to subdivide the volume into pieces, each of which has a local estimation maximum.

This method is typically unsuitable for computing transmittance, since it yields a coarse binary estimator.

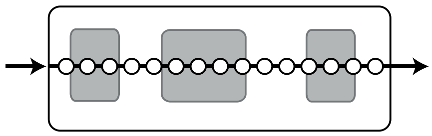

Figure 4: Ray-marching steps are unable to adapt to the local density within the media, whilst delta tracking is able to do so.

2.2.4 *Ratio/Residual Tracking*

Ratio tracking [NSJ14] is an extension of delta tracking that can be used to compute unbiased transmittance estimates. It is an efficient and unbiased method for integrating heterogeneous media. Residual ratio tracking is a further improvement that tracks a residual volume extinction estimate relative to a control variate that has an analytic solution. As with Woodcock tracking, its performance is dependent on having good bounds for the extinction coefficient along a ray.

2.2.5 *Phase Functions*

A phase function describes the angular distribution of scattering given an incoming direction. It can also be used to sample ray directions as we integrate the medium. A common choice is the Henyey-Greenstein phase function [HG41], since it offers a convenient parameterisation between backscattering, isotropic and forward scattering, and can be economically importance sampled.

Figure 5 illustrates the scattering directions generated by different parameterisations of the Henyey-Greenstein function.

Multiple phase functions can be combined to describe more complex angular scattering profiles, for example a strongly peaked forward scattering lobe with a broader backscattering lobe.

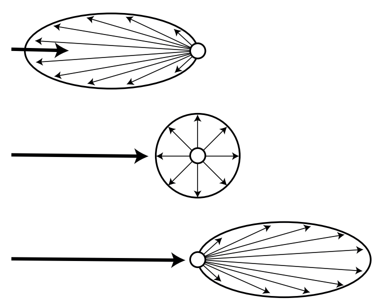

Figure 5: Top to bottom: backscattering; isotropic scattering; forward scattering.

2.3 Practice

Consideration of the various techniques led us to implement a general-purpose combination of delta, residual and ratio tacking. Our approach follows the following methodology:

- At a surface intersection, we sample a microfacet half angle, evaluate Fresnel, and select between sampling a specular or transmissive component.

- If refracting, a microfacet BTDF is evaluated to obtain a refracted ray direction into the object and an associated throughput weight.
- Based on the optical properties of the medium inside the object, we use delta tracking to sample a distance and associated PDF.
- A ray is fired to see if there is a geometric intersection up to the sampled distance.
- While there are no intersections, we create a new path vertex at the scatter distance and update the path throughput with the scatter distance PDF. A phase function is sampled at the new vertex to obtain a new direction to sample along.
- When a geometric intersection is found we update the path throughout with the probability of the scatter distance, taking into account that is larger than the geometric distance.
- We perform next event estimation at all vertices on object boundaries.

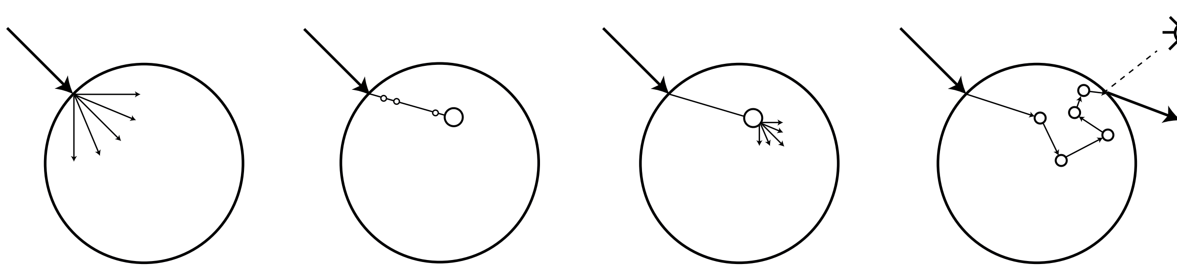

Figure 6: Steps to our method: sampling a BTDF; using delta tracking to sample a distance; sampling a phase function to obtain a direction; a completed path with next event estimation at exit vertex.

The pseudo-code of our method is illustrated in Algorithm 1.

Algorithm 1: Pseudocode of our integration method.

```
function InteriorScatter (a, b);
while path length is less than maximum do
    Choose R,G,B of extinction to scatter from, based on current path throughput;
    Sample scatter distance and PDF up to maximum distance;
    Trace ray up to scatter distance;
    if geometric intersection then
        Update path throughput with probability of scatter distance being larger than geometric distance;
        break;
    else
        Create new path vertex at scatter distance;
        Update path throughput with original scatter distance PDF;
        Sample phase function to obtain new direction;
    end
end
```

2.3.1 Heterogeneous Properties

We found that one of the most important considerations to make was to identify which areas of our volume we could consider homogeneous, and which areas had to be parameterised in a heterogeneous way. This was often a balance between visual quality, pragmatism regarding render cost, and ease of authoring the data that would be used to vary the optical parameters.

We found that most cases could be covered by a few main approaches, which we present in order of complexity:

- A simple way to vary the appearance of a surface is to use a 2D texture. This is convenient for an artist to author and visualise, and studios will typically have a robust set of tools for propagation and management throughout the pipeline. As a ray entered the medium it would access a texture at the surface and consider any volumetric properties within to be homogeneous with these values. Whilst this approach is not able to offer a true volumetric description of scattering properties, it proved to be adequate for highly scattering media.
- For thin or detailed shallow subsurface features, such as smaller veins, we found it necessary to obtain a feeling of parallax below the surface. Whilst integrating throughout the medium, we used geometrical or ray-traced projection back to the outer surface to obtain a UV parameterisation, which could then be used for texture evaluation.
- For larger-scale subsurface structure, we used a set of procedural noise functions evaluated throughout the medium. In order to have these deform correctly, they were evaluated in a volumetric space that deformed with the object.

Artists were able to blend these approaches using surface-based masks.

We also found that, in many cases, a homogeneous medium with underlying geometry works well and can be rendered economically. For a character, this would typically be a skull or other reasonably dense geometry such as cartilage. The relative depths of the internal geometry can create view-dependent visual complexity that is difficult to achieve with surface-based textures. Larger veins and arteries also worked well this way.

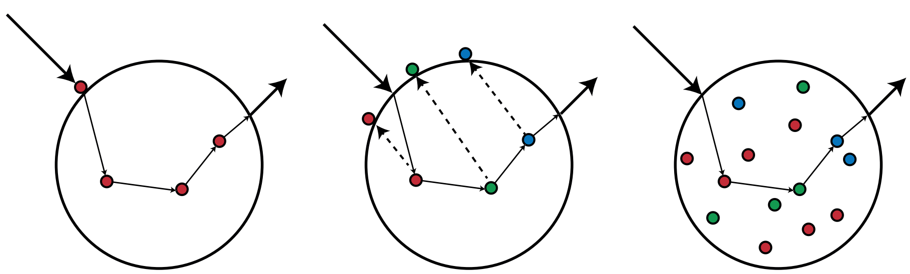

Figure 7: Methods for projecting texturing heterogeneous media. Left to right: projecting values along the incoming ray; projecting back to the bounding surface to sample a texture; sampling a procedural noise function.

2.3.2 *Interaction with Fresnel, Specular and Roughness*

At each interaction with a surface, we typically use Fresnel to select between a specular or transmissive lobe. We begin by sampling a half angle, and performing a Fresnel calculation to obtain the relative weight for each lobe. IOR and surface roughness naturally fit into this workflow. These BSDFs are evaluated at all interactions with the surface. Artists are free to add secondary specular lobes to achieve more complex effects, such as coatings.

2.3.3 *Caustics and Transparent Shadows*

We observed that different next event estimation techniques work best for different densities of media.

- For dense media, such as skin, we compute next event estimation at the exiting point.
- For less dense media, such as water or liquid, we compute next event estimation at each interior vertex (single scattering), using transparent shadows to pass through the geometry.
- For internal surfaces, we often use Manifold Next Event Extimation (MNEE) [HDF15] to produce a caustic, attenuated by transmittance through the media.

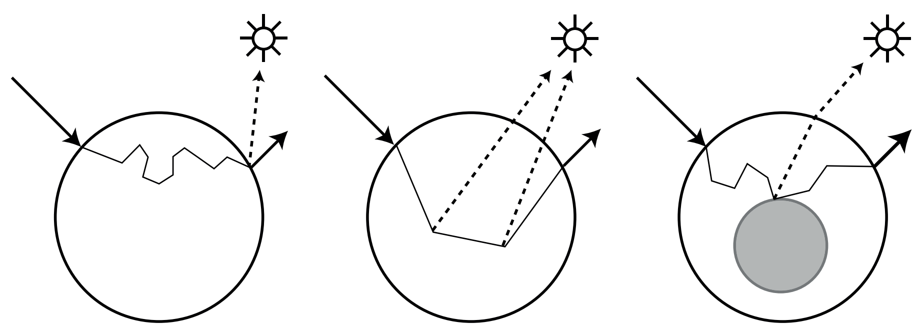

Figure 8: Next event estimation. Left to right: computed at exit point; internal scattering points; via MNEE.

Whilst the latter two techniques contain an element of bias, we found that they offered a pragmatic approach to reducing variance and reproduced the important visual quality of the medium. We leave the control to artists to choose a method, and discuss improvements to this approach at the end of the notes.

2.4 Artist Workflow

We experimented with various ways to parameterise our workflow and the following sections discuss how we decided to present it to look development artists. We will also discuss the workflow considerations that we discovered during this process.

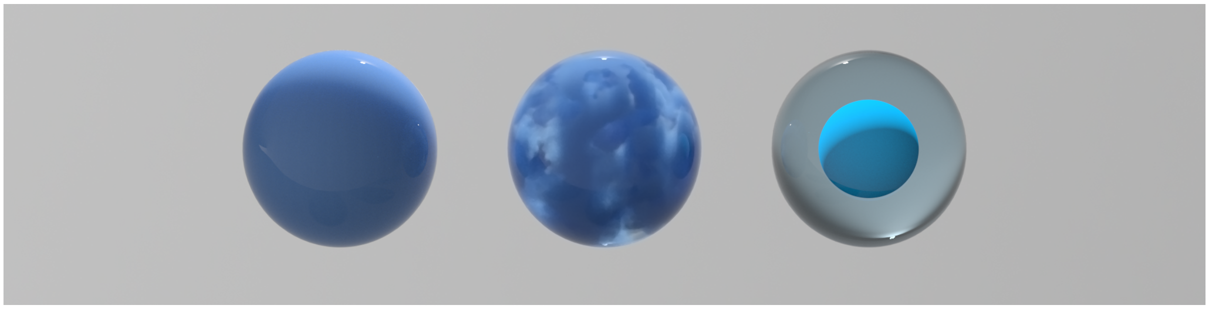

Figure 9: Example renders of different next event estimation techniques.

2.4.1 *Parameterisation*

Framestore uses a modular shading system and in order to encapsulate our volumetric shading workflow, we supplemented our existing classes of co-shader — such as `bsdf` and pattern generation — with a new class of `interior` co-shaders. Artists can choose from the following:

- `Absorption:` used for coloured glass.
- `Homogeneous Scattering:` used for the majority of skin, and other scattering media such as water.
- `Heterogeneous Scattering:` used for hero skin and bespoke volumetric characters.
- `Voxelised:` allows for spatial variation of coefficients via voxel datasets.

These can in turn be combined. For example: a homogeneous scattering ocean, with subsurface foam provided via a voxel dataset.

2.4.2 *Scattering and Absorption*

Our system provides artists with a unified set of controls to parameterise the shaders. For more intuitive control, we chose to define the volumetric absorption and scattering coefficients as `albedo` and `attenuation`. In our system, `albedo` provides a colour that the artist wants the medium to be, and `attenuation` applies a tinting effect over depth. These two parameters can be converted to the absorption (σ_a), scattering (σ_s) and extinction coefficients (σ_t) using the following formulae:

$$\sigma_t = \frac{1}{\texttt{attenuation}}, \tag{1}$$

$$\sigma_s = \texttt{albedo} \cdot \sigma_t, \tag{2}$$

$$\sigma_a = \sigma_t - \sigma_s. \tag{3}$$

As discussed in [CKB16], a single-scattering albedo does not provide a good fit to the final result, as multiple scattering begins to dominate. We use their method to convert from a artist-supplied albedo that includes multiple scattering to the single-scattering albedo used during integration. Whilst it is intended for a diffusion approximation, we found that it provided a good fit.

The following renders (Figure 10), (Figure 11), (Figure 12) show how our `albedo`, `attenuation` and `density` parameters change the appearance of the final medium.

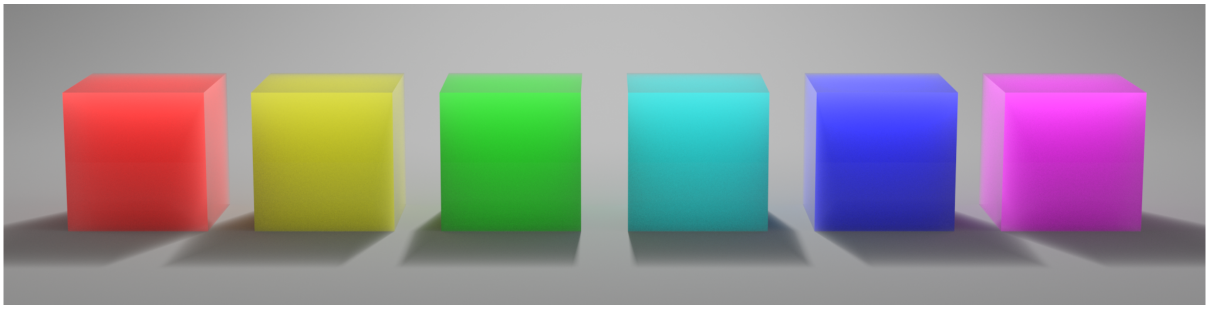

Figure 10: Example renders showing our `albedo` parameter.

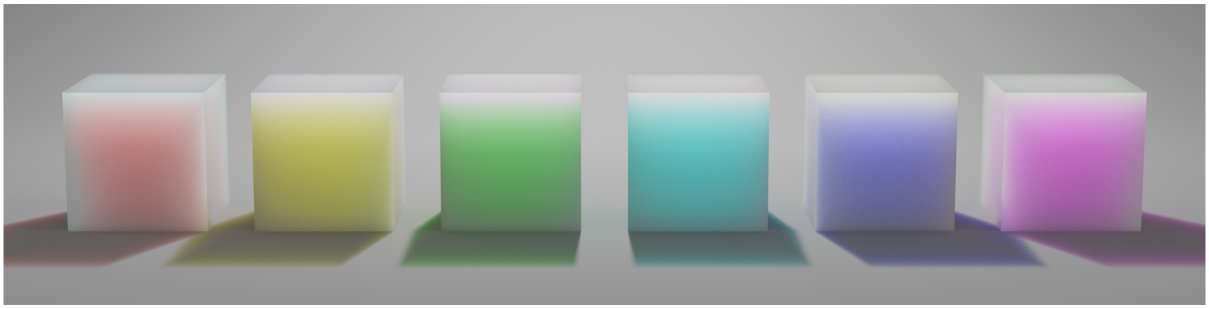

Figure 11: Example renders showing our `attenuation` parameter.

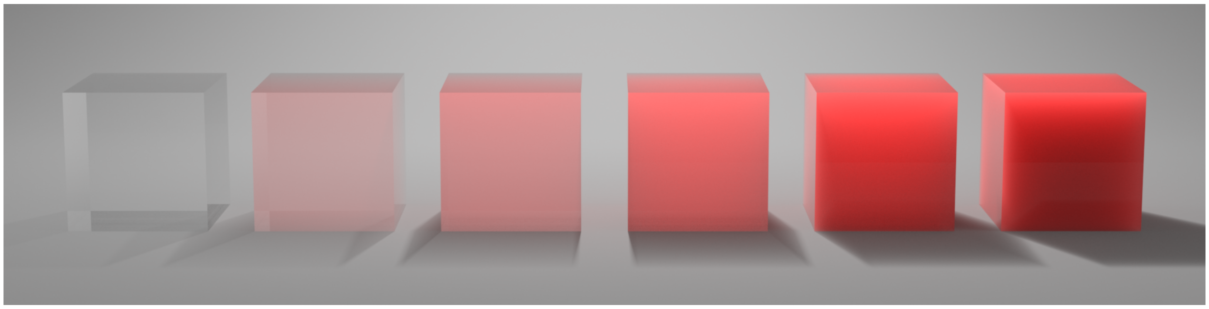

Figure 12: Example renders showing our `density` parameter.

We also provided a selection of different phase functions, with a single or double Henyey-Greenstein being the most common. Artists are also able to select the BTDF used at surface interactions. For dense media that is highly scattering, we use a transmissive diffuse BTDF; for less dense media, we use a microfacet BTDF that has either a Beckmann or GGX distribution model.

2.4.3 *Bounding the Extinction Coefficient*

The requirement of delta, ratio and residual tracking to have good bounds on the extinction coefficient presented challenges to our workflow. With a homogeneous volume, the artist-provided attenuation parameter, combined with a global density value, could be used to obtain an accurate bound. With a heterogeneous volume, we dealt with the problem in two ways:

- When rendering voxelised data, our in-house voxel format provides access to coarser levels of tiled data that could be used to provide a local estimate of the bounds.
- When rendering media with a procedural density[1], it was more difficult to obtain a local estimate. Artists could provide a global minimum and maximum, but this solution was not ideal. We will discuss future improvements to this approach later in the course notes.

2.4.4 *Shader Examples*

Figure 13 shows examples of our absorption, homogeneous scattering, heterogeneous scattering and voxelised interior shaders.

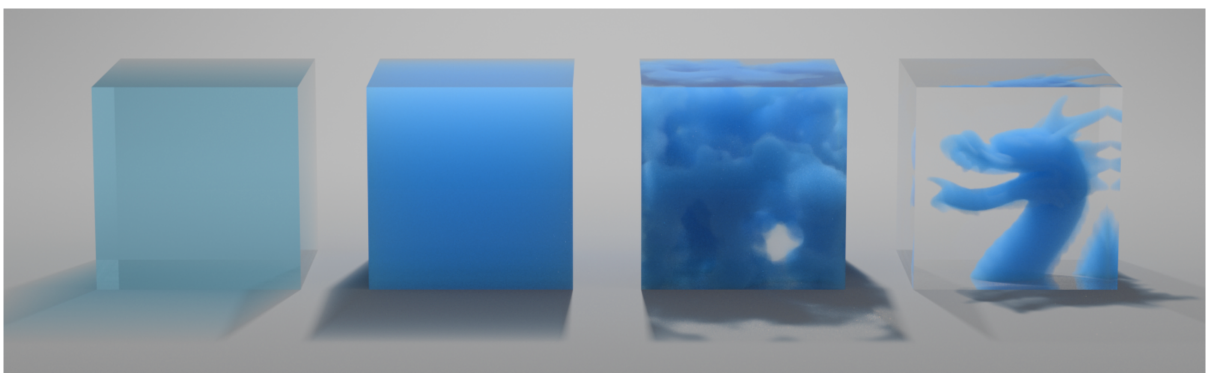

Figure 13: Example renders showing our different interior shaders.

2.4.5 *Artist Considerations*

One of the immediate observations we made was that the modelling of the geometry became very important to the look-development process. Whereas previously an artist would often use a set of textures to drive the densities of the scattering, we now wanted to rely on the underlying geometric shape to give us a more natural scattering behaviour. This process required us to improve our feedback mechanism between the modelling and look development stages, and develop better ways to review modelling so that supervisors could make more informed decisions as to how the model would perform when rendered.

[1] Such as a fractal Brownian motion (fBm) noise function.

2.4.6 *Render Times*

As discussed in the motivation for this work, the render times for our previous method had been a challenge, particularly when combining many layers of translucent media. With our new approach, we are able to draw the following comparisons:

- With dense scattering media, render times for Monte-Carlo subsurface are roughly equal to the BSSRDF IS technique. This assumes surface-based texturing and projection into the volume from the entry point.
- When blending BSSRDF IS with areas of less dense single scattering, Monte-Carlo subsurface becomes more optimal.
- When rendering multiple layers of translucent media, Monte-Carlo subsurface is significantly faster.

3 FABRIC SHADING

Fabric shading is another area where a volumetric approach can offer an improved way to represent a complex material. We had previously used microcylinder-based BRDF models [Sad+13; WXK14], but whilst these were useful for capturing far-field fabric appearance characteristics, they did not generally work for materials with larger woven structures.

3.1 Procedural Fabric Modelling

There has been significant recent work in generating procedural yarns and weave patterns [Khu+15; ZLB16]. With a limited set of parameters, these methods enable the generation of woven sets of fibres that can be used to represent fabrics. We chose to use this procedural approach as it offered, for a small overhead, a way to generate patches of fabric on the fly, thus avoiding the need to store large datasets on disk.

Once the fabric representation is generated, it can be voxelised and used as an input to a heterogeneous medium. The ray transport and evaluation uses the same framework as our skin shading. However, we introduced a new `interior` shader that consisted of the procedural generation of yarn and weave patches, voxelisation, and a probabilistic BSDF shading model.

3.2 Scattering Within Fabric

Fabric is composed of many fibres that exhibit scattering in a similar way to hair and fur. As we were voxelising our fibres and rendering at a sub-pixel level, we took the approach of treating the scattering as a probabilistic event. During voxelisation, we compute average tangent vectors for the fibre field and use these as the basis for a hair model, based on [Chi+16]. We found that a combination of R, TT and diffuse lobes offered a good compromise between visual quality, artist controllability and performance.

3.3 Workflow

Our workflow consists of the following:

- Generate yarn, ply and weave patterns, based on artist driven parameters.
- Voxelise patches of the weave curves, encoding tangent vectors and weft/warp IDs.
- At each sampling position, project back to the surface and query a voxel position.
- Drive optical properties by the averaged voxel data.
- Shade using a hair-based BSDF.

3.4 Artist Parameterisation

We chose to give artists control over several areas of the procedural fibre generation. The majority of our fabrics comprise of a small set of weave patterns combined with a fibre profile and scattering model, from which were able to get a broad range of looks. Our main parameters are:

- `Weave:` plain, twill, basket and satin weaves.
- `Velvet:` a mode to generate vertical fibres with procedural variation.
- `Flyaways:` a scalar control to add a percentage of loose flyaway fibres.
- `Weft/Warp Albedo:` separate warp and weft scattering properties, for the creation of two-tone fabrics.

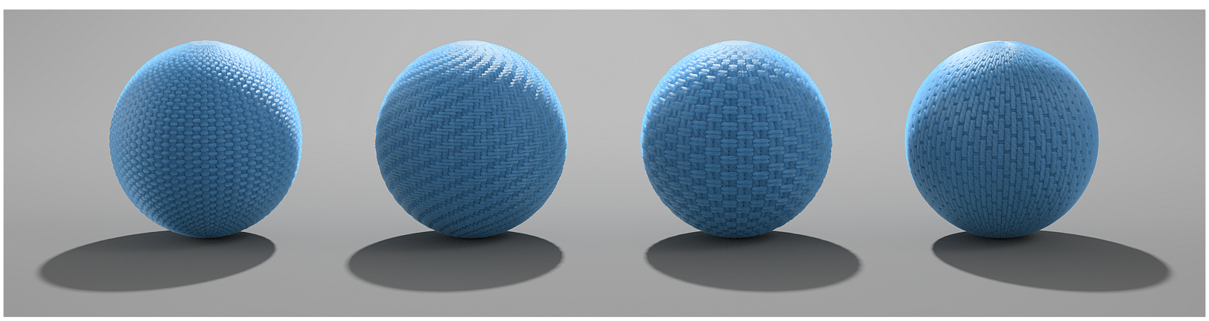

Figure 14: Example renders showing the different weave patterns presented to artists.

4 PRODUCTION RESULTS

The following examples show our technique employed in production renders for *Guardians of the Galaxy Vol. 2* (Figures 15 and 17) and *Alien: Covenant* (Figure 16).

Figure 15: Abilisk. Left to right: muscles; veins; shallow scattering; deep scattering.

Figure 16: Chest Burster. Layers of translucent structure.

5 FUTURE

5.1 Procedural volumes

We are currently looking into new distance sampling and transmittance algorithms that do not require strict bounds on the extinction coefficients used during sampling. In the case of procedurally defined interiors, localised approximate values for minimum and maximum extinction can be estimated from a few trial samples.

Figure 17: Gamora. An example of a digital double.

5.2 Microflakes

Part of our current development effort is focused around extending our volumetric model to support microflakes [Hei+15; DHd16]. We are interested in ways to represent procedurally generated surfaces as volumes and to offer artists a way to apply their existing surface-based look development techniques.

5.3 Mipmaps

We would like to investigate mip-mapping our voxelised weave patches. These could then be accessed and interpolated using ray differentials at render time. This could potentially reduce variance, especially with fine weave effects and flyaway fibres.

6 CONCLUSION

We have presented our approach to volumetrically shading skin and fabric, discussed aspects of its implementation and shown production examples of its use. We believe that it offers a good balance between visual quality, artist control and evaluation cost, and has proven to be robust in a production environment.

7 ACKNOWLEDGEMENTS

I would like to thank the shading team at Framestore, particularly Josh Bainbridge, Manuel Gamito and Serguei Lapine for developing much of the work that has been presented here.

REFERENCES

[Chi+16] M. J.-Y. Chiang, B. Bitterli, C. Tappan, and B. Burley. "A Practical and Controllable Hair and Fur Model for Production Path Tracing". In: *Computer Graphics Forum* 35.2 (2016). URL: https://benedikt-bitterli.me/pchfm/.

[Chr15] P. H. Christensen. "An Approximate Reflectance Profile for Efficient Subsurface Scattering". In: *ACM SIGGRAPH 2015 Talks.* Los Angeles, California: ACM, 2015. URL: https://graphics.pixar.com/library/ApproxBSSRDF/.

[CKB16] M. J.-Y. Chiang, P. Kutz, and B. Burley. "Practical and Controllable Subsurface Scattering for Production Path Tracing". In: *ACM SIGGRAPH 2016 Talks.* Anaheim, California: ACM, 2016. URL: https://www.disneyanimation.com/technology/publications/91.

[DHd16] J. Dupuy, E. Heitz, and E. d'Eon. "Additional Progress Towards the Unification of Microfacet and Microflake Theories". In: *Proceedings of the Eurographics Symposium on Rendering: Experimental Ideas & Implementations.* EGSR '16. Dublin, Ireland: Eurographics Association, 2016.

[Hd14] E. Heitz and E. d'Eon. "Importance Sampling Microfacet-Based BSDFs using the Distribution of Visible Normals". In: *Computer Graphics Forum.* Vol. 33. 4. 2014. URL: https://hal.inria.fr/hal-00996995v1.

[HDF15] J. Hanika, M. Droske, and L. Fascione. "Manifold next event estimation". In: *Computer Graphics Forum.* Vol. 34. 4. 2015. URL: https://jo.dreggn.org/home/#2015_mnee.

[Hei+15] E. Heitz, J. Dupuy, C. Crassin, and C. Dachsbacher. "The SGGX Microflake Distribution". In: *ACM Trans. Graph.* 34.4 (July 2015).

[HG41] L. G. Henyey and J. L. Greenstein. "Diffuse radiation in the galaxy". In: *The Astrophysical J.* 93 (1941).

[Jen+01] H. W. Jensen, S. R. Marschner, M. Levoy, and P. Hanrahan. "A Practical Model for Subsurface Light Transport". In: SIGGRAPH '01. New York, NY, USA: ACM, 2001. URL: http://www.graphics.stanford.edu/papers/bssrdf/.

[Khu+15] P. Khungurn, D. Schroeder, S. Zhao, K. Bala, and S. Marschner. "Matching Real Fabrics with Micro-Appearance Models". In: *ACM Trans. Graph.* 35.1 (Dec. 2015). URL: http://www.cs.cornell.edu/projects/ctcloth/#matching-cloth.

[Kin+13] A. King, C. Kulla, A. Conty, and M. Fajardo. "BSSRDF Importance Sampling". In: *ACM SIGGRAPH 2013 Talks.* Anaheim, California: ACM, 2013. URL: https://sites.google.com/site/ckulla/home/bssrdf-importance-sampling.

[NSJ14] J. Novák, A. Selle, and W. Jarosz. "Residual Ratio Tracking for Estimating Attenuation in Participating Media". In: *ACM Trans. Graph.* 33.6 (Nov. 2014). URL: http://drz.disneyresearch.com/~jnovak/publications/RRTracking/index.html.

[Sad+13] I. Sadeghi, O. Bisker, J. De Deken, and H. W. Jensen. "A Practical Microcylinder Appearance Model for Cloth Rendering". In: *ACM Trans. Graph.* 32.2 (Apr. 2013). URL: http://graphics.ucsd.edu/~iman/a_practical_microcylinder_appearance_model_for_cloth_rendering.php.

[Woo+65] E. Woodcock, T. Murphy, P. Hemmings, and S. Longworth. "Techniques used in the GEM code for Monte Carlo neutronics calculations in reactors and other systems of complex geometry". In: *Proc. Conf. Applications of Computing Methods to Reactor Problems.* Vol. 557. 1965.

[WXK14] C. Wang, F. Xie, and P. Krishnamachari. "Importance Sampling for a Microcylinder Based Cloth Bsdf". In: *ACM SIGGRAPH 2014 Talks.* Vancouver, Canada: ACM, 2014.

[ZLB16] S. Zhao, F. Luan, and K. Bala. "Fitting Procedural Yarn Models for Realistic Cloth Rendering". In: *ACM Trans. Graph.* 35.4 (July 2016).